



# **SDNRI 2016** Semaine du Document Numérique et de la Recherche d'Information



**9-11 Mars 2016**

**Toulouse**

## **Présidente du comité de programme CORIA**

Sylvie Calabretto, LIRIS – Lyon

## **Président du comité de programme CIFED**

Bertrand Coüasnon, IRISA – Rennes

## **Présidentes du comité de programme RJC**

CIFED : Sabine Barrat, LI – Tours

CORIA : Lorraine Goeuriot, LIG – Grenoble

## **Présidents du comité d'organisation**

Gilles Hubert, IRIT – Toulouse

Nadine Jessel, IRIT – Toulouse

Lynda Tamine-Lechani, IRIT – Toulouse

## **Conférenciers invités**

Gabriella Pasi, University of Milano-Bicocca

Benjamin Piwowarski, Université Pierre et Marie Curie

Mario Vento, University of Salerno



# Préface

Après Sousse en 2010, Bordeaux en 2012 et Nancy en 2014, l'ARIA (Association francophone de Recherche d'Information et Applications) et le GRCE (Groupement de Recherche en Communication Ecrite) ont décidé d'organiser simultanément les conférences CORIA et CIFED dans le cadre de la semaine du document numérique et de la recherche d'information (SDNRI) à Toulouse.

CORIA et CIFED sont les points de rassemblement des communautés francophones respectivement en recherche d'information, en analyse de l'écrit et des documents numérisés. Tout en préservant les spécificités de chaque conférence, cette édition constitue une opportunité pour les deux communautés de se retrouver autour de thématiques pour lesquelles il existe des synergies de recherche (recherche de documents multimédia, modèle d'interaction avec l'utilisateur, passage à l'échelle de système de recherche d'information, outils d'évaluation de performance pour la recherche d'information). CORIA a reçu 34 soumissions cette année. Le comité de programmes a sélectionné 15 papiers longs et 6 papiers courts. Les papiers retenus sont tous d'excellente qualité et recouvrent des thématiques de recherche très variées : analyse des réseaux sociaux, analyse et représentation des microblogs, expansion de requêtes, traitement de la langue, apprentissage, recherche d'information médicale et humanités numériques. Concernant CIFED, le comité de sélection a retenu cette année 21 papiers, sur les 24 soumis. Les contributions proposées couvrent largement le domaine de recherche : la classification de documents, l'analyse de la structure, la reconnaissance de documents graphiques, de document anciens, la reconnaissance de l'écriture manuscrite, ainsi que les problématiques de métriques pour l'évaluation. Des sessions communes CIFED-CORIA sont proposées et visent à permettre aux participants des deux conférences d'interagir autour de thématiques qui intéressent les deux communautés comme l'apprentissage ou les humanités numériques.

Trois conférenciers invités complètent le programme :

Gabriella Pasi est professeur à l'Université de Milano-Bicocca. Elle est un chercheur européen parmi les plus reconnus dans le domaine de la Recherche d'Information. À ce titre, elle est régulièrement sollicitée par les institutions d'évaluation de la recherche de plusieurs pays (Italie, France, Suisse, Croatie, Espagne) et par la Commission pour l'évaluation de projets européens (elle a été longtemps membre du Panel of Experts de Computer Science pour les Starting Grants et les Consolidators Grants du Programme Ideas du European Research Council). Elle a été invitée en

tant que conférencier dans une quinzaine de conférences internationales. Gabriella Pasi a beaucoup œuvré dans le rapprochement du monde industriel et de l'université et a été à l'initiative de plusieurs travaux et projets communs dont des projets européens (en particulier, elle a aussi cordonné un projet Européen). Elle est l'auteur de 36 chapitres dans des ouvrages internationaux portant sur des problématiques variées liées essentiellement à la Recherche d'Information. Elle est également auteur de 48 publications dans des journaux internationaux, et globalement l'auteur de plus de 200 publications scientifiques ;

Benjamin Piwowarski est chargé de recherche 1<sup>re</sup> classe du CNRS au LIP6 (UPMC). Son domaine de recherche est celui de l'accès à l'information et de l'apprentissage statistique appliqué, en particulier l'apprentissage de représentations. Il a travaillé sur la recherche d'information XML (thèse 1999-2003), sur les bases de données XML (post-doctorat 2004-06 à l'université du Chili), sur l'analyse des interactions entre un utilisateur et un moteur de recherche (Yahoo! 2006-08) et sur l'application du formalisme mathématique de la physique quantique aux modèles d'accès à l'information (2008-11). Ses intérêts actuels portent sur (l'apprentissage de) représentation des données textuelles de graphes. Il a été fortement impliqué dans l'initiative INEX (2004-07), et a organisé un workshop (IRGM'07), deux tutoriels (ECIR 2012 et ICTIR 2013), et a participé l'organisation de CORIA 2015 ;

Mario Vento est membre fellow de l'International Association for Pattern Recognition (IAPR). Il est actuellement professeur d'informatique et d'intelligence artificielle à l'université de Salerne (Italy), où il dirige l'Artificial Vision Lab. De 2002 à 2006 il a été président de l'IAPR Technical Committee TC15 sur "Graph Based Representation in Pattern Recognition", et depuis 2003 éditeur associé de "Electronic Letters on Computer Vision and Image Analysis". Ses recherches s'inscrivent dans les domaines de l'intelligence artificielle, l'analyse d'image, la reconnaissance de formes, l'apprentissage automatique et la vision par ordinateur. Plus précisément ses travaux portent sur l'analyse et l'interprétation de vidéos en temps réel pour des applications de surveillance de la circulation et de vidéosurveillance, sur les méthodes de classification statistiques, syntaxiques ou structurelles, sur la mise en correspondance exacte et inexacte de graphes, sur la classification multi-expert et les méthodes d'apprentissage pour des descriptions structurelles. Il est l'auteur de plus de 170 publications dans des journaux internationaux et conférences internationales.

Pendant la conférence CIFED-CORIA 2016 sont également organisées les Rencontres Jeunes Chercheurs (RJC) en recherche d'information et en analyse de l'écrit et des documents numérisés. Elles ont pour objectif de permettre à tous les doctorants de présenter leur problématique de recherche, d'établir des contacts avec des équipes travaillant sur des domaines similaires ou connexes, et d'offrir à l'ensemble de la communauté un aperçu des axes de recherche actuels. Cette année, les RJC ont permis à 7 jeunes chercheurs de proposer leur travail, ainsi qu'à 12 autres de s'exercer à la relecture d'articles. Une session Démonstrations est également organisée pendant la conférence afin de favoriser des échanges



scientifiques autour de systèmes fonctionnels. Deux événements satellites viennent enrichir scientifiquement la quatrième édition de la SDNRI. Il s'agit de l'atelier en Recherche d'Information SEmantique RISE et du Hackday qui vise à fédérer et dynamiser les communautés CORIA et CIFED autour d'un défi centré sur la Recherche d'Information et du Document. Nous tenons à remercier très chaleureusement l'ensemble des membres des comités de programme pour l'évaluation des soumissions, les organisateurs toulousains pour leur excellent accueil chaleureux, ainsi que toutes les personnes qui se sont investies pour la réussite de cet événement. Nous adressons également nos sincères remerciements aux institutions et organisations qui soutiennent la SDNRI. Nous remercions aussi tous les auteurs qui ont envoyé une soumission. Nos derniers remerciements vont vers l'ensemble des participants à la conférence.

Sylvie Calabretto (LIRIS/INSA de Lyon),  
Présidente du comité de programme CORIA

Bertrand Coüasnon (IRISA/INSA de Rennes),  
Président du comité de programme CIFED

Sabine Barrat (LI/Université de Tours) et Lorraine Goeuriot (LIG /Université de Grenoble),  
Présidentes du comité de programme RJC

## Comités de programme

### CORIA

#### Présidente

- Sylvie Calabretto (LIRIS - INSA de Lyon)

#### Membres

- Massih-Reza Amini (LIG)
- Rachid Ahmed-Ouamer (LARI University of Tizi-Ouzou)
- Ikram Amous (Université de Sfax, Tunisie)
- Michel Beigbeder (Ecole Nationale Supérieure des Mines de St-Etienne)
- Patrice Bellot (LSIS - Univ. Marseille)
- Abdelmajid Ben Hamadou (Université de Sfax, Tunisie)
- Catherine Berrut (LIG, Université Joseph Fourier Grenoble I)
- Mohand Boughanem (IRIT - Toulouse)
- Anne Boyer (Nancy Université)
- Armelle Brun (Université Nancy 2)
- Guillaume Cabanac (IRIT - Toulouse)
- Sylvie Cazalens (LIRIS - INSA de Lyon)
- Max Chevalier (IRIT - Toulouse)

- Jean-Pierre Chevallet (Grenoble Alpes University)
- Vincent Claveau (IRISA - CNRS)
- Stéphane Clinchant (Xerox Research Centre Europe)
- Thomas Demeester (Ghent University)
- Nathalie Denos (Laboratoire LIG - Grenoble Universités)
- Ludovic Denoyer (LIP6-Université Paris 6)
- Habiba Drias (University of Science and Technology USTHB Algiers)
- Antoine Doucet (L3i, University of La Rochelle)
- Elöd Egyed-Zsigmond (LIRIS - INSA de Lyon)
- Omar El Beqqali (Universite FES)
- Marc El-Bêze (LIA, UAPV)
- Rima Faiz (IHEC – University of Carthage)
- Olivier Ferret (CEA LIST)
- Patrick Gallinari (LIP6-Université Paris 6)
- Eric Gaussier (Grenoble Alpes University)
- Mathias Géry (LaHC, Université de Lyon, St-Etienne)
- Gregory Grefenstette (INRIA)
- Brigitte Grau (LIMSI-CNRS, ENSIIE)
- Lorraine Gueuriot (Laboratoire d’informatique de Grenoble)
- Gilles Hubert (IRIT - Toulouse)
- Ali Jaoua (Qatar)
- Guy Lapalme (Université de Montréal, Canada)
- Léa Laporte (LIRIS-INSA de Lyon)
- Christine Largeron (Université de Lyon)
- Edmond Lassalle (Orange Labs)
- Benjamin Lecouteux (LIG-Université Grenoble-Alpes)
- Vincent Leroy (University of Grenoble - CNRS)
- Anne-Laure Ligozat (LIMSI-CNRS)
- Philippe Lopisteguy (LIUPPA - IUT Bayonne)
- Pierre Maret (LHC Université de St-Etienne)
- Mihaella Mathieu-Juganaru (Ecole Nationale Supérieure des Mines de St-Etienne)
- Annabelle Mercier (Laboratoire LCIS)
- Marie-Francine Moens (Katholieke Universiteit Leuven, Belgique)
- Josiane Mothe (IRIT - Toulouse)
- Henning Mueller (HES-SO)
- Philippe Mulhem (LIG-CNRS)
- An-Te Nguyen (University of Science, HoChiMinh City)
- Jian-Yun Nie (Université de Montréal)
- Iadh Ounis (University of Glasgow)
- Gabriella Pasi (University of Milano Bicocca)
- Karen Pinel-Sauvagnat (IRIT - Toulouse)
- Benjamin Piwowarski (CNRS / University Pierre et Marie Curie)
- Pierre-Edouard Portier (INSA Lyon)
- Georges Quénot (Laboratoire d’Informatique de Grenoble, CNRS)
- Daniel Rocacher (Irisa)
- Mathieu Roche (Cirad, TETIS)

- Catherine Roussey (Irstea)
- Fatiha Sadat (UQAM)
- Eric Sanjuan (LIA - Université d'Avignon)
- Jacques Savoy (University of Neuchatel)
- Christoph Schommer (University of Luxembourg)
- Florence Sèdes (IRIT - Toulouse)
- Malika Smail-Tabbone (University of Lorraine)
- Chantal Soulé-Dupuy (IRIT - Toulouse)
- Laure Soulier (LIP6 - Université Paris 6)
- Lynda Tamine-Lechani (IRIT - Toulouse)
- Xavier Tannier (LIMSI-Université Paris-Sud)
- Isabelle Tellier (Lattice)
- Laurent Ughetto (Université Rennes 2)
- Nadine Jessel-Baptiste (IRIT - Toulouse)
- Thomas François (Université catholique de Louvain, Belgique)
- Antoine Doucet (University of La Rochelle)

#### **Relecteur additionnel**

- Ilyes Khennak (USTHB)

#### **CIFED**

##### **Président**

- Bertrand Couïasnon IRISA / INSA de Rennes (France)

##### **Président**

- Sébastien Adam (LITIS - Rouen, France)
- Eric Anquetil (IRISA - Rennes, France)
- Abdel Belaid (LORIA - Nancy, France)
- Stéphane Bres (LIRIS - Lyon, France)
- Clément Chatelain (LITIS - Rouen, France)
- Florence Cloppet (LIPADE - Paris 5, France)
- Mathieu Delalandre (LI - Tours, France)
- Véronique Eglin (LIRIS / INSA de Lyon, France)
- Elöd Egyed-Zsigmond (LIRIS / INSA de Lyon, France)
- Najoua Essoukri Ben Amara (ENIM - Sousse, Tunisie)
- Pierre Heroux (LITIS - Rouen , France)
- Laurent Heutte (LITIS - Rouen , France)
- Nicholas Journet (LABRI - univ. Bordeaux , France)
- Slim Kanoun (ENIS - Sfax, Tunisie)
- Christopher Kermorvant (Teklia , France)
- Bart Lamiroy (LORIA - Nancy, France)
- Frank Lebourgeois (LIRIS / INSA de Lyon, France)
- Aurélie Lemaitre (IRISA - Rennes, France)
- Laurence Likforman (TELECOM PARISTECH, France)
- Josep Lladós (CVC - Barcelone, Espagne)

- Simone Marinai (DINFO - Florence, Italie)
- Harold Mouchère (IRCCyN - Nantes, France)
- Rémy Mullot (L3I - La Rochelle, France)
- Jean-Marc Ogier (L3I - La Rochelle, France)
- Thierry Paquet (LITIS - Rouen, France)
- Vincent Poulain D’Andecy (ITESOFT, France)
- Nicolas Ragot (LI - Tours, France)
- Jean-Yves Ramel (LI - Tours, France)
- Oriol Ramos-Terrades (CVC - UAB Barcelona, Espagne)
- Marçal Rusiñol (CVC - Barcelone, Espagne)
- Robert Sabourin (LIVIA - Montréal, Canada)
- Pascale Sébillot (IRISA - Rennes , France)
- Antoine Tabbone (LORIA - Nancy, France)
- Christian Viard-Gaudin (IRCCyN – Université de Nantes , France)
- Nicole Vincent (LIPADE - Paris 5 , France)
- Muriel Visani (L3I - La Rochelle , France)
- Laurent Wendling (LIPADE - Paris 5 , France)

## **RJC**

### **Présidents**

- CIFED : Sabine Barrat (LI/Université de Tours)
- CORIA : Lorraine Goeuriot (LIG/Université de Grenoble)

### **Membres**

- CIFED
  - Elodie Carel (L3I- Université de La Rochelle)
  - Joseph Chazalon (L3I- Université de La Rochelle)
  - Mickaël Coustaty (L3I- Université de La Rochelle)
  - Mehdi Felhi, Netatmo
  - Nathalie Girard, LI, Université François Rabelais, Tours
  - Maroua Hammami, LITIS, Université de Rouen
  - Muhammad Muzzamil Luqman, Centre de Bioinformatique de Bordeaux (CBiB)
  - Frédéric Rayar, LI, Université François Rabelais, Tours
  - Christophe Rigaud, L3I, Université de La Rochelle
  - Oriol Ramos-Terrades, CVC de Barcelone
  - Romain Raveaux, LI, Université François Rabelais, Tours
- CORIA
  - Romain Beaumont, LIMSI, Paris
  - Remi Bois, IRISA, Rennes
  - Xavier Bost, LIA, Avignon
  - Adrian Chifu, LSIS, Marseille
  - Mateusz Budnik, LIG, Grenoble
  - Lina Fahed, LORIA, Nancy
  - Martin Gleize, LIMSI, Paris
  - Thi-Bich-Ngoc Hoang, IRIT, Toulouse

- Anaïs Ollagnier, LSIS-OpenEdition, Aix-Marseille
- Nawal Ould Amer, LIG, Grenoble

## **Comité d'organisation**

### **Présidents**

- Gilles Hubert, IRIT, Université Toulouse 3
- Nadine Jessel, IRIT, Université Toulouse 2
- Lynda Tamine-Lechani, IRIT, Université Toulouse 3

### **Membres**

- Rafik Abbes, IRIT, Université Toulouse 3
- Ismael Badache, IRIT, Université Toulouse 3
- Lamjed Ben Jabeur, IRIT, Université Toulouse 3
- Guillaume Cabanac, IRIT, Université Toulouse 3
- Abdelhamid Chellal, IRIT, Université Toulouse 3
- Ophélie Fraasier, IRIT, Université Toulouse 3
- Jonathan Louëdec, IRIT, Université Toulouse 3
- Gia-Hung Nguyen, IRIT, Université Toulouse 3
- Josiane Mothe, IRIT, Université Toulouse 2
- Bilel Moulahi, IRIT, Université Toulouse 3
- Manel Mezghani, IRIT, Université Toulouse 3
- Sirynia On-At, IRIT, Université Toulouse 3
- Thomas Palmer, IRIT, Université Toulouse 3
- André Péninou, IRIT, Université Toulouse 2
- Yoann Pitarch, IRIT, Université Toulouse 3
- Florence Sèdes, IRIT, Université Toulouse 3
- Thibaut Thonet, IRIT, Université Toulouse 3
- Eya Znaidi, IRIT, Université Toulouse 3



# Table des matières

|                                                                                                                                                                                                                                                       |           |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| <b>Conférences invitées</b>                                                                                                                                                                                                                           | <b>1</b>  |
| <b>Écriture manuscrite</b>                                                                                                                                                                                                                            |           |
| <b>CIFED</b>                                                                                                                                                                                                                                          | <b>9</b>  |
| Exploitation de l'échelle d'écriture pour améliorer la reconnaissance automatique des documents manuscrits arabe<br><i>Edgard Chammas, Chafic Mokbel, Laurence Likforman-Sulem</i>                                                                    | 11        |
| Un modèle syllabique pour la reconnaissance de l'écriture<br><i>Wassim Swaileh, Kamel Ait Mhand, Thierry Paquet</i> . . .                                                                                                                             | 17        |
| La CTC et son intrigant label « blank » : étude comparative de méthodes d'entraînement de réseaux de neurones pour la reconnaissance d'écriture manuscrite<br><i>Théodore Bluche, Christopher Kermorvant, Hermann Ney, Jerome Louradour</i> . . . . . | 23        |
| <b>Réseaux sociaux</b>                                                                                                                                                                                                                                |           |
| <b>CORIA</b>                                                                                                                                                                                                                                          | <b>29</b> |
| Bandit contextuel pour la capture de données temps réel sur les médias sociaux<br><i>Thibault Gisselbrecht, Sylvain Lamprier, Patrick Gallinari</i>                                                                                                   | 31        |
| Recommandation et analyse de sentiments dans un espace latent textuel<br><i>Charles-Emmanuel Dias, Vincent Guigue, Patrick Gallinari</i>                                                                                                              | 47        |
| Apprentissage de représentations probabilistes pour la prédiction de diffusions d'informations sur les réseaux sociaux<br><i>Simon Bourigault, Sylvain Lamprier, Patrick Gallinari</i> . .                                                            | 63        |
| Suggestion d'experts pour renouveler le comité de programme d'une conférence<br><i>Hong Diep Tran, Guillaume Cabanac, Gilles Hubert</i> . . .                                                                                                         | 79        |
| <b>Humanités numériques</b>                                                                                                                                                                                                                           |           |
| <b>CORIA – CIFED</b>                                                                                                                                                                                                                                  | <b>95</b> |
| Nouveau modèle pour la datation automatique de photographies à partir de caractéristiques visuelles (CIFED)<br><i>Paul Martin, Antoine Doucet, Frederic Jurie</i> . . . . .                                                                           | 97        |
| Cascade de CRFs et SVM pour la détection de références bibliographiques diffuses dans les articles scientifiques (CORIA)<br><i>Anaïs Ollagnier, Sébastien Fournier, Patrice Bellot</i> . . . .                                                        | 103       |

|                                                                                                                                                                                                           |            |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| Audit d'une base de documents étiquetée (CIFED)<br><i>Romain Giot, Romain Bourqui, Nicholas Journet, Vialard Anne</i> . . . . .                                                                           | 117        |
| Was it better before? Automated Quotation Detection in Ancient Texts (CORIA)<br><i>Samuel Gesche, Előd Egyed-Zsigmond, Sylvie Calabretto</i> .                                                            | 123        |
| <b>Documents graphiques</b>                                                                                                                                                                               |            |
| <b>CIFED</b>                                                                                                                                                                                              | <b>139</b> |
| Extraction de patterns successifs dans des images de document à partir de la combinaison d'histogrammes de forces et de droites discrètes<br><i>Isabelle Debled-Rennesson, Laurent Wendling</i> . . . . . | 141        |
| Extraction des bulles de bandes dessinées<br><i>Christophe Rigaud, Jean-Christophe Burie, Jean-Marc Ogier</i>                                                                                             | 147        |
| Using BLSTM for Interpretation of 2D Languages – Case of Handwritten Mathematical Expressions<br><i>Ting Zhang, Harold Mouchère, Christian Viard-Gaudin.</i> .                                            | 153        |
| <b>Expansion de requêtes</b>                                                                                                                                                                              |            |
| <b>CORIA</b>                                                                                                                                                                                              | <b>169</b> |
| Query Expansion by Local Context Analysis<br><i>Liana Ermakova, Josiane Mothe</i> . . . . .                                                                                                               | 171        |
| Dimensionnalité intrinsèque dans les espaces de représentation des termes et des documents<br><i>Vincent Claveau</i> . . . . .                                                                            | 187        |
| Expansion de requêtes par apprentissage<br><i>Ahlem Bouziri, Chiraz Latiri, Éric Gaussier</i> . . . . .                                                                                                   | 203        |
| <b>Rencontres Jeunes Chercheurs</b>                                                                                                                                                                       | <b>219</b> |
| Tied Spatial Transformer Networks for Character Recognition<br><i>Bogdan-Ionut Cirstea, Laurence Likforman-Sulem</i> . . . . .                                                                            | 221        |
| Building a Knowledge Base Using Microblogs: the Case of Festivals and Location-Based Events<br><i>Hoang Thi Bich Ngoc, Josiane Mothe</i> . . . . .                                                        | 231        |
| Processing Natural Language Queries to Disambiguate Named Entities and Extract Users' Goals : Application to e-Tourism<br><i>Sanjay Kamath, Lorraine Goeuriot and Marie-Christine Fauvet</i> . . . . .    | 241        |
| Extraction automatique de termes-clés : Comparaison des méthodes non supervisées de la littérature<br><i>Josiane Mothe, Faneva Ramianandroa</i> . . . . .                                                 | 251        |
| ANASTASIA : recommandation de séquences d'activités spatio-temporelles<br><i>Diana Nurbakova, Léa Laporte, Sylvie Calabretto, Jérôme Gensel</i> . . . . .                                                 | 261        |



|                                                                                                                                                                                                                                          |            |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| Étude préliminaire à la recherche de photographies muséales en mobilité<br><i>Maxime Portaz, Philippe Mulhem, Jean-Pierre Chevallet</i> .                                                                                                | 271        |
| Using BLSTM for Interpretation of 2D Languages – Case of Handwritten Mathematical Expressions<br><i>Ting Zhang, Harold Mouchère, Christian Viard-Gaudin</i> .                                                                            | 281        |
| <b>Structures de documents</b>                                                                                                                                                                                                           |            |
| <b>CIFED</b>                                                                                                                                                                                                                             | <b>283</b> |
| Utilisation de la couleur pour l'extraction de tableaux dans des images de documents<br><i>Héloïse Alhéritière, Florence Cloppet, Camille Kurtz, Nicole Vincent</i> . . . . .                                                            | 285        |
| Time-efficient Logo Spotting using Text/Non-text Separation as Preprocessing and Approximate Nearest Neighbor Search<br><i>Viet Phuong Le, Nibal Nayef, Muriel Visani, Jean-Marc Ogier, Cao De Tran</i> . . . . .                        | 291        |
| Localisation automatique de champs de saisie sur des images de formulaires couleur par isomorphisme de sous-graphe<br><i>Maroua Hammami, Pierre Héroux, Sébastien Adam, Vincent Poulain d'Andecy</i> . . . . .                           | 297        |
| Inférence semi-automatique et interactive de règles sans vérité terrain<br><i>Cérés Carton, Aurélie Lemaitre, Bertrand Coüasnon</i> . . .                                                                                                | 303        |
| <b>Domaines spécifiques</b>                                                                                                                                                                                                              |            |
| <b>CORIA</b>                                                                                                                                                                                                                             | <b>309</b> |
| Quels facteurs de pertinence pour la recherche de produits e-commerce?<br><i>Laure Soulier, Lamjed Ben Jabeur, Paul Mousset, Lynda Tamine</i> . . . . .                                                                                  | 311        |
| RI-TAL : le TAL au service de la RI<br><i>Laurent Candillier, Julien Hénot</i> . . . . .                                                                                                                                                 | 327        |
| Répondre à des requêtes cliniques PICO<br><i>Eya Znaidi, Lynda Tamine, Chiraz Latiri</i> . . . . .                                                                                                                                       | 343        |
| <b>Documents anciens</b>                                                                                                                                                                                                                 |            |
| <b>CIFED</b>                                                                                                                                                                                                                             | <b>359</b> |
| Reconnaissance et classification de lettrines à base des descripteurs de bas niveau et de représentation structurelle<br><i>Maroua Mehri, Pierre Héroux, Mickaël Coustaty, Petra Gomez-Krämer, Julien Lerouge, Rémy Mullot</i> . . . . . | 361        |
| Alignement texte-image sans apprentissage pour les manuscrits médiévaux<br><i>Yann Leydier, Véronique Eglin, Stéphane Bres, Dominique Stutzmann</i> . . . . .                                                                            | 367        |

|                                                                                                                                                                                                             |            |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| Adaptation des caractéristiques pseudo-haar pour le word spotting dans les documents manuscrits<br><i>Adam Ghorbel, Nicole Vincent and Jean-Marc Ogier</i> . . .                                            | 373        |
| <b>Microblogs</b>                                                                                                                                                                                           |            |
| <b>CORIA</b>                                                                                                                                                                                                | <b>379</b> |
| Synthèse de flux de messages en temps réel<br><i>Abdelhamid Chellal, Mohand Boughanem, Bernard Dousset</i>                                                                                                  | 381        |
| Représentation temporelle des mots : application au clustering de micro-blogs<br><i>Željko Kraljević, Nicolas Baskiotis, Benjamin Piwowarski, Patrick Gallinari</i> . . . . .                               | 397        |
| Tweet Contextualization Approach Based on Wikipedia and Dbpedia<br><i>Meriem Amina Zingla, Chiraz Latiri, Yahya Slimani, Catherine Berrut, Philippe Mulhem</i> . . . . .                                    | 411        |
| <b>Classification de documents</b>                                                                                                                                                                          |            |
| <b>CIFED</b>                                                                                                                                                                                                | <b>427</b> |
| Extension automatique d'annotation et classification de documents en utilisant un modèle graphique probabiliste<br><i>Abdessalem Bouzaïeni, Sabine Barrat, Salvatore Tabbone</i> .                          | 429        |
| Classification et extraction des documents complexes à partir des images issues d'un périphérique mobile : application aux documents d'identité<br><i>Ahmad Montaser Awal, Abdullah Almaksour</i> . . . . . | 435        |
| Filtrage de descripteurs locaux pour l'amélioration de la détection de documents<br><i>Marçal Rusiñol, Joseph Chazalon, Jean-Marc Ogier</i> . . . .                                                         | 441        |
| <b>Traitement de la langue</b>                                                                                                                                                                              |            |
| <b>CORIA</b>                                                                                                                                                                                                | <b>447</b> |
| Un modèle probabiliste pour la détection de l'incertitude dans le langage naturel<br><i>Pierreantoine Jean, Sébastien Harispe, Sylvie Ranwez, Patrice Bellot, Jacky Montmain</i> . . . . .                  | 449        |
| Structured Indexing Model for Cross-Language Information Retrieval<br><i>Chedi Bechikh Ali, Hatem Haddad</i> . . . . .                                                                                      | 465        |
| Génération automatique d'une ontologie dans le domaine des ressources humaines<br><i>Rémy Kessler, Eric Tondo, Guy Lapalme</i> . . . . .                                                                    | 479        |
| <b>Apprentissage et métriques</b>                                                                                                                                                                           | <b>495</b> |
| Réseaux de neurones pour la représentation des contextes continus des mots (CORIA)<br><i>Killian Janod, Mohamed Morchid, Richard Dufour, Georges Linarès</i> . . . . .                                      | 497        |

|                                                                                                                       |     |
|-----------------------------------------------------------------------------------------------------------------------|-----|
| Métriques statistiques pour l'évaluation de performance en présence de vérité terrain imprécise (CIFED)               |     |
| <i>Bart Lamiroy, Pascal Pierrot</i>                                                                                   | 511 |
| Apprentissage d'inférences par édition d'arbres pour répondre à des questions (CORIA)                                 |     |
| <i>Martin Gleize, Brigitte Grau</i>                                                                                   | 517 |
| Vers une approche utilisant l'apprentissage de métrique pour du clustering semi-supervisé interactif d'images (CIFED) |     |
| <i>Viet Minh Vu, Hien Phuong Lai, Muriel Visani</i>                                                                   | 533 |
| Apprentissage de modèles de langue neuronaux pour la recherche d'information (CORIA)                                  |     |
| <i>Nicolas Desprès, Sylvain Lamprier, Benjamin Piwowski</i>                                                           | 539 |





# CORIA – CIFED 2016

## Conférences invitées

Présidents :

*Josiane Mothe,  
Sylvie Calabretto,  
XX*



# Texte et Représentation en Recherche d'Information

par Benjamin Piwowarski, Université Pierre et Marie Curie

## Abstract

tbd

## Bio

Benjamin Piwowarski est chargé de recherche 1ère classe du CNRS au LIP6 (UPMC). Son domaine de recherche est celui de l'accès à l'information et de l'apprentissage statistique appliqué, en particulier l'apprentissage de représentations. Il a travaillé sur la recherche d'information XML (thèse 1999-2003), sur les bases de données XML (post-doctorat 2004-06 à l'université du Chili), sur l'analyse des interactions entre un utilisateur et un moteur de recherche (Yahoo! 2006-08) et sur l'application du formalisme mathématique de la physique quantique aux modèles d'accès à l'information (2008-11). Ses intérêts actuels portent sur (l'apprentissage de) représentation des données textuelles de graphes. Il a été fortement impliqué dans l'initiative INEX (2004-07), et a organisé un workshop (IRGM'07), deux tutoriels (ECIR 2012 et ICTIR 2013), et a participé l'organisation de CORIA 2015.





# On the Issue of Aggregation in Information Retrieval

par Gabriella Pasi, University of Milano-Bicocca

## Abstract

This lecture will address the issue of defining flexible approaches to Information Retrieval by means of the usage of aggregation operators in various phases of the retrieval process. In particular, these approaches rely on the interpretation of Information Retrieval as a Multi-Criteria Decision Making (MCDM) problem, from various perspectives. The first, more straightforward perspective, is to interpret the overall IR process as a MCDM process aimed at selecting the best alternatives (documents) based on the assessment of the performance of multiple criteria (the keywords specified in a user's query). Another and strongly related perspective is to see the assessment of the overall relevance estimate of a document (still an alternative) to a query as the process of evaluating the performance of several relevance dimensions (e.g., topicality, novelty, recency, etc.), which in this case represent the criteria to be aggregated. Another process that may require the application of appropriate aggregation operators is the indexing process, when applied to structured documents. Metasearch constitutes another interesting task that can be seen as an instance of a Multi-Expert Decision Making (MEDM) problem, also strongly relying on the appropriate choice of an aggregation operator. By this task a user query is separately evaluated by different search engines, each one providing its own relevance assessment of the considered documents. Metasearch aims to merge the ranked lists generated by the various search engines (experts) in response to a query, to the aim of providing a unique, consensual ranked list of results. A quite interesting aspect implied by the above interpretations of various phases of the IR process is that the choice of different aggregation operators can produce different results. In other words, the semantics of aggregation implies an interpretation of the affected process. For example, if considering the aggregation of different relevance assessments for a same query and the same documents, distinct rankings can be obtained by applying distinct aggregation strategies. Despite the potential impact of aggregation on the whole IR process, this aspect has not received the proper attention in the literature. Only recently some approaches have appeared demonstrating the importance of this issue, and its potential impact on the searching process. This lecture aims to shortly review the main contributions that in the literature have made use of aggregation operators in Information Retrieval.

## Bio

Gabriella Pasi est Professeur à l'Université de Milano-Bicocca. Elle est un chercheur européen parmi les plus reconnus dans le domaine de

la Recherche d'Information. A ce titre, elle est régulièrement sollicitée par les institutions d'évaluation de la recherche de plusieurs pays (Italie, France, Suisse, Croatie, Espagne) et par la Commission pour l'évaluation de projets européens (elle a été longtemps membre du Panel of Experts de Computer Science pour les Starting Grants et les Consolidators Grants du Programme Ideas du European Research Council). Elle a été invitée en tant que conférencier dans une quinzaine de conférences internationales. Gabriella Pasi a beaucoup œuvré dans le rapprochement du monde industriel et de l'université et a été à l'initiative de plusieurs travaux et projets communs dont des projets européens (en particulier, elle a aussi coordonné un projet Européen). Elle est l'auteur de 36 chapitres dans des ouvrages internationaux portant sur des problématiques variées liées essentiellement à la Recherche d'Information. Elle est également auteur de 48 publications dans des journaux internationaux, et globalement l'auteur de plus de 200 publications scientifiques.

# About the Use of Graphs in Pattern Recognition

par Mario Vento, University of Salerno

## Abstract

The talk presents an overview of how graphs have been used in Pattern Recognition; their use is strongly suggested in all those cases in which the objects of interest are represented by means of parts suitably connected with each other, so as to arrive to a description in terms of a relational graph. The overview is aimed at reveal the rationale inspiring the papers published in these years, so as to roughly classify them. Despite the extent of scientific production in this field, it is possible to identify three historical periods, each having its own connotation common to most of the corresponding papers, which are called here as the pure, the impure and extreme periods. The use of a graph-based pattern representation induces the need to formulate the main operations required in Pattern Recognition in terms of operations on graphs : classification, usually intended as the comparison between an object and a set of prototypes, and learning, which is the process for obtaining a model of a class starting from a set of known samples, are among the key issues that must be addressed using graph-based techniques. Starting from the way in which these basic operations are defined and carried out it is possible to identify the above mentioned periods and the talk is focused on the most relevant techniques proposed along these years, so as to give the audience an overview of the field.

## Bio

Mario Vento est membre fellow de l'International Association for Pattern Recognition (IAPR). Il est actuellement professeur d'informatique et d'intelligence artificielle à l'université de Salerne (Italy), où il dirige l'Artificial Vision Lab. De 2002 à 2006 il a été président de l'IAPR Technical Committee TC15 sur "Graph Based Representation in Pattern Recognition", et depuis 2003 éditeur associé de "Electronic Letters on Computer Vision and Image Analysis". Ses recherches s'inscrivent dans les domaines de l'intelligence artificielle, l'analyse d'image, la reconnaissance de formes, l'apprentissage automatique et la vision par ordinateur. Plus précisément ses travaux portent sur l'analyse et l'interprétation de vidéos en temps réel pour des applications de surveillance de la circulation et de vidéosurveillance, sur les méthodes de classification statistiques, syntaxiques ou structurelles, sur la mise en correspondance exacte et inexacte de graphes, sur la classification multi-expert et les méthodes d'apprentissage pour des descriptions structurelles. Il est l'auteur de plus de 170 publications dans des journaux internationaux et conférences internationales.





CORIA – CIFED 2016

Écriture manuscrite  
CIFED

Président : *XX*



Papier absent

Papier absent



Papier absent

Papier absent

Papier absent



Papier absent

Papier absent

Papier absent

Papier absent



Papier absent



Papier absent

La CTC et son intrigant label « blank » : étude comparative de  
méthodes d'entraînement de réseaux de neurones pour la reconnaissance  
d'écriture manuscrite

Papier absent

*Théodore Bluche, Christopher Kermorvant, Hermann Ney, Jerome  
Louradour*

Papier absent

La CTC et son intrigant label « blank » : étude comparative de  
méthodes d'entraînement de réseaux de neurones pour la reconnaissance  
d'écriture manuscrite

Papier absent

*Théodore Bluche, Christopher Kermorvant, Hermann Ney, Jerome  
Louradour*

Papier absent







CORIA – CIFED 2016

Réseaux sociaux  
CORIA

Président : *Mohand Boughanem*

La CTC et son intrigant label « blank » : étude comparative de  
méthodes d'entraînement de réseaux de neurones pour la reconnaissance  
d'écriture manuscrite

---

# Bandit Contextuel pour la Capture de Données Temps Réel sur les Médias Sociaux

**Thibault Gisselbrecht<sup>\*+</sup> — Sylvain Lamprier<sup>\*</sup> — Patrick Gallinari<sup>\*</sup>**

<sup>\*</sup> Sorbonne Universités, UPMC Univ Paris 06  
CNRS, LIP6 UMR 7606, 4 place Jussieu, 75005 Paris, France  
`prénom.nom@lip6.fr`

<sup>+</sup> IRT SystemX, 8 Avenue de la Vauve, 91120 Palaiseau, France  
`thibault.gisselbrecht@irt-systemx.fr`

---

*RÉSUMÉ. La plupart des médias sociaux offrent un accès aux flux de données produites par leurs utilisateurs. L'utilisation des API fournies pour collecter ces données, relativement à un besoin spécifique, peut se révéler être une tâche complexe car elle nécessite une sélection soignée des sources. Cela représente un problème particulièrement difficile dans les réseaux sociaux de grandes tailles étant donné le nombre important d'utilisateurs potentiellement intéressants, la non-stationnarité intrinsèque de leur comportement, et les restrictions d'accès aux données. Dans cet article, nous proposons une approche permettant d'anticiper les profils les plus susceptibles de publier des contenus pertinents et de sélectionner un sous ensemble de comptes à chaque itération. Nous formalisons cette tâche comme un problème de bandit contextuel avec sélections multiples. Les expérimentations menées sur le réseau social Twitter montrent l'efficacité de notre approche dans un scénario réel.*

*ABSTRACT. Social media usually provide streaming data access that enable dynamic capture of the social activity of their users. Leveraging such APIs for collecting data that satisfy a given pre-defined need may constitute a complex task, that implies careful stream selections. On large social media, this represents a very challenging task due to the huge number of potential targets, the intrinsic non-stationarity of user's behavior, and restricted access to the data. We propose an approach that anticipates which profiles are likely to publish relevant contents and dynamically selects a subset of accounts to follow at each iteration using a contextual bandit algorithm. We conduct experiments on Twitter that demonstrate the empirical effectiveness of our approach in real-world settings.*

*MOTS-CLÉS : Bandit Mancho, Apprentissage Statistique, Médias Sociaux.*

*KEYWORDS: Multi-armed Bandit, Machine Learning, Social Media.*

---

## 1. Introduction

Au cours des dernières années, de nombreux réseaux sociaux permettant aux utilisateurs de publier et de partager du contenu en ligne ont fait leur apparition. Par exemple Twitter, avec 302 millions d'utilisateurs actifs et plus de 500 millions de messages chaque jour, est l'un des principaux acteurs du marché. Ces médias sociaux sont devenus une source de données très importante pour de nombreuses applications. Étant donné un besoin pré-défini, deux solutions sont habituellement disponibles pour collecter des données à partir de ces médias : 1) avoir accès à de grandes bases de données ou 2) utiliser les services de *streaming* que la plupart des médias proposent et qui permettent le suivi en temps réel de l'activité de leurs utilisateurs. Tandis que la première solution implique généralement des coûts très importants, la seconde constitue une alternative moins onéreuse. Toutefois, cela implique d'être en mesure de choisir efficacement quels flux de données sont les plus pertinents. Dans cet article, nous considérons des flux de données centrés sur les utilisateurs, chaque flux fournissant l'accès aux données publiées par un utilisateur particulier, avec des contraintes restrictives liées au nombre de flux pouvant être considérés simultanément (5000 sur Twitter). Étant donné le grand nombre d'utilisateurs (302 millions sur Twitter), le ciblage des sources pertinentes représente un effort non négligeable.

Imaginons une personne intéressée par l'informatique et souhaitant rester informée des dernières nouvelles relayées par les utilisateurs en relation avec ce sujet. Une solution consiste à sélectionner manuellement un ensemble de comptes et suivre leur activité au cours du temps. Cependant, Twitter est connu pour être extrêmement dynamique et pour une raison ou une autre, des utilisateurs pourraient se mettre à poster sur ce sujet tandis que d'autres pourraient cesser à tout moment. Donc, si l'utilisateur intéressé veut être à jour, il devrait changer le sous-ensemble des comptes suivis dynamiquement, son but étant d'anticiper les utilisateurs qui sont les plus susceptibles de produire des contenus pertinents dans un futur proche. Cette tâche semble toutefois difficile à exécuter manuellement pour deux raisons majeures. Tout d'abord, les critères choisis pour prédire si un compte sera potentiellement intéressant dans un futur proche peuvent être difficile à définir. Deuxièmement, même si nous étions en mesure de définir manuellement ces critères, la quantité de données à analyser serait trop grande pour pouvoir réaliser cette sélection dans un temps raisonnable.

Au vu de ces difficultés, nous proposons une solution qui, à chaque itération du processus, sélectionne automatiquement un sous-ensemble de comptes qui sont susceptibles d'être pertinents dans la fenêtre de temps suivante, en fonction de leur activité actuelle. Ces comptes sont ensuite suivis pendant un certain temps et les contenus publiés correspondants sont évalués - par l'utilisateur ou par un classifieur - afin de quantifier leur pertinence. L'algorithme derrière le système apprend alors une politique visant à améliorer la sélection à l'étape d'après. Nous abordons cette tâche comme un problème de bandit contextuel, dans lequel à chaque étape, l'algorithme choisit une action parmi un plus grand ensemble d'actions disponibles, sur la base des observations des caractéristiques de chaque action - le contexte de l'action - et reçoit une récompense qui quantifie la qualité de l'action choisie. Dans notre cas, étant

donné que suivre une personne correspond à une action, plusieurs actions peuvent être sélectionnées à chaque pas de temps, car nous voulons utiliser la capacité maximale de capture autorisée par l'API de *streaming*.

En définissant le contexte d'un utilisateur comme son activité présente, notre tâche s'inscrit bien le cadre du bandit contextuel. Cependant, dans son instance traditionnelle, le contexte de chaque action doit être observé à chaque itération afin d'effectuer les choix successifs et pour permettre l'apprentissage d'une politique. Dans notre cas, nous ne sommes pas en mesure d'observer l'activité présente de chaque utilisateur potentiel. Par conséquent, nous ne sommes pas en mesure d'obtenir tous les contextes et donc d'utiliser des algorithmes de bandits contextuels classiques tels que *LinUCB* (Chu *et al.*, 2011). D'autre part, nous ne pouvons pas traiter cela comme un problème de bandit traditionnel (non-contextuel) puisque nous perdrons les informations utiles fournies par les contextes observés. A notre connaissance, cette instance hybride du problème de bandit n'a pas encore été étudiée. Pour résoudre ce problème, nous proposons un algorithme de capture de données qui, en plus d'apprendre une politique de sélections des  $k$  meilleurs flux à chaque itération, apprend sur les distributions des contextes eux-mêmes, afin d'être en mesure de faire des estimations lorsque ces derniers ne sont pas observés.

## 2. État de l'art

Le problème du bandit manchot, proposé dans (Lai et Robbins, 1985) sous sa forme stationnaire a été largement étudié dans la littérature. Dans sa première instance, l'agent n'a accès à aucune information relative aux actions disponibles, mais fait l'hypothèse de distributions stationnaires sur les récompenses de chaque action. Un grand nombre de méthodes ont été proposées pour définir des politiques de sélection efficace, avec les garanties théoriques de convergence correspondantes. Le célèbre algorithme *UCB*, initialement proposé dans (Auer *et al.*, 2002), est basé sur la borne supérieure de l'intervalle de confiance des récompenses espérées pour chaque action disponible. Cet algorithme et ses très nombreuses variantes (Audibert *et al.*, 2007 ; Audibert et Bubeck, 2009) se sont révélés très efficaces pour la résolution du problème du bandit *stochastique*.

Dans (Chu *et al.*, 2011) entre autres, les auteurs étudient le problème du bandit contextuel, dans lequel l'agent observe des attributs sur chaque action avant d'effectuer la sélection, et proposent l'algorithme *LinUCB*. Cet algorithme fait l'hypothèse que la récompense espérée d'une action est une fonction linéaire de ses attributs et de paramètres inconnus à estimer. Que ce soit pour le bandit stochastique ou contextuel, d'autres types de stratégies appelées *Thompson Sampling* ont été proposés (Agrawal et Goyal, 2012a ; Chapelle et Li, 2011). Plus récemment, le cas où il est possible de sélectionner plusieurs actions simultanément a été formalisé dans (Chen *et al.*, 2013) (algorithme *CUCB*) et (Qin *et al.*, 2014) (algorithme  $C^2UCB$ ) respectivement pour le cas non-contextuel et le cas contextuel.

Concernant les médias sociaux, plusieurs tâches existantes présentent des similarités

avec la nôtre. Dans (Li *et al.*, 2013), les auteurs construisent une plate-forme appelée ATM qui vise à cibler automatiquement des tweets en rapport avec un certain sujet. Ils développent un algorithme qui sélectionne efficacement des mots-clés pour couvrir les tweets cibles. Dans notre cas, nous ne nous concentrons pas sur des mots clés, mais sur les profils des utilisateurs. Dans (Colbaugh et Glass, 2011), les auteurs modélisent la blogosphère comme un réseau où chaque nœud est considéré comme un flux de données pouvant être suivi. Leur objectif est d’identifier des blogs qui contiennent des contenus pertinents pour suivre les sujets émergents. Cependant, leur approche est statique et leur modèle est basé sur des données déjà collectées. Dans (Hannon *et al.*, 2010), les auteurs construisent un système de recommandation appelé *Twittomender*, utilisant des méthodes de filtrage collaboratif pour trouver les comptes Twitter susceptibles d’intéresser un utilisateur cible. Les auteurs supposent la connaissance du graphe des *followers/followees*, ce qui est impossible pour les applications à grande échelle en raison des politiques de restriction de Twitter.

Les algorithmes de bandit ont déjà été appliqués à diverses tâches liées aux réseaux sociaux. Par exemple, dans (Kohli *et al.*, 2013), les auteurs traitent des tâches de minimisation d’abandon dans les systèmes de recommandation. Dans (Buccapatnam *et al.*, 2014), les bandits sont utilisés pour la publicité en ligne tandis que dans (Lage *et al.*, 2013), les auteurs utilisent un bandit contextuel pour une tâche de maximisation de l’audience. Dans (Gisselbrecht *et al.*, 2015), une tâche de capture de données similaire est abordée via des algorithmes de bandits non contextuels. Dans ce cas, chaque utilisateur est supposé avoir une distribution stationnaire et le processus doit trouver des utilisateurs avec les meilleures moyennes en jouant sur l’exploitation des bonnes sources déjà connues et l’exploration des inconnues. Notre approche diffère de ce travail principalement par la considération d’une hypothèse de non stationnarité de l’utilité des différents utilisateurs. Nous considérons toutefois ce dernier dans nos expérimentations pour nous comparer.

### 3. La Sélection de Flux de Données

#### 3.1. Contexte

Notre objectif est de construire un système permettant de capturer des données pertinentes à partir des API proposées par les médias sociaux. Étant donné que les API limitent généralement le nombre d’utilisateurs pouvant être suivis de façon simultanée, l’objectif est de sélectionner dynamiquement ceux qui sont susceptibles de publier des contenus pertinents relativement à un besoin prédéfini. Il est important de noter que, quand bien même il n’y aurait pas de telles restrictions, notre approche reste valide vu l’énorme quantité de données publiées. La principale difficulté est donc de sélectionner efficacement les comptes à suivre étant donné le grand nombre d’utilisateurs et sachant qu’au début du processus, aucune information sur ces derniers n’est connue. En outre, même si certains comptes spécifiques pourraient être trouvés manuellement, le changement dynamique des sources semble complexe à réaliser ma-

nuellement. Pour ces raisons, la construction d’une solution automatique pour orienter le choix des sources apparaît pertinent.

### 3.2. Un Processus de Décision sous Contraintes

Notre tâche revient à sélectionner à chaque itération  $t$  du processus un sous-ensemble  $\mathcal{K}_t$  de  $k$  profils à suivre parmi l’ensemble de tous les utilisateurs  $\mathcal{K}$  ( $\mathcal{K}_t \subseteq \mathcal{K}$ ), selon leur propension à poster des tweets pertinents. Étant donné un score  $r_{a,t}$  associé au contenu produit par l’utilisateur  $a$  à l’étape  $t$ , le but est de sélectionner le sous-ensemble d’utilisateurs maximisant la somme des scores de pertinence :

$$\max_{(\mathcal{K}_t)_{t=1..n}} \sum_{t=1}^n \sum_{a \in \mathcal{K}_t} r_{a,t} \quad [1]$$

Notons que, contrairement à la majorité des tâches classiques en recherche information, nous ne nous préoccupons pas ici de la précision des informations collectées. L’objectif est de maximiser le volume de messages pertinents collectés, un filtrage pouvant être éventuellement appliqué a posteriori. Cet aspect diffère des tâches habituelles en recherche d’information. Les scores de pertinence considérés dépendent d’un besoin d’information, défini par l’utilisateur du système, pouvant prendre des formes variées. Par exemple, l’utilisateur pourrait souhaiter suivre des profils actifs sur des thématiques précises, ou bien des profils influents (au sens où leurs messages sont souvent repostés par d’autres utilisateurs). Ces scores peuvent soit être assignés manuellement, relativement à une évaluation humaine des contenus, ou bien automatiquement, par exemple à l’aide d’un classifieur (voir partie 5).

Dans (Gisselbrecht *et al.*, 2015) les auteurs se basent sur une hypothèse de stationnarité des scores associés à chaque utilisateur, ce qui paraît une hypothèse peu réaliste sur des réseaux sociaux généralistes tels que Twitter, où les utilisateurs peuvent changer de centre d’intérêt à tout moment. Nous supposons ici qu’il est possible de mieux anticiper le comportement futur des utilisateurs (i.e. les contenus futurs) en considérant leur activité présente (i.e. les contenus déjà postés). En d’autres termes, si l’on représente l’activité de l’utilisateur  $a$  au temps  $t - 1$  par un vecteur  $z_{a,t} \in \mathbb{R}^d$ , il existe une fonction  $h : \mathbb{R}^d \rightarrow \mathbb{R}$  permettant d’expliquer le score de  $a$  au temps  $t$ . Cette fonction de corrélation doit être apprise au fur et à mesure du processus de décision.

Cependant, la maximisation de la fonction définie dans la formule 1 est contrainte par différents facteurs :

- Les scores de pertinences  $r_{a,t}$  sont seulement définis pour les utilisateurs suivis à l’itération  $t$  (i.e. pour les  $a \in \mathcal{K}_t$ ) ;
- Les vecteurs de contextes  $z_{a,t}$  sont seulement observés pour un sous-ensemble  $\mathcal{O}_t$  des utilisateurs (i.e. il est impossible d’observer la totalité de l’activité du réseau) ;

Ces contraintes proviennent des restrictions liées aux API de *streaming* que nous utilisons pour la collecte des données :

– D’une part, une *API Sample streaming*, qui renvoie en temps réel 1% de tous les tweets publics. Nous utilisons cette API pour découvrir de nouveaux utilisateurs mais aussi pour récolter les contextes (activités) d’un grand nombre d’utilisateurs actifs à un moment donné.

– D’autre part, une *API Follow streaming*, qui permet d’obtenir en temps réel les contenus produits par 5000 utilisateurs définis. Nous utilisons cette API pour capturer les données produites par les utilisateurs sélectionnés par notre système.

Pour les profils appartenant à  $\mathcal{O}_t$ , les scores de pertinences peuvent être estimés grâce à la fonction de corrélation  $h$ , qui traduit les variations temporelles de la qualité d’un utilisateur. En revanche, pour les autres, nous proposons de considérer le cas stationnaire, en introduisant une tendance générale (ou qualité intrinsèque) pour chaque utilisateur. Nous sommes alors face à un problème hybride, où les contextes observés sont utilisés pour prédire des variations autour d’une utilité moyenne estimée.

### 3.3. Description du système

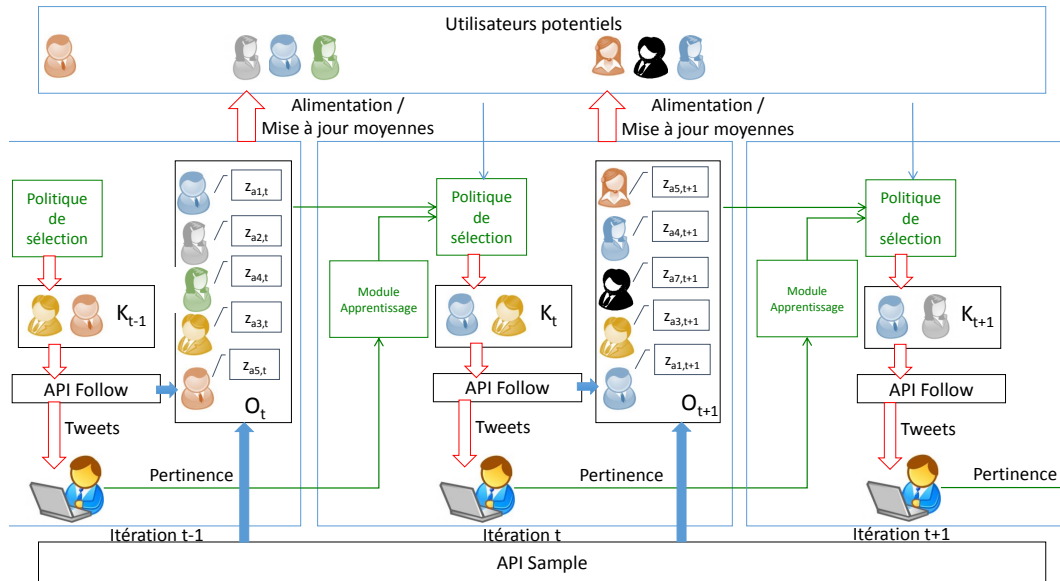


**Figure 1.** Illustration du système.

La figure 1 illustre le fonctionnement général du système. Ce dernier réitère les mêmes actions à chacune des itérations (trois itérations sont représentées). Au début de chaque étape, la politique de sélection des sources choisit un ensemble d'utilisateurs à suivre ( $K_t$ ) parmi tous les utilisateurs connus du système ( $\mathcal{K}$ ), selon des observations et des connaissances fournies par un module d'apprentissage. Ensuite, les messages postés par les  $k$  profils sélectionnés sont collectés via l'*API Follow streaming*. Comme nous pouvons le voir dans la partie centrale du schéma, après avoir suivi



les utilisateurs de  $\mathcal{K}_t$  pendant l'itération courante  $t$ , les messages collectés sont analysés (soit par un humain, soit automatiquement) et le résultat - traduisant la pertinence - est renvoyé au module d'apprentissage.

En parallèle, à chaque itération, l'activité courante de certains utilisateurs est capturée par l'API *Sample streaming*. Cela nous offre la possibilité d'enrichir la base des utilisateurs potentiels  $\mathcal{K}$ , mais aussi de construire l'ensemble des utilisateurs dont on connaît le contexte. Chaque profil ayant publié au moins un message parmi ceux qui ont été collecté via l'API *Sample streaming* à l'étape  $t$  sont inclus dans  $\mathcal{O}_{t+1}$ , les messages d'un même auteur  $a$  étant concaténés pour former  $z_{a,t+1}$  (voir la partie 5 pour des exemples de construction de vecteurs de contexte à partir des messages). À cet ensemble sont ajoutés les utilisateurs de  $\mathcal{K}_t$ , puisque leur activité complète a été suivie pendant l'étape  $t$ .

## 4. Modèle et Algorithme

### 4.1. Une approche de Bandit Contextuel

Dans le problème du bandit contextuel, l'hypothèse classique considère l'existence d'un vecteur inconnu  $\beta \in \mathbb{R}^d$  permettant d'estimer l'espérance de récompense que l'on peut observer pour chaque action et à chaque itération du processus :  $\forall t \in \{1, \dots, T\}, \forall a \in \mathcal{K} : \mathbb{E}[r_{a,t}|z_{a,t}] = z_{a,t}^T \beta$  (Agrawal et Goyal, 2012b). Dans notre cas, afin de modéliser la qualité intrinsèque de chaque action (une action correspondant à la sélection d'un utilisateur), nous introduisons également un terme de biais  $\theta_a \in \mathbb{R}$  pour chaque action  $a$ . Ainsi, sur les  $T$  itérations de notre processus de collecte :

$$\forall t \in \{1, \dots, T\}, \forall a \in \mathcal{K} : \mathbb{E}[r_{a,t}|z_{a,t}] = z_{a,t}^T \beta + \theta_a \quad [2]$$

Cette formulation correspond à un cas particulier de l'algorithme *LinUCB* hybride proposé dans (Li *et al.*, 2010) en considérant une valeur de 1 pour la valeur du contexte propre à chaque action. Il est important de noter que plusieurs paramètres individuels auraient pu être considérés dans notre étude, cependant cette approche ne nous semble pas adaptée étant donné le nombre très élevé d'actions disponibles (qui mènerait à un apprentissage d'autant plus complexe). D'autre part, l'utilisation d'un vecteur de paramètres commun permet une exploration facilitée, cela permettant la généralisation de corrélations observées pour des utilisateurs à l'ensemble des utilisateurs à l'activité courante proche. Ainsi, nous restreignons la modélisation individuelle à un terme unique de biais.

Comme précisé plus haut, notre tâche nécessite la sélection de multiples actions simultanément et donc à chaque itération  $t$ , l'algorithme doit sélectionner  $k$  ( $k < K$ ) utilisateurs parmi un ensemble de  $K$ , selon les contextes observés et les qualités intrinsèques de ces derniers. Le gain obtenu après une période d'écoute correspond donc à la somme des gains individuels. Dans notre cas, une difficulté majeure provient du fait que tous les contextes ne sont pas observables, contrairement aux problèmes de

bandits contextuels traditionnels. Dans la suite de ce travail, nous considérons pour chaque utilisateur une probabilité  $p_a$  ( $0 < p_a < 1$ ) de révéler son contexte. Notons que cette probabilité est contrainte par le problème considéré. Il s'agit alors de choisir à chaque itération  $t$  les  $k$  meilleurs utilisateurs à suivre parmi les  $K$  utilisateurs de  $\mathcal{K} = \mathcal{O}_t \cup \bar{\mathcal{O}}_t$ , avec  $\bar{\mathcal{O}}_t$  l'ensemble des utilisateurs pour lesquels on ne connaît pas le contexte pour l'étape  $t$ . La prochaine sous-section concerne le cas où l'on observe le contexte de tous les utilisateurs, la sous-section suivante étendant ce travail pour la prise en compte des utilisateurs de  $\bar{\mathcal{O}}_t$ . Enfin, cette section se termine par une description de l'algorithme de collecte global.

#### 4.2. Une approche contextuelle pour la collecte de données dynamique

Afin de dériver notre algorithme de collecte, nous considérons les hypothèses suivantes :

– **Vraisemblance** : Les scores de récompense sont identiquement et indépendamment distribués selon les contextes observés :  $r_{a,t} \sim \mathcal{N}(z_{a,t}^T \beta + \theta_a, \lambda_a)$ , avec  $\lambda_a$  correspondant à la variance de l'écart entre la récompense  $r_{a,t}$  et l'application linéaire de prédiction  $z_{a,t}^T \beta + \theta_a$  ;

– **Prior** : Les paramètres inconnus sont normalement distribués autour de l'origine :  $\beta \sim \mathcal{N}(0, b\mathbb{I}_d)$  et  $\theta_a \sim \mathcal{N}(0, \rho_a)$ , où  $b$  et  $\rho_a$  permettent de contrôler la variance des paramètres et  $\mathbb{I}_d$  est la matrice identité de taille  $d$  (taille des vecteurs de contexte).

L'idée est alors de construire un estimateur des paramètres  $\beta$  et  $\theta$  par maximum a posteriori connaissant les récompenses collectées, et les contextes associés, jusqu'à l'itération courante. Pour simplifier, nous fixons  $b$ ,  $\rho_a$  et  $\lambda_a$  à 1 pour tout  $a$ . Cependant, tous les résultats peuvent être étendus à des cas plus complexes.

**Proposition 1** *En notant  $T_a$  l'ensemble des itérations où  $a$  a été choisi durant les  $n$  premières étapes ( $T_a = \{t \leq n, a \in \mathcal{K}_t\}$ ,  $|T_a| = \tau_a$ ),  $c_a$  le vecteur de récompenses obtenues par  $a$  lorsqu'il a été écouté ( $c_a = (r_{a,t})_{t \in T_a}$ ) et  $D_a$  la matrice des contextes associée à  $a$  ( $D_a = (z_{a,t}^T)_{t \in T_a}$ ), la distribution a posteriori de  $\beta$  et  $\theta_a$  après  $n$  étapes, lorsque tous les contextes sont disponibles, est donnée par<sup>1</sup> :*

$$\beta \sim \mathcal{N}(\bar{\beta}, A_0^{-1}) \quad [3] \quad \theta_a + \bar{z}_a^T \beta \sim \mathcal{N}\left(\bar{\mu}_a, \frac{1}{\tau_a + 1}\right) \quad [4]$$

$$\text{Avec : } A_0 = \mathbb{I}_d + \sum_{a=1}^K (\tau_a + 1) \bar{\Sigma}_a ; b_0^T = \sum_{a=1}^K (\tau_a + 1) \bar{\xi}_a ; \bar{\Sigma}_a = \frac{D_a^T D_a}{\tau_a + 1} - \bar{z}_a \bar{z}_a^T$$

$$\bar{\xi}_a = \frac{c_a^T D_a}{\tau_a + 1} - \bar{\mu}_a \bar{z}_a^T ; \bar{\beta} = A_0^{-1} b_0^T ; \bar{\mu}_a = \frac{\sum_{t \in T_a} r_{a,t}}{\tau_a + 1} ; \bar{z}_a = \frac{\sum_{t \in T_a} z_{a,t}}{\tau_a + 1}$$

<sup>1</sup>Les preuves complètes des propositions sont disponibles à l'URL <http://www-connex.lip6.fr/~lampriers/CORIA2016-supplementaryMaterial.pdf>

Notons que tous ces paramètres possèdent une dépendance en  $t$ , que nous avons volontairement omise afin d'alléger les notations. De plus tous ces paramètres peuvent être mis à jour de façon peu coûteuse à mesure que de nouveaux exemples d'apprentissage arrivent (la complexité de la mise à jour des paramètres est constante sur l'ensemble du processus). En combinant les équations 2, 3 et 4, la valeur espérée  $\mathbb{E}[r_{a,t}|z_{a,t}]$ , de la récompense de l'utilisateur  $a$  à l'itération  $t$  sachant son contexte  $z_{a,t}$ , suit la distribution postérieure :

$$\mathcal{N}\left(\bar{\mu}_a + (z_{a,t} - \bar{z}_a)^T \bar{\beta}, \frac{1}{\tau_a + 1} + (z_{a,t} - \bar{z}_a)^T A_0^{-1} (z_{a,t} - \bar{z}_a)\right) \quad [5]$$

**Theorème 1** Pour tout  $0 < \delta < 1$  et  $z_{a,t} \in \mathbb{R}^d$ , en notant  $\alpha = \sqrt{2} \operatorname{erf}^{-1}(1 - \delta)^2$ , pour chaque action  $a$ , après  $t$  itérations<sup>1</sup> :

$$P\left(|\mathbb{E}[r_{a,t}|z_{a,t}] - \bar{\mu}_a - (z_{a,t} - \bar{z}_a)^T \bar{\beta}| \leq \alpha \sigma_a\right) \geq (1 - \delta)$$

$$\text{avec } \sigma_a = \sqrt{\frac{1}{\tau_a + 1} + (z_{a,t} - \bar{z}_a)^T A_0^{-1} (z_{a,t} - \bar{z}_a)} \quad [6]$$

Cette formule peut directement être utilisée pour définir une borne supérieure de l'intervalle de confiance (*UCB - Upper Confidence Bound*) de la récompense espérée pour chaque utilisateur dont le contexte a été observé pour l'itération  $t$  :

$$s_{a,t} = \bar{\mu}_a + (z_{a,t} - \bar{z}_a)^T \bar{\beta} + \alpha \sigma_a \quad [7]$$

De manière classique, les approches type *UCB* étant des approches optimistes, l'idée est alors de sélectionner à chaque instant  $t$  les  $k$  utilisateurs ayant les  $k$  bornes  $s_{a,t}$  les plus élevées. Nous rappelons qu'une formulation similaire est utilisée dans le bandit contextuel traditionnel. Dans la partie suivante, nous proposons une méthode permettant d'adapter cette dernière au cas où l'on ne peut pas observer tous les contextes.

### 4.3. Prise en compte des utilisateurs avec contexte caché

Les scores permettant de sélectionner les utilisateurs à écouter définis précédemment ne peuvent pas être calculés lorsque les contextes  $z_{a,t}$  ne sont pas observés (i.e. lorsque les utilisateurs ne sont pas dans  $\mathcal{O}_t$ ). Cela nécessite donc une méthode permettant de qualifier les personnes dont le contexte est inconnu. De plus, cela implique des problématiques de mise à jour des paramètres appris lorsqu'un utilisateur a été écouté mais dont le contexte n'a pas été révélé. Bien qu'il soit tentant de penser que le paramètre  $\beta$  commun à tous et les paramètres individuels  $\theta_a$  peuvent être appris séparément, ces derniers sont en réalité corrélés, comme le montrent les formules 3

<sup>1</sup> $\operatorname{erf}^{-1}$  est la fonction erreur inverse,  $\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$ .

et 4. Il est important de remarquer que pour conserver les garanties probabilistes, les paramètres du problème ne devraient être mis à jour que lorsqu'un profil appartient à  $\mathcal{K}_t \cap \mathcal{O}_t$ . Remplacer  $T_a$  par  $T_a^{both} = \{t \leq n, a \in \mathcal{K}_t \cap \mathcal{O}_t\}$  (nous conservons la notation  $|T_a^{both}| = \tau_a$ ), nous permet de réutiliser les formules de mise à jour des paramètres de la Proposition 1. Toutefois, le calcul du score de sélection d'un utilisateur dont le contexte est caché ne peut être effectué grâce à l'équation 7. Pour palier à ce problème, nous proposons d'utiliser un estimateur de la distribution moyenne du vecteur de contexte de chaque profil.

#### Nouvelles notations :

- L'ensemble des itérations où le contexte de  $a$  a été observé au bout de  $n$  étapes est noté  $T_a^{obs} = \{t \leq n, a \in \mathcal{O}_t\}$  avec  $|T_a^{obs}| = n_a$ .

- La moyenne empirique des vecteurs de contexte de l'utilisateur  $a$  est noté  $\hat{z}_a$ , avec  $\hat{z}_a = 1/n_a \sum_{t \in T_a^{obs}} z_{a,t}$ . Remarquons que  $\hat{z}_a$  est différent de  $\bar{z}_a$  car le premier est mis à jour à chaque fois que le contexte est observé tandis que le second l'est uniquement lorsque l'utilisateur a été observé et sélectionné lors d'une même itération.

#### Hypothèses supplémentaires :

- Sans perte de généralité, nous supposons que  $\beta$  est borné par  $M \in \mathbb{R}^{+*}$ , i.e.  $\|\beta\| \leq M$ , où  $\|\cdot\|$  désigne la norme  $L^2$  sur  $\mathbb{R}^d$ .
- Pour tout  $a$  et à chaque  $t$ , les contextes  $z_{a,t} \in \mathbb{R}^d$  sont iid depuis une distribution inconnue de moyenne  $\mathbb{E}[z_a]$  et variance  $\Sigma_a$  finies.

**Theorème 2** *Étant donné  $0 < \delta < 1$  et  $0 < \gamma < \frac{1}{2}$ , en notant  $\alpha = \sqrt{2}erf^{-1}(1 - \delta)$ , pour tout  $a$  après  $t$  itérations du processus<sup>1</sup> :*

$$P\left(\left|\mathbb{E}[r_a] - \bar{\mu}_a - (\hat{z}_a - \bar{z}_a)^T \bar{\beta}\right| \leq \alpha \hat{\sigma}_a + \frac{1}{n_a^\gamma}\right) \geq (1 - \delta) \left(1 - \frac{C_a}{n_a^{1-2\gamma}}\right)$$

$$\text{avec } \hat{\sigma}_a = \sqrt{\frac{1}{\tau_a + 1} + (\hat{z}_a - \bar{z}_a)^T A_0^{-1} (\hat{z}_a - \bar{z}_a)} \quad [8]$$

Où  $C_a$  est une constante positive spécifique à chaque utilisateur.

Étant donné que  $\gamma < \frac{1}{2}$ , la probabilité précédente tend vers  $1 - \delta$  à mesure que le nombre d'observations du contexte de l'utilisateur  $a$  augmente (comme dans l'équation 6). Ainsi, l'inégalité ci-dessus nous donne une borne relativement serrée de l'intervalle de confiance associé à la récompense espérée de l'utilisateur  $a$ , d'où nous pouvons dériver un nouveau score de sélection. À chaque itération  $t$ , si le contexte de  $a$  n'est pas observé :

$$s_{a,t} = \bar{\mu}_a + (\hat{z}_a - \bar{z}_a)^T \bar{\beta} + \alpha \hat{\sigma}_a + \frac{1}{n_a^\gamma} \quad [9]$$

Notons que puisque  $\gamma > 0$ , le terme d'exploration supplémentaire  $1/n_a^\gamma$  tend vers 0 à mesure que le nombre d'observations de  $a$  augmente, et le score tend donc vers un score *LinUCB* classique (comme dans l'équation 7) dans lequel le contexte est remplacé par sa moyenne empirique  $\hat{z}_a$ . Finalement, comme chaque utilisateur a une probabilité  $0 < p_a < 1$  de révéler son contexte à chaque temps  $t$ , nous pouvons affirmer que  $1/n_a^\gamma$  tend vers 0 lorsque  $t$  augmente.

#### 4.4. Algorithme de Capture de Données Contextuel

Les ensembles et paramètres manipulés sont tout d'abord initialisés :

- $A_0$  par une matrice identité de taille  $d \times d$  et  $b_0$  par un vecteur nul de taille  $d$ , avec  $d$  la taille des vecteurs de contexte ;
- $\mathcal{K}$  par l'ensemble vide ;
- $\mathcal{O}_1$  par les utilisateurs auteurs de messages collectés au cours d'une première utilisation de l'API *Sample* pendant une période de temps d'une durée  $\mathcal{L}$ . La représentation des messages de ces utilisateurs  $a \in \mathcal{O}_1$  correspondent à leur contexte  $z_{a,1}$ .

L'algorithme procède ensuite, pour chaque itération  $t$  de la collecte<sup>3</sup>, selon :

- 1) Insertion dans  $\mathcal{K}$  de chaque utilisateur de  $\mathcal{O}_t$  n'en faisant pas encore partie (en limitant le nombre de nouveau entrants à *newMax* utilisateurs par itération afin d'éviter une sur-exploration pour le cas des très grands réseaux tels que Twitter) ;
- 2) Mise à jour des moyennes empiriques des contextes  $\hat{z}_a$  pour les profils de  $\mathcal{O}_t$  ;
- 3) Calcul de l'estimateur du paramètre  $\hat{\beta}$  (voir proposition 1) ;
- 4) Calcul des scores de sélection  $s_{a,t}$  de tous les utilisateurs dans  $\mathcal{K}$ , selon la formule 7 pour les utilisateurs de  $\mathcal{O}_t$  et selon la formule 9 ceux de  $\bar{\mathcal{O}}_t$ . Le score des nouveaux utilisateurs est fixé à  $+\infty$  pour forcer le système à les choisir une première fois ;
- 5) Sélection des  $k$  profils ayant les meilleurs scores  $s_{a,t}$  ;
- 6) Capture simultanée via les APIs *Sample* et *Follow* pendant une période de durée  $\mathcal{L}$ . L'API *Follow* écoute les  $k$  utilisateurs sélectionnés  $\mathcal{K}_t$  ;
- 7) Calcul des récompenses des messages collectés pour les utilisateurs de  $\mathcal{K}_t$  à partir de l'API *Follow*, puis mise à jour des paramètres du modèle ;
- 8) Mise à jour de l'ensemble des utilisateurs observés  $\mathcal{O}_{t+1}$ , et des contextes associés, en considérant tous les utilisateurs pour lesquels au moins un message a été collecté pendant l'itération  $t$  (provenant soit de l'API *Sample* soit de l'API *Follow*) ;

<sup>3</sup>L'algorithme complet est disponible à l'URL <http://www-connex.lip6.fr/~lampriers/CORIA2016-supplementaryMaterial.pdf>

## 5. Expérimentations

### 5.1. Description générale

Outre une politique (appelée *Random*) sélectionnant uniformément les utilisateurs à suivre à chaque itération, nous comparons notre approche de collecte à deux autres algorithmes de bandits : *CUCB* et *CUCBV* respectivement proposés dans (Qin *et al.*, 2014) et (Gisselbrecht *et al.*, 2015). Ces algorithmes ne prennent pas en compte les contextes des actions. Dans toutes nos expérimentations, nous fixons les paramètres suivants : 1) Le terme d’exploration à  $\alpha = 1.96$ , ce qui correspond à un intervalle de confiance à 95% pour l’estimation de l’espérance de la récompense d’une action lorsque son contexte est observé ; 2) Le paramètre de réduction de l’intervalle de confiance des contextes inconnus à  $\gamma = 0.25$  ; 3) Le nombre d’utilisateurs pouvant être ajoutés à  $\mathcal{K}$  à chaque itération à  $newMax = 200$ .

### 5.2. Expérimentations hors ligne

Deux jeux de données sont utilisés pour nos expérimentations hors ligne :

1) *USElections* : Jeu de données de 2148651 messages collectés pendant les dix jours précédents les élections présidentielles américaines de 2012. Ces messages correspondent à la totalité de ceux postés pendant cette période par les 5000 premiers utilisateurs à avoir utilisé les mots “Obama”, “Romney” ou “#USElections”.

2) *Libya* : Jeu de données de 1211475 messages provenant de 17341 utilisateurs. Ces messages correspondent à l’ensemble de ceux postés sur une période de trois mois contenant le mot clé “Libya”.

**Modèle de contexte :** Dans nos expérimentations, nous considérons que le contexte  $z_{a,t}$  d’un utilisateur  $a$  pour une itération  $t$  correspond à une représentation de la concaténation des messages collectés pour cet utilisateur pendant l’étape  $t - 1$  selon l’une des deux APIs. Plutôt que d’utiliser directement une représentation fréquentielle en sac de mots, ce qui impliquerait de très longs vecteurs de contextes et un coût important lié à l’inversion des matrices requise lors de la mise à jour des paramètres, nous appliquons une réduction de dimensions à un espace de 30 concepts par l’emploi de l’algorithme LDA (Blei *et al.*, 2003). Outre une réduction de la complexité, l’utilisation de cet algorithme permet une meilleure généralisation des observations en regroupant sous un même concept des mots sémantiquement proches.

**Modèle de récompense :** Dans nos expérimentations, nous nous intéressons à des modèles de collecte visant à récompenser les messages ayant un fort impact sur une thématique donnée. Quatre thématiques sont considérées (correspondant à 4 différents modèles de récompense) : *politique*, *religion*, *sport* et *science*. L’appartenance à l’une de ces 4 thématiques se fait par l’application d’un classifieur linéaire entraîné sur le corpus *20 Newsgroups*. Ainsi, si un message correspond à la classe de la thématique recherchée, le score de récompense qui y est associé est alors égal au nombre de fois

où ce message est re-posté (*retweeté* sur Twitter) par d'autres utilisateurs pendant l'itération courante.

Afin d'obtenir des résultats généralisables, nous avons expérimenté plusieurs configurations de collecte en considérant plusieurs valeurs pour  $p$ , la probabilité d'observation des contextes,  $k$ , le nombre d'utilisateurs pouvant être suivis simultanément, et  $\mathcal{L}$ , la durée d'une itération. Plus précisément nous avons testé toutes les combinaisons possibles avec  $p \in \{0.1, 0.5, 1.0\}$ ,  $k \in \{50, 100, 150\}$  et  $\mathcal{L} \in \{2min, 3min, 6min, 10min\}$ . Néanmoins, pour des raisons de places, nous reportons seulement les résultats pour  $k = 100$ ,  $\mathcal{L} = 2min$ . De la même façon, pour le jeu de données *USElections*, seule les expérimentations considérant le modèle de récompense centré sur la classe *politique* sont reportées, tandis que pour le jeu de données *Libya*, seul celui centré sur la classe *religion* l'est. Des tendances similaires ont été observées pour les autres configurations et modèles de récompense.

**Résultats :** La figure 2 représente le gain cumulé en fonction du temps, en haut à gauche pour *USElections* et en haut à droite pour *Libya*. Tout d'abord, nous remarquons que chaque politique fonctionne mieux que la politique *Random*, ce qui est un premier élément pour affirmer la pertinence des algorithmes de bandit pour la tâche en question. Deuxièmement, nous remarquons que *CUCBV* fonctionne mieux que *CUCB*, ce qui confirme les résultats obtenus dans (Gisselbrecht *et al.*, 2015) sur une tâche similaire de capture de données ciblée. Plus intéressant, lorsque chaque contexte est observable ( $p = 1$ ), notre algorithme contextuel fonctionne mieux que les approches *CUCB* et *CUCBV*. Ce résultat montre que nous sommes en mesure de mieux anticiper les utilisateurs qui vont être les plus pertinents à l'étape suivante, compte tenu de ce qu'ils ont dit juste avant. Cela confirme aussi le comportement non-stationnaire des utilisateurs. Par exemple, les utilisateurs peuvent parler de science au cours de la journée tout en étant plus axé sur le sport quand ils reviennent à la maison. Considérer les contextes permet également de converger plus rapidement vers les utilisateurs intéressants puisque tous les comptes partagent le même paramètre  $\beta$ . En outre, les résultats montrent que, même pour de faibles probabilités d'observation des contextes  $p$ , notre politique contextuelle se comporte beaucoup mieux que les approches non-contextuelles, ce qui valide empiriquement notre approche : il est possible de tirer parti de l'information contextuelle, même si une grande partie de cette information est cachée. En particulier, pour le jeu de données *Libya* où aucune différence significative entre les deux algorithmes *CUCB* et *CUCBV* n'est remarquable, notre algorithme semble plus approprié. En donnant la possibilité de sélectionner des utilisateurs même si leur contexte est inconnu, nous permettons à l'algorithme de compléter sa sélection en choisissant les utilisateurs dont le contexte moyen correspond à un profil appris. Si aucun utilisateur dans  $\mathcal{O}_t$  ne semble actuellement pertinent, l'algorithme peut alors compter sur les utilisateurs avec une bonne qualité intrinsèque moyenne. Numériquement pour  $k = 100$ , le nombre moyen d'utilisateurs sélectionnés pour lesquels le contexte a été observé à chaque pas de temps est de 43 pour  $p = 0.1$  et 58 pour  $p = 0.5$ , ce qui confirme que l'utilisation de contextes intéressants lorsque ceux-ci sont disponibles, tout en conservant une probabilité de sélection non négligeable pour les utilisateurs de qualité dont le contexte n'est pas connu.

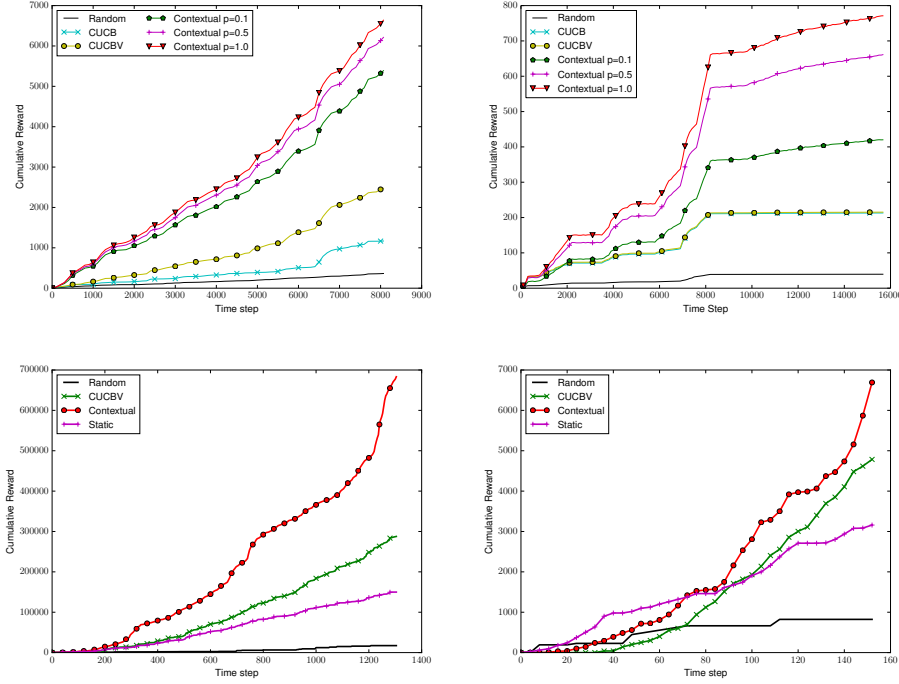


**Figure 2.** Récompense cumulée en fonction du temps. Haut gauche : USElections/politique. Haut droite : Libya/religion. Bas gauche : Expérience en ligne. Bas droite : Expérience en ligne avec un zoom sur les premiers pas de temps.

### 5.3. Expérience en ligne

Nous considérons notre tâche de collecte dans une expérimentation en ligne sur Twitter prenant en compte l'ensemble du réseau social ainsi que les contraintes de l'API. Pour ces expériences, nous avons utilisé tout le potentiel de l'API Twitter, à savoir 1% de tous les tweets publics pour l'API *Sample*, qui s'exécute en arrière-plan, et 5000 utilisateurs suivis simultanément ( $k = 5000$ ) pour l'API *Follow*. À chaque itération, les utilisateurs sélectionnés sont suivis pendant  $\mathcal{L} = 15minutes$ . Compte tenu des restrictions de Twitter, chaque expérience nécessite un compte développeur, ce qui limite le nombre de politiques que nous pouvons tester en parallèle. Nous avons choisi de tester les quatre stratégies suivantes : notre approche contextuelle, *CUCBV*, *Random* et une autre appelée *Static*, qui suit les mêmes 5000 comptes, sélectionnés a priori, à chaque étape du processus. Les 5000 comptes suivis par l'approche *Static* ont été choisis en sélectionnant les 5000 utilisateurs ayant cumulé le plus fort volume de récompenses selon les messages collectés par l'API *Sample* sur une période de 24 heures. Pour information, certains comptes célèbres tels que *@dailytelegraph*, *@Independent* ou *@CNBC* en faisaient partie.

**Résultats :** La partie inférieure gauche de la figure 2 représente l'évolution du gain cumulé sur deux semaines pour les quatre algorithmes. On remarque le très bon com-



portement de notre algorithme dans un scénario réel, car la somme des récompenses qu’il accumule croît beaucoup plus rapidement que celle des autres politiques, surtout après les 500 premières itérations. Après ces premières itérations, notre algorithme semble avoir acquis une bonne connaissance sur les distributions de récompenses en fonction des contextes observés sur les différents utilisateurs du réseau. Afin d’analyser le comportement des politiques expérimentées pendant les premières itérations de la capture, nous représentons aussi en bas à droite de la figure 2 un zoom des mêmes courbes sur les 150 premiers pas de temps. Au début, la politique *Static* est plus performante que toutes les autres, ce qui peut s’expliquer par deux raisons : 1) les algorithmes de bandits utilisés ont besoin de sélectionner tous les utilisateurs au moins une fois pour initialiser les scores et 2) les utilisateurs faisant partie de l’ensemble *Static* sont supposés être des sources relativement pertinentes étant donné la façon dont ils ont été choisis. Vers l’itération 80, aussi bien l’algorithme *CUCBV* que l’algorithme contextuel deviennent meilleurs que *Static*, ce qui correspond au moment où ils commencent à pouvoir exploiter des “bonnes” actions identifiées. Ensuite, une période d’environ 60 itérations est observée (approximativement entre les itérations 80 et 140), au cours de laquelle l’algorithme *CUCBV* et notre approche contextuelle ont des performances comparables. Cette période s’explique par le fait que notre approche nécessite un certain nombre d’itérations pour apprendre des corrélations efficaces entre contextes et récompenses afin d’en tirer avantage. Enfin, après cette période d’initialisation, on peut remarquer un changement significatif de pente pour la courbe correspondant à notre algorithme, ce qui souligne sa faculté à être bien plus réactif dans un environnement dynamique.

## 6. Conclusion

Nous avons formalisé une tâche de collecte de données sur les réseaux sociaux comme une instance spécifique du problème de bandit contextuel, dans laquelle, dû aux restrictions imposées par les médias sociaux, seule une partie des contextes est observable à chaque itération. Pour résoudre cette tâche, nous avons proposé une adaptation de l’algorithme de bandit contextuel *LinUCB* pour le cas où certains contextes sont cachés. Bien qu’aucune borne supérieure sous-linéaire sur le regret ne puisse être atteinte, en raison de la grande incertitude induite par les contextes cachés, notre algorithme offre de bons résultats pour la collecte de données, même lorsque la proportion de contextes cachés est élevée. Cela permet en outre d’envisager son application à bien d’autres tâches, pour lesquelles des contraintes restreignent l’observation des contextes associés aux actions disponibles.

## 7. Remerciements

Ce travail a été effectué dans le cadre des recherches menées au sein de l’IRT SystemX et a ainsi bénéficié d’une aide de l’Etat au titre du programme d’Investissements

d'Avenir, ainsi que dans le cadre du projet LUXIDX du dispositif RAPID du fond de compétitivité des entreprises.

## 8. Bibliographie

- Agrawal S., Goyal N., « Analysis of Thompson Sampling for the multi-armed bandit problem », *COLT*, 2012a.
- Agrawal S., Goyal N., « Thompson Sampling for Contextual Bandits with Linear Payoffs », *CoRR*, 2012b.
- Audibert J.-Y., Bubeck S., « Minimax policies for adversarial and stochastic bandits », *COLT*, 2009.
- Audibert J.-Y., Munos R., Szepesvari C., « Tuning bandit algorithms in stochastic environments », *ALT*, 2007.
- Auer P., Cesa-Bianchi N., Fischer P., « Finite-time Analysis of the Multiarmed Bandit Problem », *Mach. Learn.*, 2002.
- Blei D. M., Ng A. Y., Jordan M. I., « Latent Dirichlet Allocation », *J. Mach. Learn. Res.*, 2003.
- Buccapatnam S., Eryilmaz A., Shroff N. B., « Stochastic Bandits with Side Observations on Networks », *SIGMETRICS*, 2014.
- Chapelle O., Li L., « An Empirical Evaluation of Thompson Sampling », *NIPS*, 2011.
- Chen W., Wang Y., Yuan Y., « Combinatorial Multi-Armed Bandit : General Framework and Applications », *ICML, JMLR Workshop and Conference Proceedings*, 2013.
- Chu W., Li L., Reyzin L., Schapire R. E., « Contextual Bandits with Linear Payoff Functions », *AISTATS*, 2011.
- Colbaugh R., Glass K., « Emerging Topic Detection for Business Intelligence Via Predictive Analysis of 'Meme' Dynamics », *AAAI Spring Symposium : AI for Business Agility*, 2011.
- Gisselbrecht T., Denoyer L., Gallinari P., Lamprier S., « WhichStreams : A Dynamic Approach for Focused Data Capture from Large Social Media », *ICWSM*, 2015.
- Hannon J., Bennett M., Smyth B., « Recommending Twitter Users to Follow Using Content and Collaborative Filtering Approaches », *RecSys*, 2010.
- Kohli P., Salek M., Stoddard G., « A Fast Bandit Algorithm for Recommendation to Users With Heterogenous Tastes. », *AAAI*, 2013.
- Lage R., Denoyer L., Gallinari P., Dolog P., « Choosing Which Message to Publish on Social Networks : A Contextual Bandit Approach », *ASONAM*, 2013.
- Lai T., Robbins H., « Asymptotically efficient adaptive allocation rules », *Advances in Applied Mathematics*, vol. 6, n° 1, p. 4 - 22, 1985.
- Li L., Chu W., Langford J., Schapire R. E., « A Contextual-bandit Approach to Personalized News Article Recommendation », *WWW '10, WWW '10*, p. 661-670, 2010.
- Li R., Wang S., Chang K. C.-C., « Towards Social Data Platform : Automatic Topic-focused Monitor for Twitter Stream », *Proc. VLDB Endow.*, 2013.
- Qin L., Chen S., Zhu X., « Contextual Combinatorial Bandit and its Application on Diversified Online Recommendation », *SIAM*, 2014.

---

# Recommandation et analyse de sentiments dans un espace latent textuel

**Charles-Emmanuel Dias\*** — **Vincent Guigue\*** — **Patrick Gallinari\***

*\*Sorbonne Universités, UPMC Univ Paris 06, UMR 7606, LIP6, F-75005, Paris, France – Mail: Prénom.Nom@lip6.fr*

---

*RÉSUMÉ. Les systèmes de recommandation permettent d'aider les utilisateurs à identifier les contenus qu'ils seraient susceptibles d'apprécier dans des catalogues en expansion constante. Les méthodes traditionnelles de filtrage collaboratif se focalisent majoritairement sur les notes que les gens laissent en ligne pour établir leurs profils et ignorent les commentaires textuels éventuellement joints. Nous prenons le parti d'exploiter les avis textuels comme source principale d'information et proposons, en plus de prédire les notes, de prédire les remarques des utilisateurs. Nous montrons aussi comment cette approche peut offrir de nouvelles perspectives pour le problème du démarrage à froid. À l'aide d'une technique récente de représentation de textes, nous construisons un espace latent d'avis où chacun de leurs aspects est représenté. Tous les acteurs du système sont représentés sous forme de vecteurs et une métrique ad hoc nous permet d'attaquer différentes problématiques autour de la recommandation.*

*ABSTRACT. Recommender systems were developed to cherry-pick interesting content in an always growing environment to help users discover new products or information that they might like. Traditional collaborative filtering methods mainly focus on the ratings to establish user profiles, ignoring the joint review text. Our work does the opposite. We use the written comments as the primary source of information and propose, in addition to rating forecast, to predict review text. Our approach also enables us to elegantly tackle the cold-start problem. Using state-of-the-art text embedding technique, we build a latent review space where all the components of a review is mapped. The created vectors are used to predict both rating and review text.*

*MOTS-CLÉS : Recommandation, Analyse de sentiments, Apprentissage de représentations textuelles*

*KEYWORDS: Recommender systems, Text embedding, Sentiment classification.*

---

## 1. Introduction

Afin d'aider les utilisateurs à accéder à des ressources en ligne toujours plus foisonnantes, des systèmes de recommandation ont été développés pour sélectionner et trier les contenus susceptibles de leur plaire. Ils permettent, par exemple, de composer des newsletters ou de fournir des conseils individualisés sur les produits à acheter sur Amazon, les restaurants où dîner grâce à TripAdvisor et même les films à regarder sur Netflix. L'enjeu est de proposer un véritable accès personnalisé à l'information.

Pour atteindre cet objectif, il faut pouvoir fournir des recommandations justes. Mais, il est tout aussi important d'expliquer en quoi celles-ci sont appropriées (Tintarev et Masthoff, 2007). Les algorithmes de recommandation ont franchi un premier cap d'efficacité et de popularité avec le challenge Netflix (Bennett et Lanning, 2007) qui a consacré le filtrage collaboratif (Koren *et al.*, 2009). Ces dernières années, ils sont devenus de plus en plus performants grâce à la prise en compte d'un nombre croissant de facteurs comme le temps (Koren, 2009 ; McAuley et Leskovec, 2013a), les liens sociaux (Guy et Carmel, 2011) ou le texte (McAuley et Leskovec, 2013b). Pourtant, ces améliorations n'ont pas contribué à l'explicitation des propositions faites aux utilisateurs et les méthodes récentes de filtrage collaboratif utilisant la factorisation matricielle sont encore souvent qualifiées de boîtes noires (Ben-elazar et Koenigstein, 2014). Il reste donc des progrès à faire pour permettre à l'utilisateur d'interpréter les conseils de ces algorithmes.

Dans ce papier, nous avons choisi de travailler sur les revues en ligne. Celles-ci, laissées par chacun sur internet, renferment de précieuses informations sur les avantages et les inconvénients des éléments notés. Leur exploitation peut permettre de connaître les préférences d'un utilisateur de façon plus fine que les notes, qui peuvent être trompeuses (Gauthier *et al.*, 2015). Différents travaux explorent l'utilisation du texte en recommandation, majoritairement avec comme but d'améliorer la qualité prédiction plutôt que celle des explications (McAuley et Leskovec, 2013b ; Ling *et al.*, 2014). (Poussevin *et al.*, 2014) a proposé une nouvelle tâche de prédiction des remarques utilisateurs qui permet justement d'apporter éclaircissement quant aux suggestions faites.

Depuis le lancement de la compétition Netflix en 2006, on observe un consensus sur les données qui servent à la recommandation: des triplets (utilisateur, produit, note). Le texte n'est utilisé qu'à la marge pour affiner l'espace de représentation et donc la recommandation. Nous proposons dans cet article d'évaluer les similarités entre acteurs (utilisateur, produit...) dans un espace latent de mots et de nous baser uniquement sur le texte disponible pour la création des profils latents. Grâce à une technique récente de représentation de texte, nous construisons un espace latent d'avis en ligne entièrement basé sur le texte (Mikolov *et al.*, 2013 ; Le et Mikolov, 2014). Les similarités découlant de cet espace sont ensuite utilisées pour prédire à la fois les notes et les remarques des utilisateurs avec précision. Nous démontrons dans cet article la pertinence de l'espace sémantique comme support de projection pour les utilisateurs et les produits dans le cadre de la tâche de prédiction de note. De plus, notre démarche

apporte une solution naturelle et élégante au problème du démarrage à froid (Contardo *et al.*, 2014). Alors que les systèmes de factorisation matricielle ne peuvent pas modéliser un utilisateur ou un produit sans historique, une simple description textuelle nous permet de projeter un nouvel élément dans l’espace latent. Au bout du processus, tous les utilisateurs, produits ou éléments du système de recommandation sont positionnés dans un espace sémantique de mots, de phrases et de revues. Il est donc possible d’expliquer chaque recommandation faite à travers des mots-clés, des phrases et même des revues à la manière de (Poussevin *et al.*, 2014).

Notre contribution est d’utiliser l’information textuelle comme support principal d’un système de recommandation afin de créer des profils d’utilisateurs et de produits plus riches. D’améliorer et, surtout, d’expliquer les recommandations faites.

Ce papier est organisé de la façon suivante: tout d’abord, nous exposons le contexte bibliographique de nos travaux (section 2). Ensuite, nous expliquons comment construire notre espace latent de recommandation et comment nous en servir (section 3). Enfin, nous exposons les résultats obtenus (section 4).

## 2. État de l’art

Dans cette section, nous décrivons d’abord la tâche de prédiction des notes à travers le filtrage collaboratif et ses principaux modèles. Ensuite, nous présentons les approches de la littérature pour exploiter les représentations textuelles.

### 2.1. Filtrage collaboratif

En recommandation, le filtrage collaboratif est une méthode de prédiction personnalisée qui se base sur l’historique des notes données par les utilisateurs. L’hypothèse sous-jacente est que si plusieurs personnes ont des intérêts communs pour certaines choses alors ces intérêts s’étendent probablement à d’autres produits. En filtrage collaboratif, la factorisation matricielle draine l’attention de la plus grande partie de la communauté scientifique. Nous discutons aussi des techniques à base de voisinage sur lesquelles repose notre exploitation des espaces latents de texte pour la recommandation.

#### 2.1.1. Factorisation matricielle

La principale technique utilisée en recommandation est la factorisation matricielle. Cette méthode a fait ses preuves lors du concours Netflix (Bennett et Lanning, 2007) grâce aux travaux de référence de (Koren *et al.*, 2009). Ces modèles extraient des profils latents d’utilisateurs et de produits directement à partir de la matrice de note de façon à ce que la note prédite  $\hat{r}_{ui}$  résulte du produit scalaire entre les profils du produit  $i$  et de l’utilisateur  $u$ .

$$\hat{r}_{ui} \approx q_i^T p_u \quad [1]$$

À la manière des méthodes de segmentation thématiques on estime que chaque dimension latente des profils correspond à un aspect concret. La présence du même aspect dans deux profils provoque une union locale faisant monter la note.

En recommandation, les notes sont fortement biaisées. Certaines personnes ont tendance à toujours mettre de bonnes/mauvaises notes et certains produits ont tendance à être sur/sous notés. Classiquement, ces biais sont modélisés en ajoutant une moyenne globale  $\mu$ , un biais produit  $b_i$  et un biais utilisateur  $b_u$  à [1]. De ce fait, une note  $\hat{r}_{ui}$  est prédite de cette façon:

$$\hat{r}_{ui} = \mu + b_i + b_u + q_i^T p_u \quad [2]$$

$$\min_{q,p,b} \sum_{(u,i)} (r_{ui} - \mu + b_i + b_u + q_i^T p_u)^2 + \lambda(||p_u||^2 + ||q_u||^2 + b_u^2 + b_i^2) \quad [3]$$

Grâce à cette méthode, les profils latents capturent uniquement la déviation à la moyenne, ce qui permet une estimation moyenne lorsqu’aucune information n’est disponible. De nombreuses variantes ont été proposées pour prendre en compte le temps (Koren, 2009 ; McAuley et Leskovec, 2013a), les liens sociaux (Guy et Carmel, 2011) ou le texte (cf section 2.2).

### 2.1.2. *Modèle par voisinage*

L’approche par plus proches voisins est souvent considérée comme obsolète depuis l’avènement des techniques de factorisation matricielle (Koren, 2008). Bien qu’incontestablement lourde à mettre en œuvre du point de vue computationnel, elle est toujours d’actualité, du fait de sa précision, sa simplicité et sa popularité. Prédire la note  $\hat{r}_{ui}$  d’un utilisateur  $u$  sur un produit  $i$  en utilisant les  $k$  plus proches produits, ou utilisateurs, voisins  $\mathcal{N}$  se fait à l’aide d’une somme pondérée des notes des produits, ou utilisateurs, similaires:

$$\hat{r}_{ui} = \frac{\sum_{v \in \mathcal{N}} \alpha_{uv} r_{vi}}{\sum_{v \in \mathcal{N}} \alpha_{uv}}, \quad \hat{r}_{ui} = \frac{\sum_{j \in \mathcal{N}} \alpha_{ij} r_{uj}}{\sum_{j \in \mathcal{N}} \alpha_{ij}} \quad [4]$$

Où  $\alpha_{uv}$  et  $\alpha_{ij}$  représentent respectivement la similarité entre utilisateurs  $u$  et  $v$  et entre produits  $i$  et  $j$ . Un certain nombre de mesures de similarités existent comme la corrélation de Pearson qui mesure la tendance qu’ont les utilisateurs de noter de manière identique. Les notes peuvent être normalisées par la moyenne afin de prendre en compte les biais de notation des utilisateurs. (Desrosiers et Karypis, 2011) présente ces méthodes de manière détaillée.

Pour nous, cette stratégie présente l’intérêt de décorréler la mesure de similarité entre  $i$  et  $j$  ( $\alpha_{ij}$ ) et la tâche visée, par exemple la prédiction de note. Nous allons exploiter une telle stratégie pour démontrer la richesse des profils textuels dans la tâche de filtrage collaboratif, à la manière de ce qui est fait lorsque les recommandations sont basées sur le contenu.

## 2.2. Filtrage collaboratif avec données textuelles

Plusieurs méthodes de filtrage collaboratif utilisent déjà le texte afin d'améliorer leurs précisions. (McAuley et Leskovec, 2013b ; Ling *et al.*, 2014) alignent les représentations latentes de sujets extraits par *Latent Dirichlet Allocation* (Blei *et al.*, 2003) avec les profils des utilisateurs et des produits issus de l'analyse des triplets (utilisateur, produit, note). (Poussevin *et al.*, 2014) construit des profils textuels basés sur des sacs de mots et les incorpore dans la règle de prédiction en ajoutant un terme dans [2].

Bien que notre travail présente certaines similarités avec ceux précédemment cités, il ne s'agit pas d'une extension des travaux de (Koren *et al.*, 2009). Au contraire, nous cherchons à mettre en œuvre un algorithme de recommandation par le texte avec comme enjeu majeur l'explication en plus de la performance. Nos travaux sont philosophiquement plus proches de ceux de (Ganu *et al.*, 2009) qui cherche à démontrer l'importance du texte pour la recommandation.

## 3. Espace latent d'avis en ligne

Notre but est de construire un espace latent où chaque utilisateur, produit, note et mot est représenté par un vecteur afin d'encoder l'ensemble des informations disponibles dans un avis (figure 1). De plus, il doit exister une mesure de similarité entre chacune de ces représentations, que des utilisateurs qui aiment les mêmes choses soit plus proches que ceux qui aiment des choses différentes.

Pour obtenir ces propriétés, nous nous appuyons sur l'approche *paragraph2vec* (Le et Mikolov, 2014) qui est une extension de *word2vec* (Mikolov *et al.*, 2013) aux documents. Nous modifions celui-ci en un algorithme que nous appelons *Conceptual Skip-Gram* (CSG). Le texte devient le dénominateur commun des acteurs du système (utilisateur, produit, note). La modification du *Skip-Gram* doit permettre la projection puis la comparaison des différents concepts, même lorsqu'ils ne sont pas homogènes.

### 3.1. Conceptual Skip-Gram

Notre modèle dérive donc de *paragraph2vec* de (Le et Mikolov, 2014). Plus exactement, nous utilisons un modèle appelé *Distributed Bag-of-Word* (DBOW), extension du modèle *Skip-Gram* (SG). En effet, en considérant les utilisateurs, les produits et les notes comme des éléments conceptuels caractérisés par un ensemble de mots les représentant il est possible de les introduire dans l'algorithme précédemment cité comme des paragraphes: le but est de trouver la meilleure représentation pour chaque concept, celle qui permette au mieux de prédire ses mots (figure 2).

Nous utilisons la technique dite du *negative sampling* de (Mikolov *et al.*, 2013) décrite en détail par (Goldberg et Levy, 2014). Cette méthode est une simplification de la *Noise-Contrastive Estimation* (Gutmann et Hyvärinen, 2010) qui postule qu'un

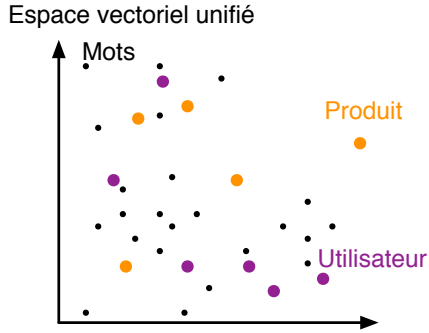


**Figure 1.** Représentation de l'espace latent d'avis en ligne. Une fois les concepts hétérogènes projetés dans le même espace, différentes tâches deviennent traitables.

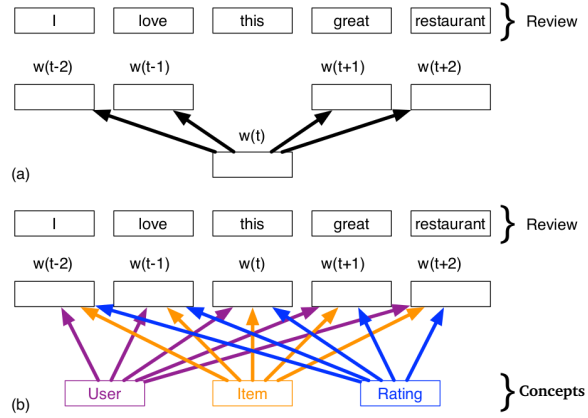


**Figure 2.** (a) Skip-Gram classique sur du texte. (b) Conceptual Skip-Gram qui projette les concepts (utilisateur, produit, note) dans l'espace des mots.

bon modèle doit être capable de différencier les données réelles du bruit grâce à la régression logistique.

Les représentations latentes de chaque mot et de chaque "concept" des avis (utilisateur, produit, note) sont apprises ensemble avec deux objectifs séparés, mais similaires: les mots cooccurents localement dans les avis (à l'intérieur d'une fenêtre) doivent être proches les uns des autres (et les autres doivent être plus loin); chaque concept doit être proche de ses mots (et éloigné de ceux des autres). Dans la pratique, les positions des mots sont apprises en premier, les concepts sont ensuite positionnés par rapport aux mots. Autrement dit, avec  $\sigma(x) = \frac{1}{1+e^{-x}}$ , les fonctions objectives sont les suivantes:

$$\operatorname{argmax}_v \left( \log \sigma(v_a'^T \cdot v_w) + \sum_{i=1}^k \mathbb{E}_{v_b \sim P_n(\mathbf{w})} \log \sigma(-v_b'^T \cdot v_w) \right) \quad [5]$$

$$\operatorname{argmax}_v \left( \log \sigma(v_a'^T \cdot v_c) + \sum_{i=1}^k \mathbb{E}_{v_b \sim P_n(\mathbf{w})} \log \sigma(-v_b'^T \cdot v_w) \right) \quad [6]$$

Où  $v_w$  et  $v_c$  sont les mots (resp. concepts) et leurs mots associés  $v_a$  (cooccurences ou contextes). Les  $v_b$  sont le bruit, tirées selon une loi unigramme, élevée à la puissance  $3/4$ <sup>1</sup>. Contrairement à (Le et Mikolov, 2014) qui utilise un softmax hiérarchique pour l'apprentissage, nous utilisons seulement le negative sampling comme décrit ci-dessus.

1. La puissance  $3/4$  permet de lisser la distribution unigramme et de faire ressortir les mots rares (Levy et al., 2015)



Après entraînement, on obtient un espace vectoriel en  $n$  dimension où les concepts et les mots similaires sont proches. La nombre de représentations obtenues est de  $|mots| + |concepts|$ .

### 3.2. Exploitation de l'espace latent d'avis

À l'instar de word2vec, le cosinus s'utilise comme mesure de similarité afin de trouver les éléments proches. Par convenance, nous basculons cette mesure sur l'intervalle  $[0, 1]$  afin de l'utiliser comme pondération dans [4].

$$\alpha_{uv} = \frac{\frac{\langle u, v \rangle}{\|u\| \times \|v\|} + 1}{2} \in [0, 1] \quad [7]$$

En considérant chaque utilisateur, produit et note comme concepts grâce aux mots qui les représentent, on obtient un espace où chaque composante des avis en ligne est représentée (figure 1). Il est également possible de projeter individuellement chaque avis et chaque phrase, mais le nombre de représentations deviendrait trop important. Les phrases sont donc projetées dans l'espace latent en moyennant les représentations de leurs mots, comme suggéré par (Kageback *et al.*, 2014).

#### Prédiction de notes

La mesure de similarité [7] rend possible le tri et la sélection des produits, des utilisateurs ou même des phrases et mots voisins à un autre produit, utilisateur ou même mot. C'est à l'aide de cette mesure que nous effectuons nos recommandations. Plus précisément, pour prédire la note  $\hat{r}_{ui}$  d'un utilisateur  $u$  sur un nouveau produit  $i$  on utilise les notes déjà données sur ce même produit  $i$  par les autres utilisateurs  $v \in \mathcal{N}$ ,  $r_{uv}$ , pondérées par leur similarité à l'utilisateur  $u$ ,  $\alpha_{uv}$ . En pratique, on calcule seulement cette moyenne sur le  $k$  plus proche voisinage  $\mathcal{N}$ . Par ailleurs, les notes données par chaque utilisateur sont normalisées par la moyenne de chaque utilisateur  $\mu_u$  pour prendre en compte leur biais de notation. Symétriquement, la même opération est possible avec les notes déjà données par l'utilisateur  $u$  sur des produits similaires au produit  $i$ . De ce fait, on obtient deux modèles de prédiction [8] qui découlent de [4] avec les  $\alpha$  issus de [7]:

$$\hat{r}_{ui} = \mu_u + \frac{\sum_{v \in \mathcal{N}} \alpha_{uv} (r_{vi} - \mu_v)}{\sum_{v \in \mathcal{N}} \alpha_{uv}}, \quad \hat{r}_{ui} = \mu_i + \frac{\sum_{j \in \mathcal{N}} \alpha_{ij} (r_{uj} - \mu_j)}{\sum_{j \in \mathcal{N}} \alpha_{ij}} \quad [8]$$

La prédiction est donc obtenue en moyennant soit:

- les avis du même utilisateur sur des produits proches,
- les avis des autres utilisateurs sur le produit cible.

| Dataset      | #Users  | #Items | #Reviews | #Appren. | #Valid. + #Test |
|--------------|---------|--------|----------|----------|-----------------|
| Ratebeer     | 29073   | 110283 | 2844778  | 2274819  | 569959          |
| BeerAdvocate | 33368   | 66032  | 1585241  | 1268174  | 317067          |
| Movies       | 1223167 | 164446 | 2985563  | 3676787  | 920365          |
| Music        | 1133669 | 420466 | 3229487  | 2583532  | 645955          |
| Yelp         | 366402  | 60784  | 1566139  | 1253144  | 312995          |

**Tableau 1.** *Détails des corpus d'évaluation*

### Prédiction de revues

La similarité [7] étant basée sur le texte écrit par les utilisateurs, il est également légitime de l'utiliser pour extraire du texte des avis existant afin de le présenter en accompagnement de la recommandation. Deux possibilités sont étudiées:

- présenter l'avis complet le plus proche,
- créer un résumé personnalisé par extraction de phrases selon la méthode de (Kageback *et al.*, 2014).

Dans les deux cas, nous avons envisagé deux options:

- prendre les avis du même utilisateur pour d'autres produits (intérêt limité mais grande proximité de style d'écriture),
- proposer les avis des autres utilisateurs sur le produit cible.

## 4. Expériences

Nous utilisons notre espace latent pour différentes tâches. D'abord sur l'épreuve classique de prédiction de notes. Puis sur la prédiction des remarques textuelles, tâche originale introduite par (Poussevin *et al.*, 2014). Enfin, nous montrons les capacités d'extraction de mots-clés et d'analyse de sentiment de notre espace latent textuel.

Nous sélectionnons différents corpus d'avis en ligne afin d'évaluer nos systèmes de recommandation (tableau 1). Pour évaluer la prédiction de notes, nous utilisons l'erreur moyenne au carré (MSE), métrique standard de cette application. Nos résultats sont comparés avec ceux publiés dans (McAuley et Leskovec, 2013b). La prédiction de remarques étant une application sans méthode d'évaluation habituelle, nous gardons celle utilisée par (Poussevin *et al.*, 2014), afin de comparer nos résultats.

#### 4.1. Prétraitement des avis et paramètres d'apprentissage

Afin d'obtenir l'ensemble d'apprentissage, il convient d'effectuer différents prétraitements sur les avis extraits. D'abord, tous les duplicatas d'avis sont retirés<sup>2</sup>. L'ensemble des marques de ponctuations à l'exception des points sont retirées. Les mots apparaissant moins de 10 000 fois sont supprimés, de ce fait chaque base d'avis contient entre 1500 et 3000 mots distincts. Les mots restants sont de véritables pivots du langage qui vont jouer le rôle des aspects latents des techniques de factorisation matricielle. Les avis sont enfin subdivisés en phrases reliées à leurs concepts à l'aide des points restants. Un extrait de phrases après prétraitement est présenté figure 3.

```
r_1.0 drinkable but not great
r_2.0 Not much else going on here
r_3.0 Spicy lactic dry taste
r_4.0 Taste quite bitter super clove esters big yeast
r_5.0 The best beer in the world
```

**Figure 3.** Phrases après prétraitement (associées à leurs notes) extraites du corpus Ratebeer.

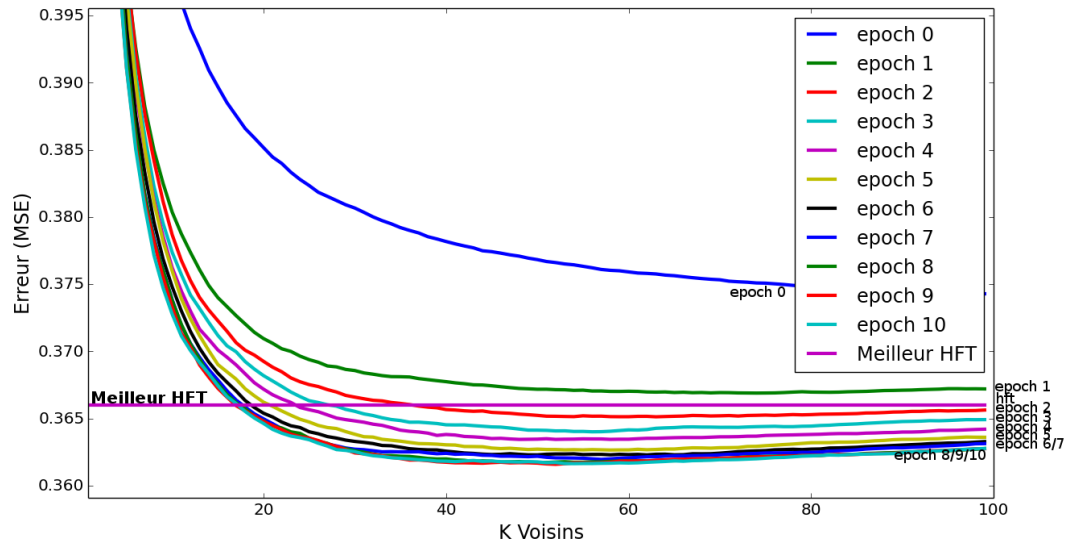


**Figure 4.** Erreur en MSE en validation en fonction de la taille du voisinage  $\mathcal{N}$  à chaque itération du modèle sur les données Beeradvocate. Le modèle converge en 5 à 8 itérations et le nombre de voisins optimaux à considérer varie de 9 à 33 selon les jeux de données.

Nous suivons le protocole de (McAuley et Leskovec, 2013b) ; 80% des avis sont destinés à l'apprentissage et 20% divisés à parts égales pour la validation et l'éva-

2. Dans les corpus, certains avis apparaissent plusieurs fois pour différentes raisons. Il est important d'avoir peu de doublons afin d'obtenir des résultats non biaisés.

luation. Nous supprimons les avis où aucun historique d'utilisateur ou produit n'est disponible.

Nous utilisons une version modifiée de *Word2Vec* pour apprendre les vecteurs des utilisateurs, des produits et des notes. Les hyper paramètres sont les suivants: les vecteurs ont une taille de 200, le pas de gradient est 0.050 et le paramètre d'échantillonnage est de  $10^{-5}$ . Nous itérons jusqu'à 10 fois sur le corpus d'apprentissage. Dans la pratique, la convergence est atteinte entre 5 et 8 itérations. Ces paramètres sont ceux proposés par (Mikolov *et al.*, 2013). Ils se sont révélés les mieux adaptés à notre problème. Enfin, le nombre  $k$  de voisins est choisi grâce à une phase de validation. Selon les jeux de données, le nombre de voisins permettant de minimiser la MSE varie de 9 à 33. La figure 4 montre l'évolution de la MSE en fonction du nombre d'itérations et du nombre de voisins. L'apprentissage est très rapide: peu d'itérations suffisent pour converger. Par contre l'inférence est plus chère, il est nécessaire de considérer un grand voisinage pour que la méthode fonctionne correctement.

| Dataset  | Moyenne | Fact. Mat    | HFT*  | CSG-kNN |              | k* |
|----------|---------|--------------|-------|---------|--------------|----|
|          |         |              |       | util.   | prod.        |    |
| Ratebeer | 0.701   | 0.306        | 0.301 | 0.336   | <b>0.286</b> | 23 |
| Beeradv. | 0.521   | 0.371        | 0.366 | 0.382   | <b>0.357</b> | 29 |
| Movies   | 1.678   | <b>1.118</b> | 1.119 | 1.39    | 1.304        | 33 |
| Music    | 1.261   | <b>0.957</b> | 0.969 | -       | 1.201        | 26 |
| Yelp     | 1.890   | 1.49         | -     | 1.591   | <b>1.407</b> | 27 |

**Tableau 2.** Précision de la prédiction de note en MSE. La référence HFT\* (meilleur modèle "Hidden Factor & Topic model") est reporté de (McAuley et Leskovec, 2013b). CSG-knn: Meilleur  $k$ -voisins dans l'espace d'avis utilisant  $k^*$  voisins

#### 4.2. Prédiction de Notes

Pour prédire une note  $\hat{r}_{ui}$  associée à un couple (utilisateur  $u$ , produit  $i$ ), nous utilisons la méthode de filtrage collaboratif par voisinage, décrite par (Desrosiers et Karypis, 2011). Deux techniques classiques ont été évaluées: La prédiction grâce au voisinage utilisateur et celle grâce au voisinage produit [8]. Il s'agit d'agréger les notes des utilisateurs similaires ou les notes précédemment mises sur des produits proches.

Les résultats obtenus en prédiction des notes sont présentés dans le tableau 2. Les meilleurs résultats obtenus à l'aide de nos modèles de plus proches voisins (CSG-knn utilisateur ou produit) sont comparés à un modèle qui prédit systématiquement la note moyenne du corpus d'apprentissage, un modèle de factorisation matriciel classique [3] et aux modèles HFT présentés par (McAuley et Leskovec, 2013b). Les résultats de HFT sur le corpus Yelp ne sont pas reportés, car le corpus utilisé est différent.

En règle générale nos performances sont proches de celles de (McAuley et Leskovec, 2013b) et même parfois meilleures. Notre modèle est particulièrement performant

sur les corpus des sites d'amateur de bières comme le montre la figure 4. Le vocabulaire utilisé y est spécifique et les revues particulièrement riches comparées aux revues type Amazon. Il est important de comprendre que contrairement aux autres modèles, l'espace latent n'est pas optimisé sur un critère de prédiction de note. Obtenir des résultats proches de l'état de l'art, voire meilleurs, souligne la pertinence de ces profils textuels pour la recommandation.

Sur Movies et Music, à l'inverse, le texte agit comme un bruit par rapport à la factorisation matricielle simple: les résultats du modèle HFT étaient légèrement impactés, mais notre plus grande dépendance au texte nous expose plus durement au phénomène.

#### **4.3. Problème du démarrage à froid**

Les techniques de factorisation matricielle se basent sur l'historique de notation de l'utilisateur et ne sont efficaces qu'à partir d'un certain nombre de voisins ( $> 20$ ). Elles ne permettent donc aucune recommandation à un nouvel utilisateur ou pour un nouveau produit, c'est le problème du démarrage à froid. Avec notre méthode, il est possible d'utiliser des textes tiers relatifs aux nouveaux utilisateurs ou aux nouveaux produits (description, blogs, tweet...) et de les utiliser pour projeter les nouveaux éléments dans notre espace textuel.

Comme expliqué en section 3.2, la note prédite pour un couple (utilisateur, produit) dépend soit des notes données par les utilisateurs proches au produit cible, soit des notes données par l'utilisateur visé à des produits proches de la cible. Une fois le nouvel élément (utilisateur ou produit) positionné dans l'espace, et en utilisant l'approche adaptée au nouvel entrant, notre système est immédiatement opérationnel sans devoir attendre la constitution d'un historique de notation. La seule différence avec [8] est que l'on ne normalise pas les notes par les moyennes utilisateurs et produit (ces données étant indisponibles dans le cas du démarrage à froid).

Cette stratégie est difficile à évaluer sur les corpus de revues où les utilisateurs et les produits ont été anonymisés. Nous proposons le protocole suivant:

- 1) deux ensembles de nouveaux utilisateurs et produits sont constitués,
- 2) dans ces ensembles, nous ne considérons que le texte (comme s'il avait été acquis depuis une source tierce),
- 3) nous apprenons des profils pour les nouveaux éléments et testons la prédiction de notes sur les données de test.

Ce protocole est certes biaisé, les données servant à constituer les profils étant exactement de la même nature que les données de test, mais il permet d'avoir une première idée de l'apport de notre stratégie.

Les résultats sont présentés dans le tableau 3. Il y a deux cas de figure à prendre en compte. L'arrivée de nouveaux utilisateurs et l'arrivée de nouveaux produits. On ob-

| Dataset  | Moy.  | Nouvel Utilisateur |              | Nouveau Produit |              |
|----------|-------|--------------------|--------------|-----------------|--------------|
|          |       | Moy. Prod.         | CSG- $k$ NN  | Moy. Util.      | CSG- $k$ NN  |
| Ratebeer | 0.701 | 0.341              | <b>0.333</b> | 0.599           | <b>0.371</b> |
| Beeradv. | 0.518 | 0.397              | <b>0.386</b> | 0.490           | <b>0.419</b> |

**Tableau 3.** Précision de la prédiction de notes en MSE avec [4] pour simuler le démarrage à froid. Moy, Prod, Util: Prédit respectivement la note moyenne du corpus d'apprentissage, globale, par produit, par utilisateur.

serve que systématiquement la prédiction grâce à l'espace latent est meilleure qu'en utilisant les moyennes produits ou utilisateurs (techniques la plus utilisée pour démarrer le processus lorsque le moteur de recommandation repose sur la factorisation matricielle).

Cette approche élégante au problème du démarrage à froid repose entièrement sur l'hypothèse d'un espace latent textuel sous-jacent au processus de recommandation. En production, il suffit d'imaginer un processus d'enregistrement des utilisateurs demandant une adresse Twitter, Facebook, un lien vers un site personnel ou une partie d'un historique de navigation web.

#### 4.4. Prédiction de remarques

Nous suivons la proposition de (Poussevin *et al.*, 2014) sur la prédiction de remarques qu'un utilisateur  $u$  écrirait sur un produit  $i$ . Cette tâche peut à la fois être vue comme un enrichissement de la recommandation, une explication, mais également comme un résumé personnalisé des qualités et défauts des produits. Nous sommes convaincus que montrer des informations textuelles relatives aux caractéristiques produits et au ressenti utilisateur ne peut qu'être bénéfique pour expliciter la recommandation faite. La génération pure de texte restant une tâche complexe, nous adoptons une méthode d'extraction de résumés et nous focalisons sur deux techniques. Premièrement, nous sélectionnons un avis complet du produit  $i$  comme résumé. Sinon, nous filtrons  $n$  phrases parmi celles sur le produit  $i$ . ( $n$  est sélectionné en fonction de la moyenne de phrases qu'écrit l'utilisateur  $u$ .)

Les résultats obtenus sont évalués à l'aide des unigrammes et des bigrammes en utilisant les mesures ROUGE-1 et ROUGE-2 (Lin, 2004). Ces métriques montrent à quel point le texte prédit est égal au texte original. Elles peuvent être vues comme des mesures de rappel sur les  $n$ -grammes. N'ayant pas de score de référence, nous en calculons deux: une borne inférieure qui correspond à une sélection d'avis et de phrases faite au hasard. Et une borne supérieure étant un oracle choisissant systématiquement l'avis ou les phrases maximisant le score ROUGE- $n$  possible.

| avis complet     |         |       |       |        |         |       |       |        |
|------------------|---------|-------|-------|--------|---------|-------|-------|--------|
| Dataset          | Rouge-1 |       |       |        | Rouge-2 |       |       |        |
|                  | Rand    | Prod. | Util. | Oracle | Rand    | Prod. | Util. | Oracle |
| Ratebeer         | 0.243   | 0.271 | 0.351 | 0.568  | 0.037   | 0.052 | 0.117 | 0.278  |
| Beeradv.         | 0.280   | 0.304 | 0.366 | 0.507  | 0.045   | 0.063 | 0.114 | 0.207  |
| Music            | 0.255   | 0.407 | 0.403 | 0.545  | 0.043   | 0.212 | 0.215 | 0.255  |
| Movies           | 0.253   | 0.345 | 0.351 | 0.565  | 0.040   | 0.146 | 0.156 | 0.208  |
| Yelp             | 0.249   | 0.290 | 0.284 | 0.512  | 0.037   | 0.067 | 0.072 | 0.154  |
| <i>n</i> phrases |         |       |       |        |         |       |       |        |
| Ratebeer         | 0.157   | 0.239 | 0.217 | 0.591  | 0.011   | 0.043 | 0.046 | 0.288  |
| Beeradv          | 0.122   | 0.225 | 0.244 | 0.606  | 0.010   | 0.046 | 0.057 | 0.249  |
| Music            | 0.167   | 0.291 | 0.220 | 0.551  | 0.017   | 0.026 | 0.041 | 0.282  |
| Movies           | 0.173   | 0.100 | 0.253 | 0.396  | 0.014   | 0.012 | 0.048 | 0.142  |
| Yelp             | 0.150   | 0.246 | 0.279 | 0.550  | 0.011   | 0.041 | 0.050 | 0.208  |

**Tableau 4.** Évaluation de la prédiction de remarques. *Prod.*: sélection des avis/phrases dans les données produit (les avis des autres utilisateurs). *Util.*: sélection dans les données de l'utilisateur (ses propres avis sur des produits similaires)

#### Vérité terrain

*Note:* 5.0

*Commentaire:* "Old school. traditional mom n'pop quality and perfection. The best fish and chips you ll ever enjoy and equally superb fried shrimp. A great out of the way non corporate vestige of Americana. You will love it."

#### Prédiction:

*Note:* 4.70

*Avis complet:* "Good fish sandwich"

*Choix d'*n* phrases:* "The staff is extremely friendly. On top of being extremely large portions it was incredibly affordable. Most of girls are good one is very slow one is amazing. The fish was very good but the Reuben was to die for. Both dishes were massive and could very easily be shared between two people."

**Figure 5.** Exemple de prédiction de note et de remarques sur le corpus Yelp

Le tableau 4 présente les résultats. En moyenne, le score ROUGE-1 est élevé, nous prédisons donc beaucoup de mots réellement utilisés dans l'avis original (de 10% à 40%). Néanmoins, le score élevé obtenu en utilisant le hasard montre que de nombreux mots doivent être présents dans tous les commentaires. Quoi qu'il en soit, nous sommes systématiquement meilleurs que cette borne inférieure. En comparant nos résultats avec ceux de (Poussevin *et al.*, 2014) sur les données Ratebeer, on peut remarquer que notre modèle est systématiquement meilleur sur l'extraction d'avis complet (+23% en ROUGE-1 et +246 % en ROUGE-2). Par contre sur la sélection de *n* phrases





## 5. Conclusion

La recommandation a pour objectif principal la personnalisation de l'accès à l'information via la construction de profils produit et utilisateur. Les techniques récentes de filtrage collaboratif par factorisation matricielle issue du concours Netflix se sont principalement focalisées sur la performance, laissant les explications liées aux suggestions au second plan.

La principale contribution de cet article est de mettre en avant le texte des avis et de s'en servir comme principale information pour la recommandation et son explication. Grâce à un espace latent textuel encodant les similarités entre utilisateurs et produits et en utilisant une technique de filtrage collaboratif basé sur cette même similarité, nous montrons qu'il est possible d'obtenir des résultats proches, voire meilleurs, que l'état de l'art sur la tâche de prédiction de note. Nous démontrons également les qualités de notre approche dans le cas d'un démarrage à froid en proposant une méthode élégante pour pallier ce problème. De plus, nous montrons qu'il est possible de prédire les remarques d'un utilisateur sur un produit avec une certaine efficacité.

Finalement, nos expériences démontrent qu'une similarité textuelle peut être utilisée pour concevoir un système de recommandation non seulement efficace en prédiction de notes mais plus large en terme d'usage. Cette nouvelle approche apporte à la fois une solution élégante en amont, pour le démarrage à froid, et des propositions intéressantes en aval avec la génération d'un texte à montrer à l'utilisateur en complément de la note.

## 6. Remerciements

Ce travail a été réalisé en partie avec le soutien du projet FUI ITER-RH (Investissement d'Avenir).

## 7. Bibliographie

- Ben-elazar S., Koenigstein N., « A Hybrid Explanations Framework for Collaborative Filtering Recommender Systems », *RecSys 2014*, 2014.
- Bennett J., Lanning S., « The Netflix Prize », *KDD Cup and Workshopp*. 3-6, 2007.
- Blei D. M., Ng A. Y., Jordan M. I., Lafferty J., « Latent Dirichlet Allocation », *Journal of Machine Learning Research*, vol. 3, p. 2003, 2003.
- Contardo G., Denoyer L., Artieres T., « Representation Learning for cold-start recommendation », , n° 1, p. 1-10, 2014.
- Desrosiers C., Karypis G., « A comprehensive survey of neighborhood-based recommendation methods », *Recommender Systems Handbook*, vol. 69, n° 11, p. 107-144, 2011.
- Ganu G., Elhadad N., Marian A., « Beyond the Stars : Improving Rating Predictions using Review Text Content », *WebDB*, vol. 9, p. 1-6, 2009.

- Gauthier L.-A., Piwowarski B., Gallinari P., « Polarité des jugements et des interactions pour le filtrage collaboratif et la prédiction de liens sociaux », *CORIA*, 2015.
- Goldberg Y., Levy O., « word2vec Explained: Deriving Mikolov et al.'s Negative-Sampling Word-Embedding Method », *arXiv preprint arXiv:1402.3722*, n° 2, p. 1-5, 2014.
- Gutmann M., Hyvärinen A., « Noise-contrastive estimation: A new estimation principle for unnormalized statistical models », *International Conference on Artificial Intelligence and Statistics*, p. 1-8, 2010.
- Guy I., Carmel D., « Social Recommender Systems », *Proceedings of the 20th international conference companion on World wide web*, Acm, p. 283-284, 2011.
- Kageback M., Mogren O., Tahmasebi N., Dubhashi D., « Extractive Summarization using Continuous Vector Space Models », *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)@EACL 2014*p. 31-39, 2014.
- Koren Y., « Factorization meets the neighborhood: a multifaceted collaborative filtering model », *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*p. 426-434, 2008.
- Koren Y., « Collaborative filtering with temporal dynamics », *Proc. of KDD '09*, ACM Press, p. 447-456, 2009.
- Koren Y., Bell R., Volinsky C., « Matrix Factorization Techniques for Recommender Systems », *Computer*, vol. 42, p. 42-49, 2009.
- Le Q., Mikolov T., « Distributed Representations of Sentences and Documents », *International Conference on Machine Learning - ICML 2014*, vol. 32, p. 1188–1196, 2014.
- Levy O., Goldberg Y., Dagan I., « Improving Distributional Similarity with Lessons Learned from Word Embeddings », *Transactions of the Association for Computational Linguistics*, vol. 3, p. 211-225, 2015.
- Lin C. Y., « Rouge: A package for automatic evaluation of summaries », *Proceedings of the workshop on text summarization branches out (WAS 2004)*p. 25-26, 2004.
- Ling G., Lyu M. R., King I., « Ratings Meet Reviews , a Combined Approach to Recommend », p. 105-112, 2014.
- McAuley J., Leskovec J., « From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews », *In Proceedings of the 22nd international conference on World Wide Web*p. 897-908, 2013a.
- McAuley J., Leskovec J., « Hidden factors and hidden topics: understanding rating dimensions with review text », *Proceedings of the 7th ACM conference on Recommender systems - RecSys '13*p. 165-172, 2013b.
- Mikolov T., Chen K., Corrado G., Dean J., « Distributed Representations of Words and Phrases and their Compositionality arXiv : 1310 . 4546v1 [ cs . CL ] 16 Oct 2013 », *arXiv preprint arXiv:1310.4546*p. 1-9, 2013.
- Poussevin M., Guigue V., Gallinari P., « Extended Recommendation Framework: Generating the Text of a User Review as a Personalized Summary », 2014.
- Tintarev N., Masthoff J., « A survey of explanations in recommender systems », *Proceedings - International Conference on Data Engineering*p. 801-810, 2007.

---

# Apprentissage de Représentations Probabilistes pour la Prédiction de Diffusions d'Informations sur les Réseaux Sociaux<sup>1</sup>

Simon Bourigault, Sylvain Lamprier, Patrick Gallinari

*Sorbonne Universités, UPMC Univ Paris 06, UMR 7606, LIP6, F-75005, Paris, France. CNRS, UMR 7606, LIP6, F-75005, Paris, France.  
prénom.nom@lip6.fr*

---

**RÉSUMÉ.** *Le modèle Independent Cascades (IC) est un modèle central pour la capture des dynamiques de diffusion d'information sur les réseaux sociaux. Nous nous intéressons ici à l'apprentissage des probabilités de transmission utilisées par ce modèle. Plutôt que de directement travailler sur le graphe du réseau social considéré, ce qui implique un coût important dû au nombre de paramètres à apprendre pour les réseaux denses, nous proposons une approche basée sur des techniques d'apprentissage de représentations, afin d'alléger le processus et de gagner en généralisabilité. L'idée est de chercher une projection des utilisateurs du réseau dans un espace vectoriel, de manière à ce que les distances entre les individus représentent leurs probabilités de transmission d'information. Les expérimentations menées démontrent la pertinence de l'approche pour la modélisation des dynamiques de diffusion.*

**ABSTRACT.** *Based on the well-known Independent Cascade (IC) model, we embed users of the social network in a representation space to extract more robust diffusion probabilities than those defined by classical graphical learning approaches for social influence modeling. Better generalization abilities provided by the use of such a projection space allows our approach to present good performances on various real-world datasets, for both diffusion prediction and influence relationships inference tasks.*

**MOTS-CLÉS :** *Apprentissage de Représentation, Diffusion d'Information.*

**KEYWORDS:** *Representation Learning, Information Diffusion.*

---

<sup>1</sup>Traduction d'un article publié en anglais à WSDM 2016 (Bourigault *et al.*, 2016)

## 1. Introduction

Au cours des quinze dernières années, les réseaux sociaux sont devenus un média d'information incontournable, mettant en jeu des dynamiques complexes de communication entre utilisateurs. La modélisation de la diffusion d'information sur les réseaux constitue depuis lors un enjeu majeur, pour diverses tâches telles que l'identification de leaders d'opinions, la prédiction ou la maximisation de l'impact d'un contenu diffusé, la détection de communautés d'opinions, ou plus généralement l'analyse des dynamiques du réseau considéré.

Nous proposons dans cet article une approche basée sur l'apprentissage de représentations. Plutôt que d'apprendre des probabilités de transmission pour tous les liens du réseau considéré, tel que c'est le cas dans les approches classiques basées sur le très populaire modèle *Independent Cascades* (IC) (Gruhl *et al.*, 2004), nous modélisons la diffusion au sein d'un espace de représentation vectoriel. Cela nous permet à la fois de limiter la complexité du modèle et de gagner en capacité de généralisation. De la même manière qu'IC, nous considérons un modèle itératif, dans lequel le contenu diffusé transite de noeuds en noeuds du réseau, mais où les probabilités de transmission sont issues des positions relatives des utilisateurs dans l'espace de projection. En outre, nous considérons que le graphe de diffusion est inconnu, et que l'information peut a priori se transmettre de n'importe quel utilisateur à n'importe quel autre du réseau. Nous modifions l'algorithme d'apprentissage proposé dans (Saito *et al.*, 2008) pour l'adapter à cette formulation.

La suite de cet article est organisée comme suit : la section 2 présente les motivations de l'approche proposée en la positionnant par rapport à d'autres travaux existants. La section 3 décrit notre approche. La section 4 présente les résultats obtenus sur différents jeux de données réels.

## 2. Motivations et Travaux Connexes

Bien que l'étude du phénomène de diffusion soit initialement issue des domaines de l'épidémiologie et de la sociologie, le développement des réseaux sociaux en ligne a permis à toute une littérature d'émerger dans le domaine de l'informatique. La plupart des travaux existants sont basés sur deux modèles fondamentaux : l'*Independent Cascades Model* (IC) (Kempe *et al.*, 2003) et le *Linear Threshold Model* (LT) (Goldenberg *et al.*, 2001). Ces deux approches modélisent la diffusion à l'échelle des utilisateurs : alors que le modèle IC fait une hypothèse d'indépendance des événements d'infection<sup>1</sup> et considère un processus de diffusion comme une cascade s'écoulant le long des liens du réseau, le modèle LT s'appuie sur une hypothèse d'additivité de l'influence

---

<sup>1</sup>Les termes "infection" ou "contamination" sont utilisés pour désigner le fait qu'un individu a été atteint par le contenu diffusé.

et décide de l'infection d'un utilisateur en fonction de la pression sociale à laquelle il est exposé. Dans cet article, nous utilisons un formalisme proche de celui du modèle IC, qui a servi de base à de très nombreux travaux. Certains s'intéressent notamment aux délais de diffusion de l'information (Saito *et al.*, 2009 ; Gomez-Rodriguez *et al.*, 2011), d'autres cherchent à prendre en compte le contenu diffusé afin d'affiner la modélisation des dynamiques de diffusion (Saito *et al.*, 2011 ; Guille et Hacid, 2012 ; Lagnier *et al.*, 2013).

Les modèles de type IC font l'hypothèse que l'information transite d'un utilisateur à l'autre en suivant les arêtes d'un graphe connu : chaque utilisateur nouvellement infecté peut à son tour infecter chacun de ses successeurs selon des probabilités de diffusion définies sur les liens du graphe. Néanmoins, en plus d'être parfois difficiles à obtenir, les liens explicites d'un réseau social (liens d'amitié par exemple) ne sont pas toujours représentatifs des vrais canaux de diffusion reliant les utilisateurs (Najar *et al.*, 2012 ; Ver Steeg et Galstyan, 2013). Les modèles doivent alors être appliqués à une population d'utilisateurs sans graphe de diffusion connu. Dans ce cas, il est possible de rechercher les liens les plus à même d'expliquer la diffusion (Gomez Rodriguez *et al.*, 2010), ou, plus simplement, de considérer un graphe complet reliant tous les utilisateurs. Malheureusement, considérer le graphe complet du réseau implique une forte complexité algorithmique, le modèle IC apprenant un paramètre indépendant pour *chaque* lien du graphe. En projetant le modèle dans un espace vectoriel, nous réduisons considérablement la complexité du modèle puisque les probabilités sont déduites directement des coordonnées des utilisateurs dans l'espace de projection.

D'autre part, les graphes sociaux possèdent de nombreuses propriétés spécifiques : distribution des degrés en loi de puissance, faible diamètre, etc... (Mislove *et al.*, 2007). Une propriété importante est la présence de *communautés* dans ces graphes, que l'on peut définir comme des groupes d'utilisateurs similaires, plus densément connectés les uns aux autres. Récemment, (Barbieri *et al.*, 2013) a montré que dans le cadre de la diffusion d'information, les communautés pouvaient non seulement être formées d'utilisateurs interagissant beaucoup les uns avec les autres (on parle de communautés *cohésives*), mais aussi d'utilisateurs interagissant avec le même autre groupe d'utilisateurs (on parle alors de communautés *bimodales*) :

- S'il y a influence entre les utilisateurs  $a$  et  $b$  d'une part, et entre les utilisateurs  $b$  et  $c$  d'autre part, alors il est vraisemblable qu'il y ait influence entre les utilisateurs  $a$  et  $c$  (communautés *cohésives*)
- S'il y a influence entre les utilisateurs  $a$  et  $c$ , entre les utilisateurs  $a$  et  $d$  et entre les utilisateurs  $b$  et  $c$ , alors il est vraisemblable qu'il y ait de l'influence entre les utilisateurs  $b$  et  $d$  (communautés *bimodales*)

L'impact de ces communautés sur la diffusion d'information a été récemment étudié dans (Barbieri *et al.*, 2013 ; Yang *et al.*, 2014). Dans l'apprentissage clas-

sique du modèle IC, les probabilités de transmission sont estimées sans aucune régularisation relative à la présence de ces communautés. Cela peut conduire à des problèmes de sur-apprentissage ou à des structures de graphes irréalistes. En revanche, bien que sans prise en compte explicite de ces communautés sociales, notre approche par projection des utilisateurs dans un espace de représentation vectoriel nous permet d'extraire naturellement ce genre de régularités. La figure 1 illustre cette approche. Nous projetons les utilisateurs de façon à ce que ceux qui interagissent soient proches les uns des autres. Deux utilisateurs similaires tendent ainsi à se rapprocher, ce qui implique des tendances de diffusion similaires sur les autres utilisateurs du réseau.

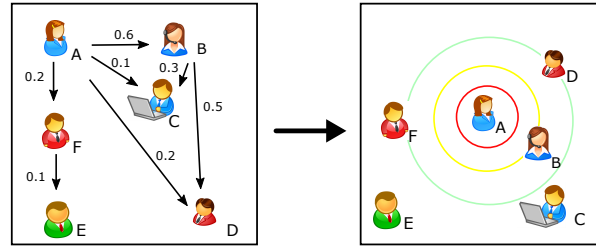


**Figure 1.** *Passage d'un graphe de diffusion à un espace de représentation vectoriel. À gauche, les valeurs associées aux arrêtes représentent les probabilités de transmission entre utilisateurs. À droite, ces probabilités de transmission sont calculées en fonction de la distance séparant les utilisateurs. Les cercles représentent des lignes de niveau d'équiprobabilité depuis l'utilisateur A.*

L'apprentissage de représentations est un domaine de recherche très actif qui a récemment donné lieu à de nombreuses approches efficaces pour diverses tâches mettant en jeu des dépendances relationnelles complexes (Bengio *et al.*, 2013). Ce genre de techniques a par exemple été employé dans les modèles de langage, comme par exemple pour le très populaire modèle *Word2Vec* (Mikolov *et al.*, 2013) où les projections des mots reflètent des similarités syntaxiques et sémantiques, et l'utilisation d'un espace de représentation vectoriel permet de régulariser ces relations. Typiquement, deux mots ayant un sens proche sont projetés plus près l'un de l'autre, et tendent donc à être prédits dans le même contexte. Plus proche de notre approche, un modèle probabiliste de prédiction de playlist a été récemment proposé dans (Chen *et al.*, 2012), dans lequel l'utilisation d'un espace de projection permet d'éviter d'avoir à définir des probabilités d'enchaînement pour chaque paire de morceaux. Notre problème diffère de ces approches par le fait que l'information se diffuse de manière arborescente<sup>2</sup>, et non selon une séquence bien définie tel que c'est le cas dans tous les domaines où ce genre de techniques ont été employées. Cela complexifie le problème car les causes des événements observés ne sont pas

<sup>2</sup>Nous nous plaçons ici dans le cadre d'un modèle de type SI (*susceptible-infected*), dans lequel seule la première infection de chaque utilisateur est considérée.

connues : on ne sait pas a priori qui a passé le contenu diffusé à tel utilisateur infecté. Il s’agit alors de faire des suppositions sur le chemin emprunté par l’information pour sélectionner les relations à intensifier dans la représentation à apprendre. Notons que récemment l’utilisation de techniques d’apprentissage de représentation a été explorée pour la prédiction de diffusion d’information (Bourigault *et al.*, 2014), mais uniquement dans un cadre statique, avec un modèle simple de prédiction des infectés finaux à partir d’infections initiales. L’approche proposée ici va plus loin en définissant un modèle itératif qui permet une capture des dynamiques du réseau et une modélisation plus fine de ses relations d’influence.

### 3. Modèle

#### 3.1. Notations

Dans la majorité des applications, la diffusion d’information dans un réseau social est observée sous la forme d’*épisodes de diffusion*. Un épisode de diffusion correspond à une séquence d’actions d’utilisateurs liées à un même contenu. Un tel épisode peut par exemple représenter une série d’utilisateurs ayant posté une vidéo particulière ou ayant utilisé une même expression se diffusant sur le réseau. L’information concernant *qui a été infecté par qui* est donc manquante. Les modèles de type cascades font généralement diverses hypothèses sur cette information manquante pour reconstituer le graphe de diffusion.

Soit un réseau social contenant  $N$  utilisateurs  $\mathcal{U} = (u_1, \dots, u_N)$ . Un épisode de diffusion  $D$  est représenté par une séquence d’utilisateurs associés à leurs temps d’infection :  $\{(u, t^D(u)) | u \in \mathcal{U} \wedge t^D(u) < \infty\}$ , où  $t^D : \mathcal{U} \rightarrow \mathbb{R}^+ \cup \{\infty\}$  retourne l’instant auquel chaque utilisateur a été infecté dans l’épisode  $D$ , ou  $\infty$  pour les utilisateurs non infectés (on considère que le premier utilisateur infecté l’est au temps  $t = 0$ , et que les autres temps sont donnés relativement à celui-ci). Nous notons  $D(t)$  l’ensemble des utilisateurs infectés dans  $D$  avant le temps  $t$  :  $D(t) = \{u \in \mathcal{U} | t^D(u) < t\}$ , et  $\bar{D}(t)$  son complémentaire. L’ensemble des utilisateurs infectés au cours de l’épisode  $D$  est noté  $D(\infty)$ .

#### 3.2. Modèle de Diffusion

Comme évoqué plus haut, nous nous basons sur le modèle IC, que nous proposons de projeter dans un espace de représentation vectoriel, pour régulariser les relations entre les utilisateurs et en réduire la complexité spatiale. Néanmoins, le modèle IC classique considère que le temps est découpé en plages et que des transmissions de contenu sont uniquement possibles entre plages de temps successifs. Cela limite grandement la capacité d’explication d’épisodes de diffusion observés dans des données réelles, la diffusion correspondant à un processus essentiellement asynchrone. D’autre part, chercher à extraire

des régularités sur des délais de diffusion observés, tel que tentent de le faire des modèles comme *NetRate* (Gomez-Rodriguez *et al.*, 2011) ou *CTIC* (Saito *et al.*, 2009), paraît difficile lorsque l'on s'intéresse à des données sociales extraites de réseaux impliquant des échanges entre humains (souvent très bruyés). Cela perturbe bien souvent l'inférence des probabilités de transmission de contenu entre les utilisateurs (Lamprier *et al.*, 2015). Nous considérons alors ici une version légèrement modifiée d'IC, dans laquelle les délais d'infections suivent une loi uniforme.

Suivant un processus de diffusion itératif, le modèle IC considère que chaque utilisateur  $u$  nouvellement infecté dispose d'une chance *unique* d'infecter chaque autre utilisateur  $v$  non encore infecté avec une probabilité  $P_{u,v}$ . Dans ce contexte, la probabilité qu'un utilisateur  $v$  devienne infecté par n'importe quel utilisateur d'un ensemble "d'infecteurs potentiels"  $I \subseteq \mathcal{U}$  est définie selon :

$$P(v|I) = 1 - \prod_{u \in I} (1 - P_{u,v}) \quad [1]$$

où  $P_{u,v}$  est la probabilité de transmission d'information de  $u$  vers  $v$ .  $P(v|I)$  est donc la probabilité qu'au moins un utilisateur de  $I$  transmette l'information à  $v$ . Dans le modèle IC classique, chaque probabilité  $P_{u,v}$  est définie de manière indépendante. Nous proposons ici de définir ces probabilités selon une fonction  $f : \mathbb{R}^d \times \mathbb{R}^d \rightarrow [0, 1]$  considérant les positions relatives des utilisateurs dans l'espace de projection, avec  $d$  la taille de l'espace considéré :

$$P_{u,v} = f(z_u, \omega_v) \quad [2]$$

où  $z_u \in \mathbb{R}^d$  est la représentation de  $u$  en tant qu'*émetteur* de contenu et  $\omega_v \in \mathbb{R}^d$  celle de  $v$  en tant que *récepteur*. Deux ensembles de coordonnées sont donc définis pour chaque utilisateur :  $Z = (z_u)_{u \in \mathcal{U}}$  et  $\Omega = (\omega_u)_{u \in \mathcal{U}}$ . Avec un seul ensemble, les probabilités de transmissions seraient symétriques, ce qui n'est pas réaliste dans le cadre de la diffusion sur les réseaux sociaux.

La fonction  $f$  peut être définie de multiples façons, la seule contrainte étant qu'elle doive fournir une valeur dans  $[0, 1]$  de manière à ce qu'elle puisse être interprétée comme une probabilité. Nous proposons de considérer la fonction logistique suivante :

$$f(z_u, \omega_v) = \frac{1}{1 + \exp(z_u^{(0)} + \omega_v^{(0)} + \sum_{i=1}^{d-1} (z_u^{(i)} - \omega_v^{(i)})^2)} \quad [3]$$

où  $x^{(i)}$  est la  $i$ -ème composante du vecteur  $x$ . Le choix d'une fonction logistique paraît pertinent puisque ce genre de fonction permet de définir des probabilités décroissantes en fonction de la distance qui sépare les représentations émetteur  $Z$  et récepteur  $\Omega$  dans l'espace de projection. D'autre part, de par sa forme en S, l'utilisation de cette fonction implique un impact plus important pour



les variations survenant sur les distances modérées, tombant dans la partie de plus forte pente de la fonction. Cela permet de focaliser l'attention sur les influences moins évidentes lors de l'apprentissage. À noter également que la fonction ainsi définie considère la première dimension de chaque représentation comme une valeur de biais :  $z_u^{(0)}$  et  $\omega_v^{(0)}$  modélisent respectivement un a priori sur la tendance générale de  $u$  à transmettre de l'information et un a priori sur la tendance générale de  $v$  à en recevoir.

### 3.3. Algorithme d'apprentissage

Selon l'ensemble de probabilités de transmission  $\mathcal{P} = \{P_{u,v} | (u,v) \in \mathcal{U}^2\}$ , la vraisemblance d'un épisode de diffusion  $D$  donné peut s'écrire (si l'on se focalise sur les ordres partiels de contamination plutôt que sur les instants d'infection exacts (Lamprier *et al.*, 2015)) :

$$P(D) = \prod_{v \in D(\infty)} P_v^D \prod_{v \in \bar{D}(\infty)} (1 - P_v^D) \quad [4]$$

avec  $P_v^D = P(v | D(t^D(v)))$  la probabilité d'observer l'infection de l'utilisateur  $v$  connaissant l'ensemble d'utilisateurs infectés avant  $v$  dans  $D$ . Il est alors possible de considérer la log-vraisemblance d'un ensemble d'épisodes  $\mathcal{D}$  :

$$\mathcal{L}(\mathcal{P}; \mathcal{D}) = \sum_{D \in \mathcal{D}} \left( \sum_{v \in D(\infty)} \log(P_v^D) + \sum_{v \in \bar{D}(\infty)} \log(1 - P_v^D) \right) \quad [5]$$

La définition de  $P_v^D$  rend cette log-vraisemblance difficile à optimiser directement. Cependant, en nous basant sur (Saito *et al.*, 2008), nous pouvons définir un algorithme de type Espérance-Maximisation (EM), basé sur la fonction d'espérance suivante :

$$\mathcal{Q}(\mathcal{P} | \hat{\mathcal{P}}) = \sum_{D \in \mathcal{D}} \left( \Phi^D(\mathcal{P} | \hat{\mathcal{P}}) + \sum_{v \in \bar{D}(\infty)} \sum_{u \in D(\infty)} \log(1 - P_{u,v}) \right) \quad [6]$$

où  $\Phi^D(\mathcal{P} | \hat{\mathcal{P}})$  correspond à l'espérance de la valeur du premier terme de la fonction de log-vraisemblance de  $D$  selon les paramètres courants  $\hat{\mathcal{P}}$ . Cette valeur est calculée en considérant les probabilités conditionnelles de succès des différentes tentatives de transmission du contenu diffusé : Sachant que  $v$  est infecté dans  $D$  avec une probabilité  $\hat{P}_v^D$  (calculée selon l'équation 1 avec l'estimation courante des probabilités de transmission), la probabilité conditionnelle que la transmission de  $u$  vers  $v$  ait eu lieu est donnée par  $\hat{P}_{u,v} / \hat{P}_v^D$ .

$$\Phi^D(\mathcal{P} | \hat{\mathcal{P}}) = \sum_{v \in D(\infty)} \sum_{u \in D(t^D(v))} \frac{\hat{P}_{u,v}}{\hat{P}_v^D} \log(P_{u,v}) + \left(1 - \frac{\hat{P}_{u,v}}{\hat{P}_v^D}\right) \log(1 - P_{u,v}) \quad [7]$$

Du fait de l'utilisation d'un espace de représentation vectoriel, les probabilités de transmission entre chaque paire d'utilisateurs ne sont plus libres. Les contraintes géométriques liées à l'espace rendent leurs valeurs interdépendantes. Maximiser une telle log-vraisemblance ne peut alors plus se décomposer en un ensemble de sous-problèmes convexes comme cela peut être le cas avec un modèle IC classique. Néanmoins, il est possible de définir une procédure de montée de gradient stochastique convergeant vers un maximum local efficace, lorsque l'on détermine un nombre de dimensions adapté (25 dimensions semble être un bon compromis, cf. section 4).

L'algorithme 1 détaille la procédure utilisée pour apprendre  $Z$  et  $\Omega$ . Il s'agit d'un algorithme EM où l'étape de maximisation effectue un pas de gradient. Une itération se déroule ainsi :

- 1) Ligne 7 : tirage uniforme d'un épisode de diffusion  $D$  et d'un utilisateur  $v$  n'étant pas la source de  $D$  ;
- 2) Lignes 9 à 15 : si  $v$  fait partie de  $D$ , calcul des estimations courantes  $\hat{P}_v^D$  et  $\hat{P}_{u,v}$  pour chaque utilisateur  $u$  infecté avant  $v$  (selon les formules 1 et 2 en utilisant les valeurs courantes de  $z_u$  et  $\omega_v$ ) ;
- 3) Lignes 16 à 25 : mise à jour des valeurs de  $Z$  et  $\Omega$  de façon à augmenter la valeur de  $\mathcal{Q}(\mathcal{P}|\hat{\mathcal{P}})$ . Si  $v$  est infecté dans  $D$ , cette mise à jour fait intervenir la dérivée de  $\Phi^D(\mathcal{P}|\hat{\mathcal{P}})$  (lignes 19 à 22). Sinon, elle fait intervenir la dérivée de  $\log(1 - P_{u,v})$  (lignes 22 à 24). Le pas d'apprentissage  $\epsilon$  est fixé à  $10^{-4}$ .
- 4) Lignes 27 à 34 : test de convergence toutes les freq itérations (1000000 dans notre cas). Le processus s'arrête si la log-vraisemblance sur  $\mathcal{D}$  n'a pas significativement augmenté depuis le dernier test.

Notons enfin qu'à cause du tirage effectué à la ligne 7, les différentes paires  $(u, v)$  ne sont pas associées aux épisodes de  $\mathcal{D}$  dans les mêmes proportions que dans la formule 6 : les paires d'utilisateurs apparaissant dans les épisodes plus courts sont tirées plus souvent. Nous calculons donc, en ligne 8, un poids  $\alpha$  permettant de corriger ce biais.

$$\alpha = \frac{|\mathcal{D}| \times |\bar{D}(1)|}{\sum_{D \in \mathcal{D}} \sum_{u \in D(\infty)} |\bar{D}(t^D(u) + 1)|}$$

Ce poids est considéré lors de la mise à jour des paramètres (lignes 16 à 25).

## 4. Expériences

### 4.1. Jeux de données

Dans nos expériences, les jeux de données suivants sont utilisés :

– **Digg** : corpus constitué à partir du portail d'information collaboratif *Digg*, sur lequel les utilisateurs postaient des messages auxquels d'autres utilisateurs

---

**Algorithme 1** : Apprentissage du modèle IC projeté

---

**Entrées :**

$\mathcal{U}$  : l'ensemble des utilisateurs;  $\mathcal{D}$  : l'ensemble des épisodes de diffusion;  
 $d$  : le nombre de dimensions;  $\epsilon$  : le pas d'apprentissage;  
 $freq$  : la fréquence des tests de convergence;

**Sorties :**

$Z = \{\forall u \in \mathcal{U} : z_u \in \mathbb{R}^d\}$ ;  $\Omega = \{\forall u \in \mathcal{U} : \omega_u \in \mathbb{R}^d\}$ ;

```

1  $nbProbas \leftarrow \sum_{D \in \mathcal{D}} \sum_{u \in D(\infty)} |\bar{D}(t^D(u) + 1)|$ ;
2 pour  $u \in \mathcal{U}$  faire
3   Tirage uniforme de  $z_u \in [-1, 1]^d$ ;   Tirage uniforme de  $\omega_u \in [-1, 1]^d$ ;
4 fin
5  $oldL \leftarrow -\infty$ ;    $it \leftarrow 0$ ;
6 tant que true faire
7   Tirage uniforme de  $D \in \mathcal{D}$  et  $v \in \bar{D}(1)$ ;
8    $\alpha \leftarrow |\mathcal{D}| \times |\bar{D}(1)| / nbProbas$ ;
9   si  $t^D(v) < \infty$  alors
10     $\hat{P}_v^D \leftarrow 1$ ;
11    pour  $u \in D(t^D(v))$  faire
12       $\hat{P}_{u,v} \leftarrow f(z_u, \omega_v)$ ;    $\hat{P}_v^D \leftarrow \hat{P}_v^D \times (1 - \hat{P}_{u,v})$ ;
13    fin
14     $\hat{P}_v^D \leftarrow 1 - \hat{P}_v^D$ ;
15  fin
16  pour  $u \in D(t^D(v))$  faire
17     $\xi_u^+ \leftarrow \frac{\partial \log f(z_u, \omega_v)}{\partial z_u}$ ;    $\xi_u^- \leftarrow \frac{\partial \log(1 - f(z_u, \omega_v))}{\partial z_u}$ ;
18     $\xi_v^+ \leftarrow \frac{\partial \log f(z_u, \omega_v)}{\partial \omega_v}$ ;    $\xi_v^- \leftarrow \frac{\partial \log(1 - f(z_u, \omega_v))}{\partial \omega_v}$ ;
19    si  $t^D(v) < \infty$  alors
20       $z_u \leftarrow z_u + \alpha \times \epsilon \times \left( \frac{\hat{P}_{u,v}}{\hat{P}_v^D} \xi_u^+ + \left(1 - \frac{\hat{P}_{u,v}}{\hat{P}_v^D}\right) \xi_u^- \right)$ ;
21       $\omega_v \leftarrow \omega_v + \alpha \times \epsilon \times \left( \frac{\hat{P}_{u,v}}{\hat{P}_v^D} \xi_v^+ + \left(1 - \frac{\hat{P}_{u,v}}{\hat{P}_v^D}\right) \xi_v^- \right)$ ;
22    sinon
23       $z_u \leftarrow z_u + \alpha \times \epsilon \times \xi_u^-$ ;    $\omega_v \leftarrow \omega_v + \alpha \times \epsilon \times \xi_v^-$ ;
24    fin
25  fin
26   $it \leftarrow it + 1$ ;
27  si  $it \bmod freq = 0$  alors
28     $\mathcal{P} \leftarrow \{P_{u,v} | (u, v) \in \mathcal{U}^2 \wedge P_{u,v} = f(z_u, \omega_v)\}$ ;
29     $L \leftarrow$  Calcul de la log-vraisemblance selon (5) avec  $\mathcal{P}$ ;
30    si  $L \leq oldL$  alors
31      retourner  $(Z, \Omega)$ ;
32    fin
33     $oldL \leftarrow L$ ;
34  fin
35 fin

```

---

| Corpus      | Util. | Liens  | Densité | $ \mathcal{D} $ Appr. | $ \mathcal{D} $ Test | Taille Episode Moy. |
|-------------|-------|--------|---------|-----------------------|----------------------|---------------------|
| Irvine      | 847   | 74871  | 0.1     | 433                   | 49                   | 14.6                |
| Icwsn       | 2270  | 4775   | 0.001   | 19027                 | 1000                 | 2.22                |
| Memetracker | 498   | 229073 | 0.9     | 10000                 | 1000                 | 2.17                |
| Digg        | 3295  | 689416 | 0.06    | 17000                 | 1000                 | 2.43                |
| Twitter     | 2841  | 884832 | 0.09    | 10000                 | 1000                 | 20.5                |
| LastFm      | 986   | 708159 | 0.72    | 10000                 | 1000                 | 7.25                |

**Tableau 1.** *Quelques statistiques sur les jeux de données.*

pouvaient ensuite attribuer des “Diggs”. Chaque épisode regroupe le message initial et l’ensemble des “Diggs” correspondants.

– **Lastfm** : corpus, issu d’un site d’écoute de musique en streaming, collecté pendant un an par (Celma, 2010). Chaque épisode regroupe les événements d’écoute d’un morceau.

– **Irvine** : corpus présenté dans (Opsahl et Panzarasa, 2009) regroupant les participations d’étudiants de l’université d’Irvine à des forums en ligne. Chaque épisode regroupe l’ensemble des participations à un forum particulier.

– **Twitter** : corpus correspondant à l’activité sur Twitter d’un ensemble d’environ 5000 utilisateurs durant 2 semaines. Chaque épisode regroupe l’ensemble des utilisations d’un hashtag particulier.

– **Memetracker** : corpus contenant des millions d’articles de blogs (Leskovec *et al.*, 2009). Chaque épisode regroupe les utilisations d’une des expressions récurrentes extraites par (Leskovec *et al.*, 2009).

– **ICWSM** : corpus contenant plusieurs millions de posts de blogs collectés en 2009. À la manière de (Gomez-Rodriguez *et al.*, 2011), chaque épisode correspond à une composante connexe du graphe des citations entre messages.

Comme expliqué en section 3.2, nous ne disposons *pas* de graphe social sur lequel apprendre les modèles. Pour les modèles de référence basés sur un graphe, une possibilité serait de considérer le graphe complet reliant toutes les paires d’utilisateurs du réseau considéré. Néanmoins, pour réduire la complexité, nous limitons grandement le nombre de liens considérés lors de l’apprentissage en nous appuyant sur le graphe de “co-participations” aux épisodes de diffusion : un lien  $(u, v)$  est envisagé si et seulement si il existe au moins un épisode de diffusion  $D$  contenant  $u$  et  $v$ , avec  $t^D(u) < t^D(v)$ . À noter que cela aboutit pour les modèles de référence envisagés (voir ci-après), après convergence de l’apprentissage, au même ensemble de probabilités que si nous avions considéré un graphe complet, car les liens sans exemples de diffusion obtiennent une probabilité de transmission nulle.

La table 1 donne quelques statistiques sur les jeux de données utilisés : nombre d’utilisateurs, nombre de liens et densité du graphe de co-participation, nombre d’épisodes en apprentissage et en test et enfin taille moyenne des épisodes de diffusion.

## 4.2. Modèles de référence

Nous comparons notre modèle aux approches suivantes :

- **IC** : version du modèle *Independent Cascades* décrite dans (Lamprier *et al.*, 2015).
- **Netrate** : modèle génératif définissant une distribution des délais de transmission sur chaque lien du graphe social de façon à reproduire les temps d’infection observés (Gomez-Rodriguez *et al.*, 2011).
- **CTIC** : extension du modèle IC, dans laquelle chaque lien est associé à deux paramètres contrôlant la probabilité et le délai de transmission entre ses utilisateurs.

## 4.3. Prédiction de diffusion

La principale tâche sur laquelle nous comparons les modèles pour en évaluer les performances est une tâche en prédiction de diffusion : le but est de retrouver, à partir d’une ou plusieurs sources  $D(1)$  d’un épisode de diffusion  $D$ , l’ensemble des utilisateurs finalement infectés  $D(\infty)$ . Du fait de la complexité induite par la présence probable de cycles dans les graphes de diffusion appris, il est impossible de calculer directement la probabilité d’infection finale de chaque utilisateur pour un épisode donné. La procédure d’inférence emploie alors un processus de Monte-Carlo pour estimer ces probabilités : à partir d’un ensemble d’utilisateurs initial  $D(1)$ , nous simulons 1000 processus de diffusion selon chaque modèle, et estimons ainsi pour chaque utilisateur sa probabilité  $P_u^D$  d’appartenir à  $D(\infty)$ .

Les modèles sont ensuite évalués selon plusieurs mesures :

- **MSE** : les probabilités d’infection prédites sont comparées aux vraies valeurs (0 ou 1) avec une mesure d’erreur quadratique.
- **Log-Vraisemblance** : la log-vraisemblance de l’ensemble des épisodes de diffusion de test selon les différents modèles. Les probabilités sont projetées sur l’intervalle  $[10^{-5}, 1 - 10^{-5}]$  pour éviter  $\log(0)$
- **MAP** : les utilisateurs sont classés par ordre décroissant de probabilités d’infection, et la liste est évaluée par Mean-Average-Precision.
- **F1** : la mesure classique  $F1$ , définie comme suit :

$$p = \frac{\sum_{u \in \mathcal{D}(\infty)} P_u^D}{\sum_{u \in \mathcal{U}} P_u^D} \quad r = \frac{\sum_{u \in \mathcal{D}(\infty)} P_u^D}{|\mathcal{D}(\infty)|} \quad F1 = \frac{2 * p.r}{p + r}$$

Avant de comparer notre modèle à ceux présentés ci-dessus, nous étudions l’impact du nombre de dimensions sur la durée de l’apprentissage (sur un ordinateur de bureau équipé d’un processeur Intel(R) Core(TM) i7 CPU 950@3.07GHz) et sur les performances du modèle. Nous ne reportons en figure 2 que les résultats obtenus sur Digg, mais les tendances observées sur les autres

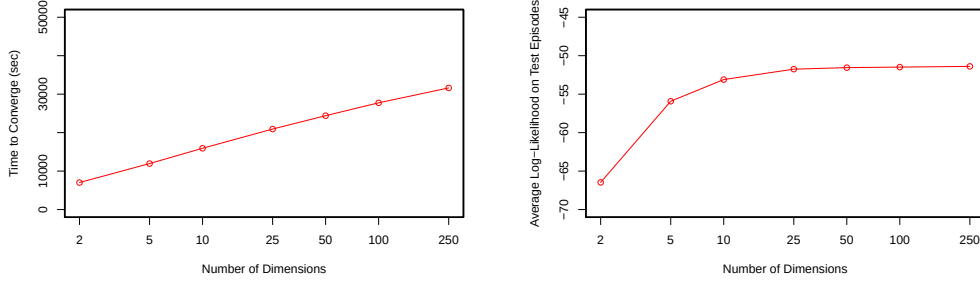


**Figure 2.** *Durée (en secondes) de l'apprentissage de notre modèle jusqu'à convergence, et log-vraisemblance obtenue en test, sur le jeu de données Digg.*

jeux de données sont similaires. Nous remarquons que la durée de l'apprentissage augmente logarithmiquement avec le nombre de dimensions, mais que la qualité du modèle augmente peu au delà de 25 dimensions. Dans la suite, nous utilisons donc un espace de dimension  $d = 25$  pour l'apprentissage.

Les résultats sont moyennés et regroupés dans le tableau 2. Nous pouvons tout d'abord remarquer que notre modèle (Embedded IC) obtient des résultats toujours *au moins aussi bons* que ceux des autres modèles pour une complexité bien inférieure. Les résultats assez mauvais obtenus par CTIC et NetRate peuvent être expliqués par le fait qu'il est difficile d'extraire des régularités sur les délais de transmission. Les bons résultats obtenus par notre approche montrent que nous pouvons projeter correctement les probabilités de transmission dans un espace vectoriel. Il est important de remarquer que cela se fait avec un nombre de paramètres (colonne de droite) largement inférieur au modèle IC, sauf sur le corpus ICWSM, dont le graphe de co-participations est très creux. Nous remarquons également que les mesures se comportent différemment :

- La MSE de notre modèle est la meilleure sur Irvine, Digg et LastFM. Sur ces jeux de données, la diffusion est en fait délicate à caractériser : le fait que deux utilisateurs interagissent avec le même item ne veut pas forcément dire qu'il y a eu diffusion de l'un à l'autre, il peut s'agir uniquement d'une corrélation sur leurs centres d'intérêts. Notre approche par apprentissage de représentations continues parvient à mieux identifier les similarités entre utilisateurs, ce qui conduit à une meilleur MSE (voir section 4.4).

- La log-vraisemblance de notre modèle est meilleure sur Irvine, ICWSM, Digg et Twitter. Ces jeux de données ont en commun qu'ils possèdent un graphe de co-participations bien moins dense que les autres corpus (table 1). Cela implique qu'un modèle classique, basé sur le graphe, ne pourra apprendre qu'un petit nombre de probabilités, et sera donc incapable de prédire la transmission d'information entre la plupart des utilisateurs. Notre approche, en revanche, est capable d'étendre des observations de transmission à des couples non observés.

| Corpus      | Modèle      | MSE           | LogVrai.        | MAP           | F1             | nbParams |
|-------------|-------------|---------------|-----------------|---------------|----------------|----------|
| Irvine      | IC          | 15,31         | -960,5          | 0,079         | 0,020          | 74871    |
|             | NetRate     | 15,42         | -892,13         | 0,078         | 0,019          | 74871    |
|             | CTIC        | 15,29         | -771,42*        | <b>0,080</b>  | 0,020          | 149742   |
|             | Embedded IC | <b>14,53*</b> | <b>-532,5*</b>  | 0,079         | <b>0,025*</b>  | 42350    |
| ICWSM       | IC          | 0,2           | -8,3            | 0,77          | <b>0,651</b>   | 4775     |
|             | NetRate     | 0,23          | -9,01           | 0,72          | 0,357          | 4775     |
|             | CTIC        | 0,22          | -8,46           | 0,76          | 0,482          | 9550     |
|             | Embedded IC | <b>0,19</b>   | <b>-6,14*</b>   | <b>0,78</b>   | <b>0,651</b>   | 113500   |
| MemeTracker | IC          | 32,62         | -795,85         | 0,22          | 0,0585         | 229073   |
|             | NetRate     | 34,55         | -850,48         | 0,17          | 0,0442         | 229073   |
|             | CTIC        | 33,27         | -802,52         | 0,22          | 0,0551         | 458146   |
|             | Embedded IC | <b>32,15</b>  | <b>-791,3</b>   | <b>0,23</b>   | <b>0,0632*</b> | 24900    |
| Digg        | IC          | 2,1           | -69,5           | 0,411         | <b>0,201</b>   | 689416   |
|             | NetRate     | 1,95          | -64,01*         | 0,409         | 0,199          | 689416   |
|             | CTIC        | 1,92*         | -64,18*         | 0,413         | <b>0,201</b>   | 1378832  |
|             | Embedded IC | <b>1,79*</b>  | <b>-51,75*</b>  | <b>0,434*</b> | 0,198          | 164750   |
| Twitter     | IC          | 6,70          | -412,75         | 0,047         | 0,012          | 884832   |
|             | NetRate     | 6,91          | -428,78         | 0,039         | 0,011          | 884832   |
|             | CTIC        | 6,72          | -401,56         | 0,049         | 0,012          | 1769664  |
|             | Embedded IC | <b>5,47*</b>  | <b>-223,15*</b> | <b>0,056*</b> | <b>0,013</b>   | 142050   |
| LastFM      | IC          | 12,13         | -409,5          | 0,132         | 0,026          | 708159   |
|             | NetRate     | 13,91         | -413,02         | 0,112         | 0,022          | 708159   |
|             | CTIC        | 12,12         | -409,3          | 0,128         | 0,025          | 1416318  |
|             | Embedded IC | <b>11,62*</b> | <b>-405</b>     | <b>0,151*</b> | <b>0,027</b>   | 49300    |

**Tableau 2.** Résultats obtenus sur les différents corpus. Les valeurs marquées d’un astérisque sont significativement meilleures que celles obtenues par IC (Test-t de Student 95%), et celles en gras indiquent le meilleur résultat obtenu sur chaque corpus.

– Enfin, notre modèle obtient une meilleure MAP sur Digg, Twitter et LastFM. Ces sites web ont également une propriété particulière : ils fonctionnent comme des “portails”. Quand un utilisateur se connecte sur un de ces sites, la première chose qu’il trouve est une liste d’items populaires (même Twitter dispose d’une section “Trends”). Les interactions utilisateurs-utilisateurs sont ainsi moins importantes que dans les autres jeux de données. Ainsi, sur ces sites, la diffusion a lieu de manière plus globale : plus une information est populaire, plus elle a de chances d’infecter de nouveaux utilisateurs. Ce phénomène est mieux représenté par notre modèle, où chaque utilisateur a au moins une petite chance d’infecter tous les autres.

Les résultats obtenus nous apprennent que la diffusion d’information est un phénomène complexe et protéiforme. Les jeux de données collectés ici sont de tailles et de densités très différentes, et chacun possède son propre “type” de diffusion (retweets, discussion, recommandation). Cela explique pourquoi les modèles se comportent aussi différemment sur chaque corpus. Néanmoins, d’une façon générale les résultats mettent en avant les propriétés avantageuses

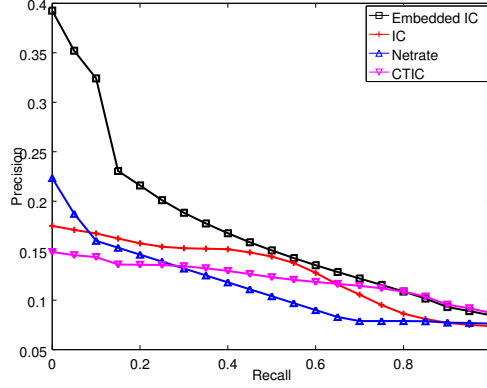


**Figure 3.** *Précision-Rappel de la detection de relations sur Memetracker.*

de notre approche : moins de paramètres à apprendre, une meilleure généralisabilité et une meilleure capacité à modéliser différents types de diffusion.

#### 4.4. Détection de Relations d'Influence

Pour compléter les expérimentations menées, nous évaluons notre modèle sur sa capacité à retrouver un ensemble de relations de diffusion connues à partir d'un ensemble  $\mathcal{D}$  d'épisodes de diffusion observés. À l'instar de (Gomez-Rodriguez *et al.*, 2011 ; Gomez Rodriguez *et al.*, 2010), l'idée est d'utiliser les hyperliens liant des messages du corpus memetracker comme l'ensemble de relations à identifier. La présence d'un hyperlien entre deux messages rend en effet compte de l'influence qu'a eu le message pointé sur celui qui le référence, et donc de l'influence de l'auteur du premier sur l'auteur du second.

Pour chaque paire  $(u, v)$ , la probabilité  $P_{u,v}$  apprise selon chacun des modèles est interprétée comme la probabilité d'existence d'un lien entre ces deux utilisateurs. Nous trions les probabilités ainsi apprises par chaque modèle et évaluons les résultats avec une courbe de Précision-Rappel. Les résultats sont présentés en figure 3. Cette expérience illustre la meilleure capacité de notre modèle à inférer des liens d'influence à partir d'épisodes de diffusion, sa courbe de precision en fonction du rappel se situant clairement au dessus de celle des autres modèles. Le fait de projeter les utilisateurs dans un espace vectoriel pour définir les probabilités de transmission permet au modèle de découvrir des relations sans exemple de diffusion dans les épisodes observés. En outre, la régularisation induite par les contraintes géométriques de l'espace permet de minorer la probabilité de certains liens non pertinents, grâce notamment à l'effet repoussoir de certains utilisateurs projetés proches de leur utilisateur destination selon la représentation récepteur.



## 5. Conclusion

La projection d'un graphe de relations sociales dans un espace vectoriel permet d'en extraire des régularités, afin de découvrir de nouvelles relations ou au contraire d'identifier des relations peu pertinentes. Cela peut s'avérer très bénéfique pour des tâches autour de la diffusion d'information, puisque permettant de se concentrer sur les canaux principaux de communication du réseau. De plus, l'utilisation d'un espace vectoriel réduit l'impact de la présence importante de bruit dans les données sociales collectées, ainsi que le nombre de paramètres à apprendre et à stocker pour les réseaux denses. Dans cet esprit, notre proposition d'apprendre les probabilités du modèle *Independent Cascades* par projection des utilisateurs dans un espace de représentation nous a permis d'observer de bonnes performances pour les tâches de prédiction de diffusion et de découverte de relations d'influence.

La piste principale de poursuite du travail mené concerne la considération de l'objet des diffusions dans l'apprentissage du modèle. Nous envisageons d'apprendre des opérateurs de transformation de l'espace de représentation en fonction du contenu diffusé. D'autre part, nous envisageons l'application du processus d'apprentissage proposé ici dans un cadre "en ligne", de manière à capturer les évolutions du réseau d'influence au cours du temps.

## Remerciements

Ce travail a été effectué avec le support des projets REQUEST (Investissement d'avenir, 2014-2017) et LOCUST (ANR, 2015-2018).

## 6. Bibliographie

- Bakshy E., Rosenn I., Marlow C., Adamic L., « The role of social networks in information diffusion », *WWW'12*, ACM, p. 519-528, 2012.
- Barbieri N., Bonchi F., Manco G., « Cascade-based Community Detection », *WSDM'13*, ACM, p. 33-42, 2013.
- Bengio Y., Courville A. C., Vincent P., « Representation Learning : A Review and New Perspectives », *IEEE TPAMI*, vol. 35, n° 8, p. 1798-1828, 2013.
- Bordes A., Usunier N., García-Durán A., Weston J., Yakhnenko O., « Translating Embeddings for Modeling Multi-relational Data », *NIPS'13*, p. 2787-2795, 2013.
- Bourigault S., Lagnier C., Lamprier S., Denoyer L., Gallinari P., « Learning Social Network Embeddings for Predicting Information Diffusion », *WSDM'14*, ACM, p. 393-402, 2014.
- Bourigault S., Lamprier S., Gallinari P., « Representation Learning for Information Diffusion through Social Networks : an Embedded Cascade Model », *WSDM'16*, ACM, p. 573-582, 2016.
- Celma O., *Music Recommendation and Discovery in the Long Tail*, Springer, 2010.

- Chen S., Moore J. L., Turnbull D., Joachims T., « Playlist prediction via metric embedding », *KDD'12*, ACM, p. 714-722, 2012.
- Cheng J., Adamic L., Dow P. A., Kleinberg J. M., Leskovec J., « Can Cascades Be Predicted ? », *WWW'14*, ACM, p. 925-936, 2014.
- Goldenberg J., Libai B., Muller E., « Talk of the network : A complex systems look at the underlying process of word-of-mouth », *Marketing letters*, vol. 12, n° 3, p. 211-223, 2001.
- Gomez-Rodriguez M., Balduzzi D., Schölkopf B., « Uncovering the Temporal Dynamics of Diffusion Networks », *ICML'11*, ACM, p. 561-568, 2011.
- Gomez Rodriguez M., Leskovec J., Krause A., « Inferring networks of diffusion and influence », *KDD'10*, ACM, 2010.
- Gruhl D., Guha R., Liben-Nowell D., Tomkins A., « Information Diffusion Through Blogspace », *WWW'04*, ACM, p. 491-501, 2004.
- Guille A., Hacid H., « A predictive model for the temporal dynamics of information diffusion in online social networks », *WWW Companion*, ACM, 2012.
- Kempe D., Kleinberg J., Tardos E., « Maximizing the spread of influence through a social network », *KDD'03*, ACM, p. 137-146, 2003.
- Lagnier C., Denoyer L., Gaussier E., Gallinari P., « Predicting Information Diffusion in Social Networks using Content and User's Profiles », *ECIR '13*, 2013.
- Lamprier S., Bourigault S., Gallinari P., « Extracting Diffusion Channels from Real-World Social Data : a Delay-Agnostic Learning of Transmission Probabilities », *ASONAM'15*, IEEE Computer Society, 2015.
- Leskovec J., Backstrom L., Kleinberg J., « Meme-tracking and the dynamics of the news cycle », *KDD'09*, ACM, p. 497-506, 2009.
- Mikolov T., Sutskever I., Chen K., Corrado G. S., Dean J., « Distributed Representations of Words and Phrases and their Compositionality », *NIPS'13*, 2013.
- Mislove A., Marcon M., Gummadi K. P., Druschel P., Bhattacharjee B., « Measurement and Analysis of Online Social Networks », *IMC '07*, ACM, p. 29-42, 2007.
- Najar A., Denoyer L., Gallinari P., « Predicting information diffusion on social networks with partial knowledge », *WWW '12 Companion*, ACM, p. 1197-1204, 2012.
- Opsahl T., Panzarasa P., « Clustering in weighted networks », *Social networks*, vol. 31, n° 2, p. 155-163, 2009.
- Saito K., Kimura M., Ohara K., Motoda H., « Learning Continuous-Time Information Diffusion Model for Social Behavioral Data Analysis », *ACML '09*, Springer-Verlag, p. 322-337, 2009.
- Saito K., Nakano R., Kimura M., « Prediction of Information Diffusion Probabilities for Independent Cascade Model », *KES '08*, Springer-Verlag, p. 67-75, 2008.
- Saito K., Ohara K., Yamagishi Y., Kimura M., Motoda H., « Learning Diffusion Probability Based on Node Attributes in Social Networks. », *ISMIS'11*, Springer, p. 153-162, 2011.
- Ver Steeg G., Galstyan A., « Information-theoretic measures of influence based on content dynamics », *WSDM '13*, ACM, p. 3-12, 2013.
- Yang J., McAuley J., Leskovec J., « Detecting Cohesive and 2-mode Communities Indirected and Undirected Networks », *WSDM '14*, ACM, p. 323-332, 2014.

---

# Suggestion d'experts pour renouveler le comité de programme d'une conférence

**Hong Diep Tran, Guillaume Cabanac, Gilles Hubert**

*IRIT UMR 5505 CNRS, Université Paul Sabatier – Toulouse 3  
{hong-diep.tran, guillaume.cabanac, gilles.hubert}@irit.fr*

---

**RÉSUMÉ.** *Le processus d'évaluation par les pairs permet de valider les progrès scientifiques communiqués dans des articles de recherche. Cette grande responsabilité repose sur les comités éditoriaux des journaux, sur les comités de programme des conférences et sur chacun de leurs membres. De plus, avec un grand nombre de conférences scientifiques organisées chaque année, la recherche d'experts pour participer au comité de programme devient une tâche fréquente et coûteuse. Dans cet article, nous proposons une modélisation d'expert basée sur différentes preuves d'expertise, notamment sur les citations, pour émettre des suggestions d'experts dans le cas d'une recherche de membres pour renouveler le comité de programme d'une conférence.*

**ABSTRACT.** *The peer review process enables the scientific community to validate the new knowledge documented in research activities. It relies on the editorial boards of academic journals, on the program committees of conferences, and their members. In addition, with a large number of scientific conferences held each year, searching for experts that would be invited to join program committees is an increasingly hard task. In this paper, we propose an expert modeling method based on different features of expertise — in particular on citation networks — in order to issue suggestions for renewing the members of conference program committees.*

**MOTS-CLÉS :** *Recherche d'experts, graphe de citations, comité de programme.*

**KEYWORDS:** *Expert retrieval, citation graph, program committee.*

---

## 1. Introduction

La recherche d'experts est une problématique posée depuis une quinzaine d'années (Balog *et al.*, 2012). Des systèmes automatiques ont été proposés pour répondre à ce problème dans de nombreux cas spécifiques (Fang et Zhai, 2007 ; Rodriguez et Bollen, 2008 ; Afzal et Maurer, 2011), bien qu'ils ne semblent pas, en réalité, largement utilisés. En parallèle au fort développement de la science et de la technologie, un grand nombre de conférences sont organisées annuellement afin de diffuser les découvertes scientifiques et de les partager avec la communauté scientifique. Notre étude s'intéresse à l'une des tâches importantes de l'organisation d'une conférence : trouver des experts pour renouveler son comité de programme. La réalisation de cette tâche, basée habituellement sur les accointances entre chercheurs, est lourde et présente le risque d'oublier des experts qui pourraient être sollicités. Rodriguez et Bollen (2008) ont montré la nécessité d'automatiser cette tâche. De nombreuses études s'intéressent à la recherche d'experts pour des buts différents, par exemple trouver un relecteur pour un manuscrit soumis à un journal (Rodriguez et Bollen, 2008), suggérer des chercheurs correspondant à des intérêts scientifiques (Cabanac, 2011), trouver le directeur de thèse approprié pour un étudiant (Liu et Dew, 2004), ou encore trouver un professionnel de santé pour remplacer un absent en cas d'urgence (McDonald et Ackerman, 2000). Cependant, à notre connaissance, il n'existe pas d'approche relative à la recherche de membres de comité de programme.

Différents problèmes de recherche d'experts ont trouvé une réponse dans les systèmes de recherche d'information (Macdonald et Ounis, 2008 ; Serdyukov et Hiemstra, 2008), c'est-à-dire une recherche de profils thématiques d'experts à partir de la formulation d'un besoin d'expertise. Une difficulté qui influence la qualité de la recherche réside dans la détermination du besoin de l'utilisateur, davantage pour la recherche scientifique car ce besoin est souvent implicite (Balog *et al.*, 2012). Les profils des experts qui représentent le domaine, les compétences ou les intérêts scientifiques des individus sont déterminés à partir de diverses preuves d'expertise, telles que les activités professionnelles, les activités scientifiques ou encore les citations formulées par les chercheurs. Plusieurs études ont montré une bonne efficacité de systèmes de recherche basés sur les citations (Strohman *et al.*, 2007 ; De Bellis, 2009 ; Liang *et al.*, 2011) et l'apport de la combinaison avec d'autres traces d'activités scientifiques des chercheurs (Cabanac, 2011).

Les citations constituent un objet d'étude central en sociologie des sciences (Kessler, 1963 ; Small, 1973 ; Cronin, 1984). Cependant, saisir le sens exact de chaque citation de manière automatique reste un problème ouvert. À ce jour, peu de travaux ont utilisé les graphes de citations pour la recherche d'experts, la plupart se sont retraits aux tâches de recherche d'articles ou de documents scientifiques.

L'objectif de cet article vise la suggestion d'experts-candidats pour renouveler le comité de programme d'une conférence. Ce contexte diffère des précédents évoqués précédemment par le peu de contenu textuel décrivant une conférence. En effet, la description d'une conférence se limite en général à un ensemble de sujets dont les

grandes lignes sont évoquées par quelques mots-clés. Une recherche dans une base bibliographique à partir de ces mots-clés conduirait inévitablement à un résultat fortement « bruité ». Ainsi, l’approche proposée dans cet article se base sur des éléments autres que textuels. Notre contribution dans cet article est triple, en proposant :

- 1) une modélisation de l’espace de recherche se basant sur différents éléments (tels que la conférence, les publications, les experts-candidats) et les types de relations entre ces éléments (comme être auteur d’une publication, citer un article, participer à un comité de programme précédent),
- 2) une méthode de calcul de la proximité entre un expert-candidat et la conférence,
- 3) une série d’expérimentations pour valider notre approche.

Cet article est organisé comme suit. La section 2 présente un état de l’art relatif à la recherche d’expertise et à la recommandation d’experts. La section 3 introduit notre contribution concernant la modélisation de l’espace de recherche d’experts et la proximité entre expert-candidat et conférence. La section 4 décrit les expérimentations réalisées pour valider notre approche. Enfin, la section 5 conclut l’article et présente les perspectives à ce travail en lien avec la diversification des suggestions selon les caractéristiques des individus (genre, thème, localisation, ancienneté).

## 2. État de l’art

### 2.1. Recherche d’expertise

Selon Balog *et al.* (2012), la recherche d’experts comprend deux tâches principales : *profiler l’expert* et *trouver l’expert*. Nous les détaillons dans les sections suivantes.

#### 2.1.1. Profiler l’expert

La recherche d’experts est motivée par une première question : *quelle est l’expertise d’un chercheur* ? Une réponse à cette question consiste à profiler les experts, c’est-à-dire extraire les données appropriées indiquant le domaine et la thématique de la personne afin de les analyser et de les comparer. Il existe différents types de profils d’experts liés au type de preuve d’expertise utilisé. Par exemple, un profil thématique se basera sur le contenu textuel des publications alors qu’un profil social se basera sur les activités relationnelles en situation professionnelle de l’expert.

Le profil d’un expert est très souvent thématique (Balog *et al.*, 2012) et représenté sous forme de vecteur de termes représentant ses connaissances. Elles sont analysées suivant des éléments de connaissance élémentaires. Pour la recherche d’experts, il faut déterminer l’ensemble de ces connaissances élémentaires et mesurer ensuite le niveau d’expertise sur chaque connaissance élémentaire. Le nombre de connaissances utilisées influence la précision et l’efficacité du système. Trop peu de connaissances peut conduire à ne représenter que trop partiellement l’expertise de la personne. Trop de connaissances conduit à un nombre et une complexité de calculs trop importants. La

détermination de l'espace de recherche est également essentielle et repose généralement sur les bases bibliographiques. Cependant, ces données sont souvent hétérogènes et ne sont pas toujours disponibles dans tous les domaines scientifiques. Par exemple, Liang *et al.* (2011) mentionnent l'*ACL Anthology Network* comme seul jeu de données exploitable pour leur modèle de recommandation de publications.

Une fois la zone de recherche déterminée, la modélisation des experts et l'estimation du niveau d'expertise sur chaque aspect de connaissance constituent d'autres problématiques. Ces travaux dépendent du type de preuve d'expertise utilisé. Balog *et al.* (2012) présentent un panorama des différentes approches, chaque approche utilisant un ensemble spécifique de preuves et y appliquant une modélisation ainsi qu'une mesure de similarité ad hoc. De même, Cabanac (2011) utilise les titres d'articles comme preuve et modélise les experts sous forme de vecteurs de termes, le niveau d'expertise de l'expert sur chaque terme étant établi sur la base de la valeur *tf.idf*.

#### 2.1.2. Trouver l'expert

Une seconde question liée à la recherche d'experts est : *quels sont les experts dans ce domaine ?* Pour une thématique donnée, il faut estimer le niveau d'expertise des experts-candidats afin de les classer. Il existe un grand nombre de méthodes pour mesurer la force du lien entre une thématique et un expert-candidat. En particulier, on peut estimer le lien thématique par les interactions entre l'objet qui porte le thème donné et l'expert-candidat. Par exemple, si un chercheur a publié de nombreux articles dans une conférence, il est évident que ce chercheur travaille dans le domaine de cette conférence ; si un chercheur a participé à de nombreuses reprises à la conférence, c'est sûrement un expert du domaine de cette conférence.

Une première problématique concerne l'identification de preuves d'expertise de l'expert-candidat. Les preuves sont principalement liées aux publications scientifiques. Ce sont les articles dont l'expert est auteur, les articles qu'il a téléchargés, ceux qu'il a lus ou ceux qu'il a cités. Les activités professionnelles de l'expert sont d'autres preuves montrant efficacement le lien thématique entre experts. Il existe de nombreuses activités et liens professionnels qui peuvent être utilisés comme preuves, par exemple, les liens de collaboration entre collègues, les co-signatures d'articles, les participations à une même conférence ou une même manifestation scientifique, les liens de citation (un article cite l'autre), les liens de co-citation (deux articles citent un même troisième) et les liens de couplage bibliographique (deux articles sont cités par un même troisième), etc.

Une seconde problématique consiste à modéliser et mesurer la similarité thématique entre un thème donné et les experts-candidats. En fonction du type de preuve d'expertise, les approches appliquent une modélisation et une mesure appropriées. Une méthode largement utilisée pour la modélisation thématique repose sur le modèle vectoriel et les mesures de similarité associées. Plus récemment, des approches se sont intéressées aux modèles de graphes, où les objets de recherche et l'espace de recherche sont modélisés sous la forme d'un graphe. Déterminer la proximité thématique dans le graphe se ramène alors un problème de parcours de graphe. Strohman

*et al.* (2007) ainsi que Liang *et al.* (2011) définissent, par exemple, des modèles basés sur les graphes de citations. Cabanac (2011) propose un modèle basé sur le graphe des co-auteurs et le graphe des participants aux conférences.

## **2.2. Utilisation du graphe des citations pour la recherche d'expertise**

En réalisant une citation, un auteur exprime un lien conceptuel entre l'auteur, son article et l'article cité. Les citations peuvent donc être considérées comme une preuve d'expertise intéressante. Plusieurs approches ont ainsi cherché à analyser les relations de citation et à créer des bases de données de citations (Garfield, 1955 ; Nanba et Manabu, 1999 ; Tang *et al.*, 2008 ; Tang *et al.*, 2009 ; Huang et Qiu, 2010). Cependant, un problème est que les citations véhiculent des sens différents, et chaque sens de citation correspond à un niveau de lien entre l'article qui cite et l'article cité. Par exemple, Nanba et Manabu (1999), Tang *et al.* (2009) ainsi que Liang *et al.* (2011) extraient le contexte des citations pour classer les citations, alors que Huang et Qiu (2010) et Liang *et al.* (2011) s'intéressent à la sémantique des citations. En réalité, il existe un grand nombre de raisons de citer (Cronin, 1984). Extraire les citations dans le contenu de l'article et déterminer précisément leur raison d'être est très délicat.

L'exploitation de la relation de citation pour la recherche d'expertise passe généralement par la création d'un graphe de citations. Sur le graphe des citations, la proximité entre deux articles n'est pas seulement attestée par la citation directe entre ces deux articles, mais aussi par la citation indirecte. Clairement, deux articles proches sur un thème ne se citent pas forcément l'un l'autre (en particulier, un article antérieur à un autre ne peut mécaniquement pas le citer). Pour estimer la proximité entre les nœuds dans le graphe des citations, deux types d'approches se distinguent :

- *celles basées sur la proximité entre les nœuds* : c'est-à-dire basées sur la citation directe. Small (1973) examine la relation de co-citation : deux articles sont proches s'ils citent un même troisième article. De même, Kessler (1963) définit la relation de « couplage bibliographique » lorsque deux articles sont cités par un même troisième article. Cependant, ces deux cas exhibent une faiblesse illustrée dans l'exemple dans la Figure 1. Quand trois articles *d1*, *d2* et *d3* citent un même quatrième article, l'article *d1* est considéré en relation avec les articles *d2* et *d3*, mais on ne peut pas estimer si *d1* est plus proche de *d2* ou *d3* ;

- *celles basées sur la structure du graphe*, c'est-à-dire basées sur la force et le nombre de chemins entre les nœuds du graphe de citations. Liben-Nowell et Kleinberg (2003) ainsi que Liang *et al.* (2011) ont utilisé la distance de Katz (1953) pour estimer la pertinence entre deux nœuds. Dans le cas de Liben-Nowell et Kleinberg (2003), toutes les citations directes sont traitées de la même façon : toute citation est considérée de poids identique. Liang *et al.* (2011) considèrent, par contre, que les citations ont différentes forces. Ils déterminent une force de citation basée sur la classification des citations ; puis la pertinence entre deux nœuds est estimée sur la base de cette différence des citations. Cependant, comme évoqué précédemment, l'utilisation du contexte de citation pour estimer la force du lien nécessite des données qui sont

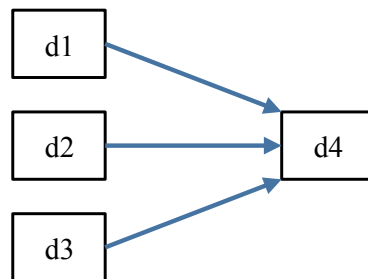


**Figure 1.** Proximité des nœuds *d1*, *d2* et *d3* basée sur la relation de co-citation.

difficiles à collecter à large échelle. Un autre problème relatif à la relation de citation entre deux articles concerne l'orientation des arcs : un article ne peut qu'en citer un autre qui a été publié avant. Elle est donc représentée par un arc (c.-à-d., orienté) dans le graphe. Selon Katz (1953), il existe une relation entre deux nœuds quand il y a au moins un chemin reliant l'un à l'autre. Des articles conceptuellement proches (par co-citation ou couplage bibliographique) ne sont cependant pas forcément reliés par un chemin (cf. Figure 2) et il serait préjudiciable d'ignorer cette proximité.

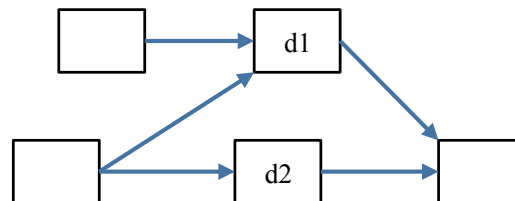


**Figure 2.** Exemple de documents proches *d1* et *d2* pour lesquels il n'existe pourtant pas de chemin entre les co-citations et les couplages bibliographiques.

### 2.3. Système de recommandation d'experts

Les systèmes de recommandation visent à suggérer des éléments pouvant intéresser un utilisateur sans qu'il ait à expliciter un besoin. Une problématique majeure concerne la détermination des intérêts de l'utilisateur. Concrètement, il s'agit de modéliser les éléments liés à l'utilisateur pour ensuite analyser ces éléments et construire un profil complet représentant les centres d'intérêt de l'utilisateur. Il existe une grande diversité d'approches de systèmes de recommandation. Cependant, deux approches sont principalement utilisées (Bobadilla *et al.*, 2013) : le *filtrage basé sur le contenu* et le *filtrage collaboratif*.

Le filtrage basé sur le contenu utilise les informations (généralement des termes) décrivant les éléments manipulés afin de calculer des similarités entre éléments et recommander des éléments candidats similaires à un élément initial. Par exemple, pour



recommander des articles scientifiques au regard du domaine d'expertise d'un chercheur donné, on cherchera la preuve d'expertise dans le contenu textuel des articles dont il est l'auteur (mais aussi éventuellement les articles qu'il a téléchargés, qu'il a lus, etc.) pour construire un profil et calculer ensuite une similarité entre le contenu d'autres d'articles et le profil du chercheur.

Différentes études ont porté sur la recherche des preuves d'expertise concernant les articles, comme les termes apparaissant dans ces articles. Ces termes sont extraits, par exemple, du contenu des articles, de leur titre (Lin et Wilbur, 2007 ; Lao et Cohen, 2010 ; Cabanac, 2011), de leur résumé (Ekstrand *et al.*, 2010), de leur introduction ou de leurs mots-clés (Huang *et al.*, 2002).

Cependant, il existe d'autres preuves d'expertise, sans contenu textuel, qui ne sont donc pas exploitables par ce type d'approche. Le filtrage basé sur le contenu est souvent combiné à un filtrage collaboratif pour améliorer le système de recommandation. Le filtrage collaboratif a été popularisé par Goldberg *et al.* (1992), et depuis, de nombreux modèles utilisent ce principe. Il s'agit de déterminer les éléments similaires à un élément initial par l'intermédiaire d'éléments tiers. Les preuves d'expertise utilisées par ce type d'approche sont alors les activités professionnelles, les activités scientifiques des éléments initiaux que sont les individus, comme les activités de citation (Liang *et al.*, 2011), de co-signature et de co-participation à des conférences (Cabanac, 2011).

### **3. Suggestion de membres pour constituer un comité de programme**

Suggérer des experts pour renouveler un comité de programme requiert de modéliser les preuves d'expertise des scientifiques de la communauté. Il existe par exemple des preuves spécifiques, comme la participation de l'expert aux comités de programme passés. Nous proposons un modèle de suggestion d'experts basé sur diverses preuves. Les différences entre notre modèle et les travaux existants sont les suivantes :

- au niveau de la modélisation, le graphe de citation est intégré dans un graphe plus riche, où non seulement les articles sont modélisés en tant que nœuds, mais aussi les experts-candidats et la conférence donnée. De même, les liens entre les nœuds représentent différents types de relations ;
- différentes définitions sont proposées pour estimer la force des relations variées dans le graphe (différents arcs). La relation entre une paire d'articles est notamment calculée sans considérer le contexte de citation, pour éviter de dépendre de données « plein texte » souvent indisponibles ;
- le graphe initial est transformé en graphe non-orienté pour déterminer les chemins entre la conférence et un expert candidat. Cela évite de perdre des relations de co-citation et de couplage bibliographique, comme évoqué précédemment.

### 3.1. Modélisation de l'espace de recherche

Les notations utilisées pour la modélisation de l'espace de recherche sont listées dans le Tableau 1.

| Notation                   | Description                                                                               |
|----------------------------|-------------------------------------------------------------------------------------------|
| $Co$                       | Conférence                                                                                |
| $d$                        | Document correspondant à un article scientifique                                          |
| $d_{citant}$               | Article $d_{citant}$ qui cite un autre article                                            |
| $d_{cité}$                 | Article $d_{cité}$ qui est cité par un autre article                                      |
| $c$                        | Chercheur expert-candidat                                                                 |
| $l_d$                      | Force du lien de publication entre la conférence et l'article $d$ qui y est publié        |
| $v_{d_{citant}, d_{cité}}$ | Force du lien de citation de $d_{citant}$ vers $d_{cité}$                                 |
| $w_{d,c}$                  | Force du lien d'auteur entre un expert-candidat $c$ et son article $d$                    |
| $u_c$                      | Force du lien entre la conférence et l'expert $c$ qui a été membre du comité de programme |
| $M_a$                      | Ensemble des membres du comité de programme de l'édition de l'année $a$                   |
| $Q_d$                      | Ensemble des articles qui citent l'article $d$                                            |
| $R_d$                      | Ensemble des auteurs de l'article $d$                                                     |
| $t_d$                      | Année de publication de l'article $d$                                                     |
| $t_{p_c}$                  | Année de participation de l'expert $c$ au comité de programme $p$                         |
| $t_x$                      | Année de nouvelle édition de conférence pour laquelle suggérer des membres de comité      |

**Tableau 1.** Notations utilisées pour la modélisation de l'espace de recherche.

Nous construisons un graphe orienté avec :

– trois types de **nœuds** :

- 1) la conférence  $Co$  donnée pour laquelle on renouvelle le comité de programme (par exemple, SIGIR). Chaque graphe possède un nœud unique de ce type ;
- 2) les articles publiés à la conférence  $Co$ , ainsi que tous les articles, extérieurs à la conférence, qu'ils citent et qui les citent ;
- 3) les auteurs de ces articles et les membres de comités de programme passés.

– et quatre types de **liens** entre les nœuds suivants :

- 1) d'un article vers un autre pour modéliser une citation ;
- 2) d'un article publié dans une édition de la conférence  $Co$  donnée ;
- 3) d'un auteur vers l'article qu'il co-signe ;
- 4) d'un membre de comité de programme vers la conférence  $Co$  donnée.

Pour deux documents  $d_{citant}$  et  $d_{cité}$ , l'auteur de  $d_{citant}$  connaît le contenu scientifique de  $d_{cité}$  puisqu'il y fait référence dans son article. Il indique ainsi une proximité conceptuelle entre les deux articles. Certaines approches, comme (Liang *et al.*, 2011), parcourent le graphe orienté, d'un document vers d'autres. Dans notre approche, se restreindre au graphe *orienté* ne permet pas d'exploiter toute la richesse des relations de proximité conceptuelle entre l'ensemble des éléments portés par le graphe.

En revanche, notre proposition considère les relations conférence-article, conférence-expert et article-auteur comme non-orientées pour déterminer la proximité

entre la conférence et les nœuds experts-candidats. Un exemple de graphe est présenté dans la Figure 3.

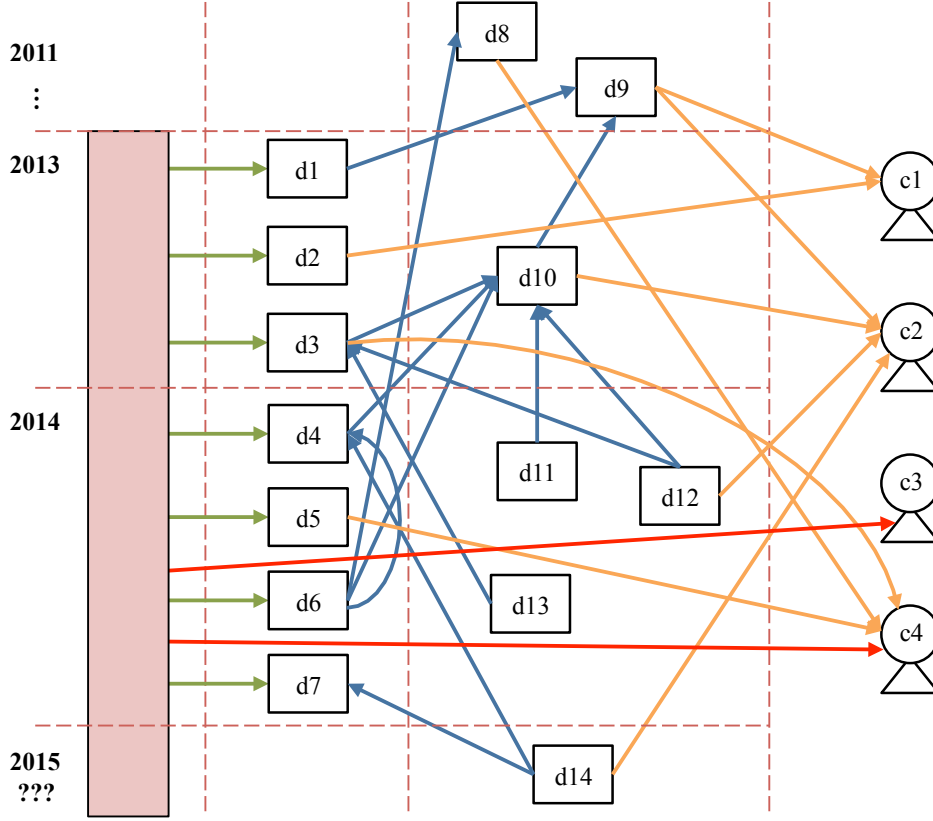


**Figure 3.** Exemple de graphe modélisant l'espace de recherche.

### 3.2. Estimation de la force des liens dans le graphe

Dans cette section, nous proposons une méthode pour mesurer la dépendance entre les objets modélisés qui correspondent aux quatre types de lien ci-dessus.

#### 3.2.1. Estimation de la force du lien de citation entre articles

La force du lien de citation est estimée par deux facteurs :

1) l'écart de temps entre la publication de l'article  $d_{cité}$  et celle de l'article  $d_{citant}$ . Une citation vers un article publié il y a longtemps suggère que cet article est encore d'actualité. Ainsi, la force de la citation croît en fonction de l'écart entre les années de publication ;

2) le nombre d'articles qui citent  $d_{cité}$ . Il s'agit de valoriser les articles citant des articles eux-mêmes très cités, qui témoignent d'un fort impact auprès de la communauté scientifique.

La force du lien de citation quand  $d_{citant}$  cite  $d_{cité}$  est donc définie par

$$v_{d_{citant}, d_{cité}} = \frac{|Q_{d_{cité}}|}{Q_{max}} \cdot e^{\left(\frac{t_{d_{citant}} - t_{d_{cité}}}{\Delta t_{max}}\right)} \quad [1]$$

où  $Q_{max}$  est le nombre maximum d'articles qui citent un article et  $\Delta t_{max}$  est le plus long écart de temps, en valeur absolue, entre deux articles liés par une relation de citation. L'utilisation de l'exponentielle pour le facteur relatif à l'écart de temps permet d'accentuer les écarts entre les valeurs.

### 3.2.2. Estimation de la force du lien entre une conférence et un article publié

La relation entre la conférence et ses articles atteste de leur similarité thématique. Cette relation est a priori identique entre la conférence et tous les articles qui y sont publiés. Cependant, les articles les plus cités contribuent davantage à la notoriété de la conférence. De plus, les articles récemment publiés sont porteurs, a priori, des thématiques actuelles. Cette relation dépend ainsi de deux facteurs :

1) du nombre d'articles citant l'article  $d$ . Parmi les articles qui sont publiés par la conférence, ceux qui sont les plus cités constituent un point d'entrée plus important vers la conférence. Ils contribuent à la visibilité et à l'intérêt pour la conférence ;

2) de l'écart de temps entre la publication de l'article  $d$  et l'édition de la conférence  $t_x$  pour laquelle on souhaite renouveler le comité de programme. Les articles les plus récents traitent des thèmes les plus actuels, et par conséquent plus importants dans une optique de renouvellement de comité de programme.

Ainsi, la force du lien entre une conférence et un article publié lors de l'édition de l'année  $t_d$  est estimée par

$$l_d = \frac{|Q_d|}{Q_{max}} \cdot e^{\left(\frac{t_d - t_x}{\Delta t_{max}}\right)} \quad [2]$$

où  $\Delta t_{max}$  est l'écart de temps maximum, en valeur absolue, entre l'article le plus ancien et l'édition de conférence considérée.

### 3.2.3. Estimation de la force du lien entre article et auteur

Étant donné qu'il est difficile de quantifier la participation relative de chaque co-signataire à un article à la lumière des seules méta-données d'une notice bibliographique, nous considérons un taux participation identique pour chaque co-signataire. La force de la relation entre un article  $d$  et un auteur  $c$  est donc définie par

$$w_{d,c} = \frac{1}{|R_d|}. \quad [3]$$

### 3.2.4. Estimation de la force du lien entre conférence et membre de comité

Une participation au comité de programme d'une conférence indique que l'expertise d'un individu est reconnue sur les thématiques de la conférence. La dépendance

est définie en fonction de l'écart de temps entre l'année de participation de l'expert au comité de programme  $t_{p_c}$  de cette conférence et l'édition de la conférence  $t_x$  pour laquelle on souhaite renouveler le comité de programme. Un expert ayant participé au comité de programme d'une édition récente est considéré plus pertinent qu'un expert ayant participé à un comité plus ancien.

Étant donné un expert  $c$  ayant une participation au comité de programme  $p$ , la force de la relation entre  $c$  et la conférence est définie par la formule

$$u_c = e^{\left(\frac{t_{p_c} - t_x}{\Delta t_{max}}\right)} \quad [4]$$

où  $\Delta t_{max}$  est l'écart de temps maximum, en valeur absolue, entre l'édition de conférence considérée et le plus ancien comité de programme. L'utilisation de l'exponentielle permet d'accentuer les écarts de temps.

### 3.3. Émettre une suggestion d'experts pour un comité de programme

Pour émettre une suggestion, la proximité entre la conférence donnée et l'expert-candidat dépend de deux facteurs. Premièrement, elle dépend du nombre de chemins reliant le nœud de la conférence et le nœud de l'expert-candidat. La proximité augmente avec ce nombre. Deuxièmement, elle dépend de la force de chaque chemin reliant le nœud de la conférence et le nœud de l'expert-candidat. La force d'un chemin est déterminée par la somme des forces (normalisées) des liens qui constituent le chemin. Plus il y aura de chemins avec des forces élevées, plus la proximité entre la conférence et l'expert-candidat sera élevée.

Pour déterminer les chemins entre le nœud de la conférence et le nœud d'un expert-candidat, le graphe est considéré comme non-orienté. Trois types de chemin sont considérés :

– le premier type appelé *auteur externe* (AE) considère l'expert-candidat  $c$  en tant qu'auteur d'un article  $d'$  qui est cité ou qui cite l'article  $d$  de cette conférence. Dans ce cas, le  $i^e$  chemin possède trois segments dont les forces sont calculées par les fonctions  $l_d$ ,  $v_{d_{citant}, d_{cité}}$  et  $w_{d,c}$  selon

$$P_c^{(i)}(AE) = \begin{cases} l_d^{(i)} + v_{d,d'}^{(i)} + w_{d',c}^{(i)} & \text{si l'article } d \text{ publié dans la conférence} \\ & \text{cite l'article } d', \\ l_d^{(i)} + v_{d',d}^{(i)} + w_{d',c}^{(i)} & \text{si l'article } d \text{ publié dans la conférence} \\ & \text{est cité par l'article } d'. \end{cases} \quad [5]$$

– le deuxième type appelé *auteur interne* (AI) considère l'expert-candidat  $c$  en tant qu'auteur de l'article  $d$  publié par la conférence donnée. Dans ce cas, le  $i^e$  chemin possède deux segments dont les forces sont calculées par les fonctions  $l_d$  et  $w_{d,c}$  selon

$$P_c^{(i)}(AI) = l_d^{(i)} + w_{d,c}^{(i)}. \quad [6]$$

– le troisième type appelé *comité de programme* (CP) considère l'expert-candidat  $c$  en tant que membre d'un précédent comité de programme. Ici, le chemin possède un seul segment dont la force est calculée par la fonction  $u_c$  :

$$P_c^{(i)}(CP) = u_c^{(i)} \quad [7]$$

Pour chaque type de chemin  $X \in \{AE, AI, CP\}$  est ensuite calculée la somme des forces de tous les chemins menant du nœud de la conférence à celui de l'expert-candidat selon

$$P_c^{(*)}(X) = \sum_i P_c^{(i)}(X). \quad [8]$$

Enfin, la proximité entre la conférence et chaque expert-candidat  $c$  est déterminée par la somme de tous les chemins entre les deux nœuds :

$$Prox_c = \frac{a \cdot P_c^{(*)}(AE)}{\max_{c' \in R_{\bullet} \cup M_{\bullet}} (P_{c'}^{(*)}(AE))} + \frac{b \cdot P_c^{(*)}(AI)}{\max_{c' \in R_{\bullet} \cup M_{\bullet}} (P_{c'}^{(*)}(AI))} + \frac{c \cdot P_c^{(*)}(CP)}{\max_{c' \in R_{\bullet} \cup M_{\bullet}} (P_{c'}^{(*)}(CP))} \quad [9]$$

où  $a$ ,  $b$  et  $c$  sont les paramètres pour ajuster l'importance de chaque facteur motivant l'expertise et  $R_{\bullet} \cup M_{\bullet}$  désigne l'ensemble des auteurs et membres de comités de programme recensés.

## 4. Expérimentations

### 4.1. Collection de données

Pour valider notre proposition, nous l'avons expérimentée sur les données des différentes éditions de la conférence SIGIR (*Special Interest Group on Information Retrieval*). L'intérêt de cette conférence est qu'elle fait partie des principales conférences du domaine de la recherche d'information, qu'elle existe depuis 1971 et possède donc de nombreuses éditions et de nombreux comités de programmes. De plus, elle totalise un grand nombre de publications, de chercheurs qui ont participé aux comités de programme ou comme auteurs. Un autre atout est que la plupart des données nécessaires à nos expérimentations sont disponibles dans les bases bibliographiques ACM et DBLP (depuis l'édition de 1978) permettant d'avoir ainsi des données homogènes.

Un premier travail a consisté en la collecte de l'ensemble des articles publiés dans les éditions de la conférence : 3 761 articles des 38 éditions de la conférence SIGIR depuis 1978. Ces articles forment un premier ensemble noté « B1 ». Un second travail de collecte a porté sur les articles cités par les articles de la conférence SIGIR (ensemble noté « B2 »). Nous avons également collecté les articles qui citent les articles de la conférence SIGIR (ensemble « B3 »). Enfin, les participations aux différents comités de programme de la conférence SIGIR ont également été collectées manuellement depuis l'édition 2005, en consultant les actes du congrès.

## 4.2. Résultats expérimentaux

Les expérimentations ont consisté à produire des suggestions pour les CP des cinq éditions de 2011 à 2015. Pour chaque édition, les articles des ensembles B2 et B3 ont été restreints aux éditions précédentes de SIGIR de un à dix ans auparavant. Par exemple, les données de 2001 à 2010 ont été utilisées pour les suggestions du comité de 2011. Nous avons testé quatre configurations de paramètres pour  $a$ ,  $b$  et  $c$  indiquées dans les tableaux de résultats présentés.

Les suggestions de membres de CP sont évaluées par rapport à une vérité terrain exploitant l'historique de SIGIR. Ainsi, pour une année donnée, les suggestions sont comparées par intersection avec le CP de cette année là (Tableau 2), puis avec l'ensemble des membres des CP antérieurs (Tableau 3).

Le Tableau 2 présente les résultats (cardinal de l'intersection et taux de recouvrement du CP par les suggestions) pour une suggestion du même nombre (respectivement de 50% en plus et de 100 % en plus) d'experts que le comité correspondant.

| Année<br>(a)                          | Nb de<br>membres<br>CP ( $ M_a $ ) | Nb d'experts = $ M_a $     |                      | Nb d'experts = 150 % $ M_a $ |                      | Nb d'experts = 200 % $ M_a $ |                      |
|---------------------------------------|------------------------------------|----------------------------|----------------------|------------------------------|----------------------|------------------------------|----------------------|
|                                       |                                    | Nb d'experts<br>dans $M_a$ | Taux<br>recouvrement | Nb d'experts<br>dans $M_a$   | Taux<br>recouvrement | Nb d'experts<br>dans $M_a$   | Taux<br>recouvrement |
| Configuration 1 : a=1 ; b=1 ; c=1     |                                    |                            |                      |                              |                      |                              |                      |
| 2011                                  | 423                                | 224                        | 52,96 %              | 249                          | 58,87 %              | 299                          | 70,69 %              |
| 2012                                  | 478                                | 215                        | 44,98 %              | 251                          | 52,51 %              | 312                          | 65,27 %              |
| 2013                                  | 403                                | 175                        | 43,42 %              | 216                          | 53,60 %              | 229                          | 56,82 %              |
| 2014                                  | 421                                | 178                        | 42,28 %              | 235                          | 55,82 %              | 251                          | 59,62 %              |
| 2015                                  | 440                                | 213                        | 48,41 %              | 311                          | 70,68 %              | 440                          | 100,0 %              |
| Configuration 2 : a=1 ; b=0,5 ; c=0,5 |                                    |                            |                      |                              |                      |                              |                      |
| 2011                                  | 423                                | 182                        | 43,03 %              | 252                          | 59,57 %              | 268                          | 63,36 %              |
| 2012                                  | 478                                | 186                        | 38,91 %              | 251                          | 52,51 %              | 284                          | 59,41 %              |
| 2013                                  | 403                                | 153                        | 37,97 %              | 202                          | 50,12 %              | 232                          | 57,57 %              |
| 2014                                  | 421                                | 184                        | 43,71 %              | 225                          | 53,44 %              | 266                          | 63,18 %              |
| 2015                                  | 440                                | 201                        | 45,68 %              | 252                          | 57,27 %              | 376                          | 85,45 %              |
| Configuration 3 : a=0,5 ; b=1 ; c=0,5 |                                    |                            |                      |                              |                      |                              |                      |
| 2011                                  | 423                                | 213                        | 50,35 %              | 249                          | 58,87 %              | 273                          | 64,54 %              |
| 2012                                  | 478                                | 218                        | 45,61 %              | 250                          | 52,30 %              | 278                          | 58,16 %              |
| 2013                                  | 403                                | 152                        | 37,72 %              | 209                          | 51,86 %              | 225                          | 55,83 %              |
| 2014                                  | 421                                | 184                        | 43,71 %              | 235                          | 55,82 %              | 258                          | 61,28 %              |
| 2015                                  | 440                                | 201                        | 45,68 %              | 239                          | 54,32 %              | 365                          | 82,95 %              |
| Configuration 4 : a=0,5 ; b=0,5 ; c=1 |                                    |                            |                      |                              |                      |                              |                      |
| 2011                                  | 423                                | 229                        | 54,14 %              | 253                          | 59,81 %              | 384                          | 90,78 %              |
| 2012                                  | 478                                | 248                        | 51,88 %              | 314                          | 65,69 %              | 354                          | 74,06 %              |
| 2013                                  | 403                                | 197                        | 48,88 %              | 206                          | 51,12 %              | 262                          | 65,01 %              |
| 2014                                  | 421                                | 206                        | 48,93 %              | 224                          | 53,21 %              | 290                          | 68,88 %              |
| 2015                                  | 440                                | 326                        | 74,09 %              | 376                          | 85,45 %              | 433                          | 98,41 %              |

**Tableau 2.** Comparaison des experts suggérés par notre système avec le CP original de l'année correspondante. Les résultats présentés correspondent à un nombre d'experts suggérés égal à  $|M_a|$ , puis 150 % de  $|M_a|$  et enfin 200 % de  $|M_a|$ .

Pour une taille de comité stable d'une année à l'autre, notre méthode arrive à suggérer approximativement la moitié de membres (effectifs) de CP. L'autre moitié comprend des experts suggérés via l'analyse des publications, des citations et des participations aux CP précédents. Ces personnes sont des candidats potentiels à considérer

## Suggestion d'experts pour renouveler le comité de programme d'une conférence

pour renouveler le CP. Pour évaluer l'implication passée dans des CP de SIGIR de ces experts, le Tableau 3 montre la proportion des experts répondant à ce critère.

| Année<br>(a)                          | Nb de<br>membres<br>CP ( $ M_a $ ) | Nb d'experts = $ M_a $                      |                      | Nb d'experts = 150% $ M_a $                 |                      | Nb d'experts = 200% $ M_a $                 |                      |
|---------------------------------------|------------------------------------|---------------------------------------------|----------------------|---------------------------------------------|----------------------|---------------------------------------------|----------------------|
|                                       |                                    | Nb d'experts dans<br>$M_{a'}$ avec $a' < a$ | Taux<br>recouvrement | Nb d'experts dans<br>$M_{a'}$ avec $a' < a$ | Taux<br>recouvrement | Nb d'experts dans<br>$M_{a'}$ avec $a' < a$ | Taux<br>recouvrement |
| Configuration 1 : a=1 ; b=1 ; c=1     |                                    |                                             |                      |                                             |                      |                                             |                      |
| 2011                                  | 423                                | 408                                         | 96,45 %              | 612                                         | 96,53 %              | 773                                         | 91,37 %              |
| 2012                                  | 478                                | 456                                         | 95,40 %              | 682                                         | 95,12 %              | 869                                         | 90,90 %              |
| 2013                                  | 403                                | 385                                         | 95,53 %              | 581                                         | 96,19 %              | 754                                         | 93,55 %              |
| 2014                                  | 421                                | 404                                         | 95,96 %              | 600                                         | 95,09 %              | 795                                         | 94,42 %              |
| 2015                                  | 440                                | 421                                         | 95,68 %              | 632                                         | 95,76 %              | 845                                         | 96,02 %              |
| Configuration 2 : a=1 ; b=0,5 ; c=0,5 |                                    |                                             |                      |                                             |                      |                                             |                      |
| 2011                                  | 423                                | 375                                         | 88,65 %              | 568                                         | 89,59 %              | 746                                         | 88,18 %              |
| 2012                                  | 478                                | 414                                         | 86,61 %              | 632                                         | 88,15 %              | 819                                         | 85,67 %              |
| 2013                                  | 403                                | 353                                         | 87,59 %              | 512                                         | 84,77 %              | 703                                         | 87,22 %              |
| 2014                                  | 421                                | 366                                         | 86,94 %              | 533                                         | 84,47 %              | 713                                         | 84,68 %              |
| 2015                                  | 440                                | 392                                         | 89,09 %              | 570                                         | 86,36 %              | 790                                         | 89,77 %              |
| Configuration 3 : a=0,5 ; b=1 ; c=0,5 |                                    |                                             |                      |                                             |                      |                                             |                      |
| 2011                                  | 423                                | 389                                         | 91,96 %              | 591                                         | 93,22 %              | 748                                         | 88,42 %              |
| 2012                                  | 478                                | 419                                         | 87,66 %              | 667                                         | 93,03 %              | 822                                         | 85,98 %              |
| 2013                                  | 403                                | 376                                         | 93,30 %              | 548                                         | 90,73 %              | 741                                         | 91,94 %              |
| 2014                                  | 421                                | 385                                         | 91,45 %              | 564                                         | 89,38 %              | 759                                         | 90,14 %              |
| 2015                                  | 440                                | 408                                         | 92,73 %              | 585                                         | 88,64 %              | 805                                         | 91,48 %              |
| Configuration 4 : a=0,5 ; b=0,5 ; c=1 |                                    |                                             |                      |                                             |                      |                                             |                      |
| 2011                                  | 423                                | 422                                         | 99,76 %              | 624                                         | 98,42 %              | 836                                         | 98,82 %              |
| 2012                                  | 478                                | 474                                         | 99,16 %              | 698                                         | 97,35 %              | 937                                         | 98,01 %              |
| 2013                                  | 403                                | 401                                         | 99,50 %              | 597                                         | 98,84 %              | 780                                         | 96,77 %              |
| 2014                                  | 421                                | 418                                         | 99,29 %              | 624                                         | 98,89 %              | 815                                         | 96,79 %              |
| 2015                                  | 440                                | 434                                         | 98,64 %              | 654                                         | 99,09 %              | 872                                         | 99,09 %              |

**Tableau 3.** Comparaison des experts suggérés par notre système avec les CP originaux des années précédentes. Les résultats présentés correspondent à un nombre d'experts suggérés égal à  $|M_a|$ , puis 150 % de  $|M_a|$  et enfin 200 % de  $|M_a|$ .

Les résultats montrent l'effet des configurations sur l'ampleur du renouvellement suggéré pour les CP par des experts n'ayant jamais siégé. Par exemple, la configuration 4 favorise l'endogamie en promouvant les membres de CP précédents qui représentent quasiment 100 % des suggestions. Elle contraste avec la configuration 2 qui favorise l'exogamie en suggérant environ 15 % de scientifiques n'ayant pas siégé en CP quoique publiant dans le même espace intellectuel que la conférence ciblée. Ces deux exemples soulignent le potentiel d'adaptation des suggestions aux besoins exprimés par les présidents de CP souhaitant tantôt entretenir tantôt renouveler les pairs en charge de l'évaluation.

## 5. Conclusion et perspectives

Pour suggérer des experts afin de renouveler le CP d'une conférence, nous avons défini un modèle intégrant différents objets et relations. Ceux-ci constituent un graphe portant les preuves d'expertises observées dans diverses sources liées aux conférences, articles et participants à des CP.

Les suggestions produites par notre modèle ont été confrontées aux CP sous-tendant plusieurs éditions de SIGIR. Un point délicat de la validation de notre approche concerne la vérité terrain employée. Faute de pouvoir questionner les prési-



dents de CP à propos des listes que nous suggérons, nous avons fait l'hypothèse que la suggestion d'un membre de CP « effectif » (potentiellement d'un CP dans le passé) était pertinente. C'est cependant une approche biaisée, en défaveur des scientifiques proches de la communauté ciblée quoique n'ayant jamais siégé au CP.

Nous cherchons à relever un défi : suggérer des scientifiques experts pour une conférence sans exploiter le contenu textuel (au sens RI) des articles publiés. Cette contrainte vient notamment des barrières posées par la plupart des éditeurs bloquant l'accès au plein texte des publications. Par conséquent, il nous semble pertinent d'exploiter les relations de citation et de co-signature traduisant une proximité conceptuelle entre scientifiques et la communauté formée par une conférence. Afin de promouvoir certains critères dans les suggestions (tels que le genre, l'âge et la localisation géographique), nous envisageons d'intégrer ces caractéristiques des scientifiques dans de futurs travaux.

## 6. Bibliographie

- Afzal M., Maurer H., « Expertise Recommender System for Scientific Community », *Journal of Universal Computer Science*, vol. 17, p. 1529-1549, 2011.
- Balog K., Fang Y., de Rijke M., Serdyukov P., Si L., « Expertise Retrieval », *Foundations and Trends in Information Retrieval*, vol. 6, p. 127-256, 2012.
- Bobadilla J., Ortega F., Hernando A., Gutiérrez A., « Recommender systems survey », *Knowledge-Based Systems*, vol. 46, p. 109 - 132, 2013.
- Cabanac G., « Accuracy of inter-researcher similarity measures based on topical and social clues », *Scientometrics*, vol. 87, n° 3, p. 597-620, 2011.
- Cronin B., *The Citation Process : The Role and Significance of Citations in Scientific Communication*, Taylor Graham, London, 1984.
- De Bellis N., *Bibliometrics and Citation Analysis*, Scarecrow Press, Lanham, 2009.
- Ekstrand M., Kannan P., Stemper J. A., Butler J. T., Konstan J. A., Riedl J., « Automatically building research reading lists », *Conference on Recommender Systems (RecSys 2010)*, p. 159-166, 2010.
- Fang H., Zhai C., « Probabilistic Models for Expert Finding », *29th European conference on IR research (ECIR'07)*, p. 418-430, 2007.
- Garfield E., « Citation Indexes for Science : A New Dimension in Documentation through Association of Ideas », *Science*, vol. 122, p. 108-111, 1955.
- Goldberg D., Nichols D., Oki B. M., Terry D., « Using collaborative filtering to weave an information tapestry », *Communications of the ACM – Special issue on information filtering*, vol. 35, n° 12, p. 61-70, 1992.
- Huang Z., Chung W., Ong T., Chen H., « A graph-based recommender system for digital library », *ACM/IEEE Joint Conference on Digital Libraries (JCDL 2002)*, p. 65-73, 2002.
- Huang Z., Qiu Y., « A multiple-perspective approach to constructing and aggregating Citation Semantic Link Network », *Future Generation Computer Systems*, vol. 26, n° 3, p. 400-407, 2010.

- Katz L., « A new status index derived from sociometric analysis », *Psychometrika*, vol. 18, n° 1, p. 39-43, 1953.
- Kessler M., « Bibliographic coupling between scientific papers », *American Documentation*, vol. 14, p. 10-25, 1963.
- Lao N., Cohen W., « Relational retrieval using a combination of path-constrained random walks », *Machine Learning*, vol. 81, p. 53-67, 2010.
- Liang Y., Li Q., Qian T., « Finding Relevant Papers Based on Citation Relations », *Web-Age Information Management*, vol. 6897, p. 403-414, 2011.
- Liben-Nowell D., Kleinberg J., « The link-prediction problem for social networks », *Twelfth international conference on Information and knowledge management (CIKM'03)*, p. 556-559, 2003.
- Lin J., Wilbur W. J., « PubMed related articles : a probabilistic topic-based model for content similarity », *BMC Bioinformatics*, 2007.
- Liu P., Dew P., « Using Semantic Web Technologies to Improve Expertise Matching within Academia », *International conference I-KNOW'04*, p. 370-378, 2004.
- Macdonald C., Ounis I., « Voting techniques for expert search », *Knowledge and Information Systems*, vol. 16, n° 3, p. 259-280, 2008.
- McDonald D. W., Ackerman M. S., « Expertise Recommender : A Flexible Recommendation System and Architecture », *ACM conference on Computer supported cooperative work (CSCW'00)*, p. 231-240, 2000.
- Nanba H., Manabu M., « Towards multi-paper summarization reference information », *16th international joint conference on Artificial intelligence (IJCAI'99)*, p. 926-931, 1999.
- Rodriguez M., Bollen J., « An Algorithm to Determine Peer-Reviewers », *17th ACM conference on Information and knowledge management (CIKM'08)*, p. 319-328, 2008.
- Serdyukov P., Hiemstra D., « Modeling Documents As Mixtures of Persons for Expert Finding », *Proceedings of the IR Research, 30th European Conference on Advances in Information Retrieval, ECIR'08*, Springer-Verlag, Berlin, Heidelberg, p. 309-320, 2008.
- Small H., « Co-citation in the scientific literature : A new measure of the relationship between two documents », *Journal of the American Society for Information Science*, vol. 24, p. 265-269, 1973.
- Strohman T., Croft W., Jensen D., « Recommending Citations for Academic Papers », *30th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '07)*, p. 705-706, 2007.
- Tang J., Zhang J., Yao L., Li J., Zhang L., Su Z., « ArnetMiner : extraction and mining of academic social networks », *14th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '08)*, p. 990-998, 2008.
- Tang J., Zhang J., Yu J. X., Yang Z., Cai K., Ma R., Zhang L., Su Z., « Topic Distributions over Links on Web », *Ninth IEEE International Conference on Data Mining (ICDM'09)*, p. 1010-1015, 2009.



CORIA – CIFED 2016

Humanités numériques  
CORIA – CIFED

Présidente : *Brigitte Grau*

Suggestion d'experts pour renouveler le comité de programme d'une  
conférence

Papier absent

Nouveau modèle pour la datation automatique de photographies à  
partir de caractéristiques visuelles (CIFED)

Papier absent

*Paul Martin, Antoine Doucet, Frederic Jurie*

# Papier absent

Nouveau modèle pour la datation automatique de photographies à  
partir de caractéristiques visuelles (CIFED)

Papier absent



# Papier absent



---

# Cascade de CRFs et SVM pour la détection de références bibliographiques diffuses dans les articles scientifiques

Anaïs Ollagnier<sup>\*,\*\*</sup> — Sébastien Fournier<sup>\*</sup> — Patrice Bellot<sup>\*,\*\*</sup>

<sup>\*</sup> Aix-Marseille Université, CNRS, LSIS UMR 7296, 13397 Marseille, France

<sup>\*\*</sup> Aix-Marseille Université, CNRS, CLEO OpenEdition UMS 3287, 13451 Marseille, France

*anaïs.ollagnier, sebastien.fournier, patrice.bellot@lsis.org*

---

**RÉSUMÉ.** Dans le contexte d'une bibliothèque d'articles scientifiques, les références bibliographiques sont une source majeure de liens. Parmi elles, certaines sont explicites comme les références que nous pouvons retrouver à la fin des articles ou des livres. Tandis que d'autres sont dispersées selon un degré de diffusion plus ou moins fort dans le corps du texte. Nous proposons de nous focaliser sur la détection de ce type de références que nous nommons références bibliographiques diffuses présentes dans des articles scientifiques. Nous prenons exemple du domaine des Sciences Humaines et Sociales qui est particulièrement riche de telles citations. Afin de pallier les difficultés inhérentes à la détection de ce type de références, nous introduisons une méthodologie qui consiste, d'une part, à identifier les paragraphes qui contiennent des références via un processus de classification supervisée et, d'autre part, dans l'application de CCRFs afin de détecter plus précisément les zones bibliographiques et d'annoter leurs contenus.

**ABSTRACT.** In the context of a library of scientific articles, bibliographic references are a major source of links. Among them, some are explicit references as we can find at the end of articles or books. While others are scattered according to an implicit degree more or less strong in the text. We propose to focus on the detection of this type of references that we call diffuse in scientific articles from the field of Humanities and Social Sciences. To overcome the difficulties inherent in the detection of such references, we present a method which comprises, firstly, to identify paragraphs that contain references via a classification process and, secondly, in applying CCRFs to more accurately detect the bibliographic entries and annotate their contents.

**MOTS-CLÉS :** CCRFs<sub>1</sub>, Système de recommandation<sub>2</sub>, Détection de référence bibliographique<sub>3</sub>.

**KEYWORDS :** CCRFs<sub>1</sub>, Recommender system<sub>2</sub>, Bibliographical reference detection<sub>3</sub>.

---

## 1. Introduction

Par définition, une référence bibliographique regroupe l'ensemble des éléments de données nécessaires pour identifier un document ou une partie de document de tout type, sur tout support (livre, article, site web, etc.). L'utilisation des références bibliographiques est un élément crucial dans la création et la diffusion d'informations (Cronin, 1984). Le travail présenté dans cet article s'intègre dans le cadre de la réalisation d'un système de recommandation de lectures axé sur l'exploitation de liens entre les contenus. L'exploitation de liens entre les contenus est un élément crucial des approches de recommandations. Nous nous intéressons particulièrement à un aspect inhérent à ce type de système : la caractérisation du contenu du document. Dans les systèmes de recommandation de lectures, l'utilisation du contenu au sein des systèmes de recommandation fonctionnent par la caractérisation du contenu de l'information à filtrer. Les représentations des documents et des profils dans ce type de filtrage exploitent seulement les informations qui peuvent être dérivées de leur thème respectif. Autrement dit, la sélection de documents est fondée sur une comparaison des sujets abordés dans des documents relatifs à des thèmes d'intérêt pour l'utilisateur. Or, de nombreuses informations, telles que les références bibliographiques et les entités nommées dispersées au sein du document peuvent s'avérer utiles lors de la recommandation car elles peuvent servir de liaison entre des informations et des documents directement liés au contenu ciblé. C'est dans cet intérêt que le travail présenté dans cet article se focalise sur la détection des références bibliographiques et plus particulièrement des références bibliographiques diffuses.

Pour rappel, l'extraction d'informations bibliographiques consiste à identifier automatiquement chaque mot d'une référence bibliographique comme l'un des champs bibliographiques prédéfinis, tels que, l'auteur, le titre, la date, etc. L'extraction d'informations bibliographiques est souvent restreinte aux zones les plus formelles, à savoir, les références que nous pouvons trouver généralement à la fin des articles ou des livres. Les résultats dans ce domaine montrent par ailleurs des performances très satisfaisantes (Kim *et al.*, 2011 ; Anzaroot et McCallum, 2013). Cependant, les références bibliographiques peuvent également se trouver à différents niveaux du document tels que les notes de bas de pages ou dans le corps du texte. Ce type de références que nous nommons références diffuses possèdent de nombreuses particularités comme une structuration et une répartition de ses éléments dans le texte très variées. Ces deux aspects sont influencés par le degré d'implicite de la référence. En effet, les références bibliographiques diffuses peuvent être plus ou moins explicites, c'est à dire, que nous pouvons les trouver disséminées dans le texte ou sous la forme de citations qui vont être normées par des conventions selon le type de publication. Ces différents aspects rendent difficiles leur identification par des méthodes classiques d'analyse de références. Le degré d'implicite de la référence peut la rendre difficilement identifiable via de simples caractéristiques lexicales ou morphologiques souvent exploités dans l'annotation automatique de références bibliographiques.

Notre contribution s'intègre autour de la réalisation d'une méthodologie dédiée à la détection de ces références bibliographiques diffuses. Cette méthodologie consiste, d'une part, à identifier les paragraphes qui contiennent des références via un processus

de classification supervisée et, d'autre part, dans l'application de CCRFs afin de détecter plus précisément les zones bibliographiques et d'annoter leurs contenus. L'originalité de notre contribution est d'exploiter les références diffuses en vue d'établir des liens entre les documents dans le cadre de la réalisation d'un système de recommandation de lectures. De plus, dans notre travail, nous traitons avec des données de Sciences Humaines et Sociales (SHS) qui est un défi, car en plus de traiter un domaine multidisciplinaire très diversifié la présence de nombreuses variations structurelles des références bibliographiques rend l'homogénéisation des traitements difficiles.

Le document est structuré de la manière suivante. La section 2 présente un état de l'art des méthodes traditionnelles de détection et d'annotation de références bibliographique ainsi que des outils déjà existant. La section 3 se focalise sur la présentation des données que nous allons utiliser afin de valider notre approche. La section 4 présente notre approche et la section 5 expose les expérimentations et les résultats. Enfin, la section 6 conclut.

## 2. Travaux connexes

Depuis les années 90, l'analyse automatique de références suscite beaucoup d'attention pas seulement pour leur utilisation dans les processus d'indexation mais aussi pour améliorer les performances lors de la récupération et l'extraction d'information. CiteSeer<sup>1</sup> (Giles *et al.*, 1998) qui est un moteur de recherche publique ainsi qu'une bibliothèque numérique pour les articles scientifiques appartenant au domaine informatique, est l'un des premiers systèmes d'indexation automatique de citations. Depuis, plusieurs outils permettent l'analyse du réseau de citation de la littérature scientifique avec une vocation disciplinaire ou pluridisciplinaire (Bellis, 2009). Au niveau disciplinaire, des bases de données fournissent aux spécialistes du domaine via des systèmes autonomes d'analyse de citations, notamment ceux fournis par le *Chemical Abstract Service*<sup>2</sup> (CAS) pour la chimie et d'autres sciences connexes ; le *SAO/NASA Astrophysics Data System*<sup>3</sup> (ADS) pour l'astronomie ; *IEEE Xplore*<sup>4</sup> pour le domaine informatique. Au niveau multidisciplinaire, nous pouvons citer *Google Scholar*<sup>5</sup> et *Scopus*<sup>6</sup>.

Le problème de l'analyse des références bibliographiques est considéré comme un problème de labellisation de séquence dans lequel la référence est considérée comme une chaîne de caractère. Beaucoup de travaux se sont penchés sur l'annotation de séquences selon des classes prédéfinies, nous pouvons classer ces travaux en trois grandes approches : expressions régulières basées sur des heuristiques, algorithmes d'apprentissage et des systèmes à base de connaissances. La première approche fon-

---

1. <http://citeseerx.ist.psu.edu/>

2. <https://www.cas.org/>

3. <http://ads.harvard.edu/>

4. <http://ieeexplore.ieee.org/Xplore/home.jsp>

5. <https://scholar.google.fr/>

6. <http://www.scopus.com/>

dée sur des expressions régulières basées sur des heuristiques, comme le définissent (Huang *et al.*, 2004), consiste à extraire des séquences redondantes via l'utilisation d'algorithmes heuristiques. Ces séquences permettent ensuite d'établir des modèles pour l'analyse des séquences. L'utilisation de ces approches basées sur des heuristiques sont un bon compromis entre la qualité des solutions trouvées et la rapidité de la mise en oeuvre du procédé, toutefois, ils ne garantissent pas nécessairement des solutions optimales selon la complexité de la tâche à réaliser (Deconinck, 2010). La seconde approche fondée sur des algorithmes d'apprentissage emploie des algorithmes qui opèrent en construisant un modèle basé sur un corpus donné en entrée. Le modèle construit procède ensuite à des prédictions ou des décisions afin d'analyser les séquences. Bien que ces approches ont une bonne capacité d'adaptation et de bons résultats (Anzaroot et McCallum, 2013), elles ont des limitations et, en particulier, le corpus d'apprentissage est souvent hautement dépendant des données d'application. La troisième approche fondée sur des systèmes à base de connaissances utilise, dans la plus part des cas, les connaissances de domaine pour dériver une ontologie qui décrit les données d'intérêt. Cette connaissance va alors inclure des informations comme les relations, les caractéristiques lexicales et des mots clés contextuels (Rezaei et Muntz, 2013). Par l'analyse de l'ontologie, plusieurs règles et extracteurs peuvent être ainsi générés. Ces règles et extracteurs sont ensuite utilisés pour effectuer l'extraction de l'information. Cependant, il nécessite un expert du domaine pour maintenir la base de connaissances.

En dehors des techniques largement employées pour l'analyse des références, de nombreux outils ont été réalisés. La plupart de ces outils sont basés sur des algorithmes d'apprentissage, en particulier sur les CRFs. ParsCit<sup>7</sup> (Councill *et al.*, 2008) basé sur des CRFs fournit une boîte à outil qui permet l'annotation générale de références dans les zones bibliographiques. Biblio Citation Parser<sup>8</sup> est développé simultanément que ParsCit, et il est également conçu pour les références à la fin des articles. Freecite<sup>9</sup> est également inspiré par ParsCit, il utilise des CRFs avec l'implémentation de bibliothèques CRF++. Grobid<sup>10</sup> (Lopez, 2009) permet l'extraction et l'analyse de références bibliographiques sur la base de CRFs. Un point intéressant est que cet outil peut enrichir l'annotation via l'utilisation de données externes. Bibpro<sup>11</sup> (Chen *et al.*, 2012) capture les propriétés structurelles et transforme des propriétés en un modèle de séquence. Son modèle est basé sur l'alignement de séquence, des algorithmes d'apprentissage et des systèmes à base de connaissances. Bilbo<sup>12</sup> (Kim *et al.*, 2012a) permet l'annotation des références présentes dans les zones bibliographiques ainsi que celles présentes dans les notes de bas de pages. Ses modèles sont basés sur l'utilisation de CRFs mais également de machine à vecteurs de support (*Support Vector Machine* (SVM)) pour la classification des notes de bas de pages. Nous constatons, suite à l'étude de ces dif-

7. <http://aye.comp.nus.edu.sg/parsCit/>

8. <http://paracite.eprints.org/developers/>

9. <http://freecite.library.brown.edu/welcome>

10. <https://github.com/grobid/grobid>

11. <https://github.com/ice91/BibPro>

12. <https://github.com/OpenEdition/bilbo>

férents outils, qu’aucun ne permet l’annotation des références disséminées au sein du corps du texte.

Dans le cadre de notre travail, nous avons choisi d’utiliser une approche basée sur des algorithmes d’apprentissage, et plus particulièrement fondée sur des CRFs, parce qu’ils constituent un très bon compromis entre performances et rapidité de mise en oeuvre. De plus la littérature démontre leur grande adaptabilité et leurs bonnes performances sur l’analyse de séquences (Kim *et al.*, 2012b).

### 3. Données

Dans le cadre de notre travail, nous utilisons les données fournis par le *Centre pour L’édition Électronique Ouverte* (CLEO) extraites du Portail OpenEdition<sup>13</sup>. OpenEdition est une plateforme en ligne destinée à des articles électroniques, des livres, des blogs scientifiques dans le domaine des SHS. Il se compose de quatre sous plateformes, Revues.org, Calenda, Books et Hypotheses. Le domaine des SHS est un domaine multidisciplinaire qui nous permet d’avoir une représentation de la grande majorité des types de formats des références bibliographiques. En effet, les conventions employées dans ce domaine sont très diversifiées par rapport à celles utilisées dans les sciences dites «dures». Bien qu’il existe des conventions plus largement utilisées comme celle de l’Association Américaine de Psychologie (*American Psychological Association*<sup>14</sup>) ou encore de l’Association du Langage Moderne (*Modern Language Association*<sup>15</sup>) de nombreuses organisations établissent leur propre convention afin de répondre à leurs besoins.

Les données que nous utilisons proviennent de l’index SoLr d’OpenEdition au format XML. Dans le cadre d’un programme de R&D du laboratoire OpenEdition trois corpus ont été annotés manuellement suivant le formalisme de la TEI<sup>16</sup>. Le premier corpus est exclusivement destiné à l’apprentissage de modèles pour les références bibliographiques présentes dans les zones bibliographiques. Le second corpus se concentre uniquement sur les notes de bas de pages fortement normées. Et enfin, le troisième corpus est constitué de références bibliographiques diffuses. Tous ces corpus sont établis uniquement sur les articles extraits de la plateforme Revues.org. Concernant la constitution du corpus des références bibliographiques diffuses, les revues sélectionnées ont été choisies aléatoirement dans la base de données d’OpenEdition sans tenir compte de l’appartenance disciplinaire de la revue ou du style bibliographique de la référence.

Dans le cadre de notre travail, nous nous intéressons uniquement au troisième cor-

13. <http://www.openedition.org/>

14. <http://www.apa.org/>

15. <https://www.library.cornell.edu/research/citation/mla>

16. TEI est un consortium qui développe, définit et maintient un langage de balisage pour décrire les caractéristiques structurelles et conceptuelles de textes.

pus orienté sur la détection des références bibliographiques diffuses<sup>17</sup>. Ce corpus se décompose en trois sous catégories, à savoir :

- **Notes de bas de pages particulières** : composée de 43 références bibliographiques, ces références sont des cas particuliers que l'on retrouve dans les notes de bas de pages.

```
<note> Qui a cependant été, ensuite, jugée d'une sévérité excessive envers la bourgeoisie protestante du xviie siècle. Ainsi, <bibl><author><forename>E.</forename> <surname>Labrousse</surname></author> montre qu'à partir du moment où la logique absolutiste était en marche, quelle que soit la stratégie mise en œuvre par les protestants, elle se retournait contre eux (Cf. <title>« Une foi, une loi, un roi » ? La révocation de l'Édit de Nantes</title>, <pubPlace>Paris-Genève</pubPlace>, <publisher>Payot-Labor</publisher> et <publisher>Fides</publisher>, <date>1985</date>)</bibl>.</note>
```

Figure 1 – Exemple de référence bibliographique de la catégorie Notes de bas de pages particulières

- **Références bibliographiques courtes** : composée de 449 références bibliographiques, cette catégorie réfère à des références composées de très peu d'éléments (de nombreuses variations entre auteur/date, titre/auteur, titre/page, etc.)

La pragmatique, qui a marqué le paysage des études de textes au cours des dernières décennies, se fonde sur un retour de l'intentionnalité, et conjointement, sur le rôle du contexte de la communication (<bibl><author><surname>Strawson</surname></author> <date>1970</date></bibl>).

Figure 2 – Exemple de référence bibliographique de la catégorie Références bibliographiques courtes

- **Références bibliographiques entrelacées** : composée de 553 références bibliographiques, cette catégorie se compose de références qui sont « implicites », c'est-à-dire, qu'elles sont exprimées de manière informelle.

Dans <bibl>le <biblScope>premier numéro</biblScope> de l'<title>Almanach des gourmands</title> (<author><forename>Grimod</forename> <surname>de la Reynière</surname></author>, <date>1804</date> cité in <bibl> <author><surname>Duhart</surname></author> <date>2001</date></bibl>)</bibl>, il est ainsi indiqué qu'il est possible de se procurer des pâtés de foie gras auprès des boutiques de comestibles les plus réputées, telles que Corcellet, Rouget et l'Hôtel des Américains (Duhart, 2001).

Figure 3 – Exemple de référence bibliographique de la catégorie Références bibliographiques entrelacées

Suite à l'étude des références présentes au sein des revues constitutives du corpus des références bibliographiques diffuses nous avons identifié différents types de références ce qui explique les proportions différentes de références pour chacune des sous-catégories.

#### 4. Méthodologie

Cette section présente, premièrement, le modèle de classification mis en place afin de détecter les zones contenant potentiellement des références bibliographiques. Et

17. <http://lab.hypotheses.org/>. Écrire à [marin.dacos@openedition.org](mailto:marin.dacos@openedition.org) pour avoir accès à la collection.



secondement, les différents CRFs établis pour la détection des zones bibliographiques et l'annotation de leurs contenus.

#### 4.1. Classification supervisée des paragraphes contenant des références bibliographiques

Au vu de la quantité importante de paragraphes ne comportant pas de références bibliographiques nous décidons d'effectuer un pré-filtrage via l'utilisation d'une classification supervisée. Pour ce faire, nous établissons deux classes : zone bibliographique/zone non bibliographique. La catégorie «Notes de bas de pages particulières» comporte 4,3% de paragraphes avec des zones bibliographiques sur un total de 306 paragraphes. La catégorie «Références bibliographiques courtes» contient 20,8% de paragraphes avec des zones bibliographiques sur un total de 725 paragraphes. Et la catégorie «Références bibliographiques entrelacées» dénombre 23,3% de paragraphes avec des zones bibliographiques sur un total de 1342 paragraphes. Ensuite, nous choisissons d'utiliser une technique de classification utilisant SVM. Pour l'implémentation du SVM, nous utilisons l'outil *SVMLight*<sup>18</sup> développé par (Thorsten, 1999). Concernant les paramétrages effectués, nous établissons une liste de mots les plus caractéristiques de chaque classe que nous utilisons comme attributs. Cette liste est réalisée grâce à l'algorithme *InfoGainAttribute* (IGA). IGA permet d'effectuer un ratio d'informations pour acquérir les informations intrinsèques, il est utilisé pour réduire un biais en faveur des attributs à valeurs multiples. Après plusieurs tests, nous avons choisi d'utiliser une fréquence minimale d'apparition des termes de 1 combinée à une liste pour laquelle nous avons supprimé les mots dont le score « Élimination récursive de caractéristiques »<sup>19</sup> (RFE) est égal à 0. Pour chaque sous-catégorie, nous effectuons 10 validations croisées afin d'évaluer la façon dont les résultats se généralisent à un ensemble de données. Le tableau 1 présente les résultats obtenus suite à la classification pour chacune des sous-catégories.

| Nom corpus                              | accuracy | précision | rappel |
|-----------------------------------------|----------|-----------|--------|
| Notes de bas de pages particulières     | 59,1%    | 66,7%     | 36,4%  |
| Références bibliographiques courtes     | 74,8%    | 71,0%     | 83,1%  |
| Références bibliographiques entrelacées | 79,0%    | 88,6%     | 71,2%  |

Tableau 1 – Résultats pour la classification supervisée

Concernant les performances de classification, pour la catégorie «Notes de bas de pages particulières», nous obtenons une «accuracy» sur le jeu de test de 59,1% ce

18. <http://svmlight.joachims.org/>

19. *Recursive Feature Elimination* : élimination récursive de caractéristiques.

qui correspond à 13 références classées correctement et 9 références classées comme incorrectes. Pour la catégorie «Références bibliographiques courtes», nous obtenons une «accuracy» sur le jeu de test de 74,8% ce qui correspond à 89 références classées correctement et 30 références classées comme incorrectes. Et pour la catégorie «Références bibliographiques entrelacées», nous obtenons une «accuracy» sur le jeu de test de 79,0% ce qui correspond à 156 références classées correctement et 44 références classées comme incorrectes. Nous pouvons noter que les résultats les plus faibles sont obtenus pour la catégorie «Notes de bas de pages particulières» ce qui s'explique par la faible quantité de paragraphes comportant des références bibliographiques (seulement 4,3%). Les deux autres catégories obtiennent des résultats similaires ce qui s'explique par une proportion égale de paragraphes comportant des zones bibliographiques comparativement à ceux n'en comptant pas.

#### **4.2. Utilisation des CCRFs pour l'annotation des références bibliographiques**

Pour construire nos CCRFs, nous utilisons plusieurs caractéristiques afin de construire nos vecteurs d'informations. Les principales caractéristiques exploitées dans la littérature pour l'annotation automatique de références bibliographiques sont fondées sur un certain nombre d'observations tel que des caractéristiques lexicales ou morphologiques, à la fois, sur les champs et sur les mots contenus dans les champs. Afin d'élargir notre étude, nous avons également étudié les caractéristiques utilisées dans la détection des entités nommées qui est une tâche connexe. Faisant un parallèle entre la tâche de détection des entités nommées et l'analyse des références bibliographiques, nous sommes en mesure d'extraire des informations plus utiles dans la caractérisation des champs et des mots contenus dans les champs. La typologie des caractéristiques mise en place est la suivante :

1) **Caractéristiques contextuelles** : exploitation des mots ainsi que des catégories syntaxiques précédant et suivant le mot courant. Suite à des expériences simples, nous retenons comme caractéristiques, les trois mots précédents et suivants le mot courant ainsi que les deux catégories syntaxiques précédents et suivants le mot courant.

2) **Caractéristiques locales** : elles se divisent en quatre sous-catégories : les caractéristiques morphologiques, locationnelles, lexicales et syntaxiques. Les caractéristiques morphologiques permettent de définir la forme des mots (Ex : majuscule/minuscule). Les caractéristiques locationnelles définissent la position des champs dans la séquence. Les caractéristiques lexicales correspondent à l'exploitation de listes de mots prédéfinis. Enfin, les caractéristiques syntaxiques réfèrent à la ponctuation.

À partir de ces caractéristiques nous construisons des vecteurs d'informations pour chaque mot. La combinaison de ces caractéristiques nous permet d'établir des vecteurs d'informations robustes pour la définition de nos différents champs. La construction des CRFs est réalisée par l'outil *Wapiti*<sup>20</sup> développé par (Lavergne *et al.*, 2010). Nous

---

20. <http://wapiti.limsi.fr>

utilisons un jeu de 13 labels établi en fonction des différents champs rencontrés lors de l'étude des références bibliographiques. Le tableau 2 présente les labels employés.

| Type          | Label               | Description                                                             |
|---------------|---------------------|-------------------------------------------------------------------------|
| Bibliographie | bibl                | zone bibliographique                                                    |
| Auteur        | surname<br>forename | nom de famille<br>prénom                                                |
| Titre         | title               | titre de l'article visé                                                 |
| Éditeur       | publisher           | éditeur                                                                 |
| Date          | date                | date, la plupart des années                                             |
| Place         | place               | nom de lieux                                                            |
| Édition       | biblscope           | informations sur les pages, volume, numéro, etc.                        |
| Détail        | extent<br>edition   | nombre total de pages<br>autres renseignements descriptifs de l'édition |
| Etc.          | abbr<br>punc<br>ref | abréviation<br>ponctuation<br>url                                       |

Tableau 2 – Descriptions des labels

Suite à la classification, nous obtenons une liste des paragraphes contenant potentiellement des références bibliographiques. Le premier CRF permet d'effectuer un premier filtrage afin d'identifier la zone dans laquelle les différents champs bibliographiques sont situés. Ce premier CRF permet de réduire le champ d'exécution du deuxième CRF via l'apposition de la balise <bibl> autour de la séquence contenant des champs bibliographiques. Le deuxième CRF nous permet d'annoter le contenu de la séquence détectée par le premier CRF via l'identification des différents champs bibliographiques, tels que, l'auteur, le titre, etc.

## 5. Expérimentations et résultats

Dans la section suivante, nous présentons les différentes expérimentations effectuées ainsi que les performances obtenues lors de la détection des zones bibliographiques et l'annotation de leurs contenus.

### 5.1. Détections des zones bibliographiques

Cette section présente les résultats obtenus suite à l'apposition de la balise <bibl> autour des séquences contenant des champs bibliographiques. Pour effectuer nos ex-

périmentations, nous établissons un corpus composé uniquement de paragraphes comprenant des références bibliographiques (Corpus de référence), c'est à dire, que ce corpus est établi indépendamment de la classification présentée précédemment. Ce qui nous donne pour la catégorie «Notes de bas de pages particulières» 30 paragraphes contenant des références bibliographiques, pour la catégorie «Références bibliographiques courtes» 158 paragraphes contenant des références bibliographiques et pour la catégorie «Références bibliographiques entrelacées» 311 paragraphes contenant des références bibliographiques. Nous avons fait ce choix afin de mesurer la robustesse des caractéristiques que nous avons choisies afin de constituer nos vecteurs d'informations. Nous présentons des expérimentations basées sur 10 validations croisées composées de 70% de corpus d'apprentissage et de 30% de corpus de test afin d'éviter un sur-apprentissage dû à une quantité trop importante de données d'apprentissage.

| Nom corpus                              | Corpus de référence |        |              |
|-----------------------------------------|---------------------|--------|--------------|
|                                         | précision           | rappel | F-mesure     |
| Notes de bas de pages particulières     | 72,2%               | 73,4%  | 72,8%        |
| Références bibliographiques courtes     | 94,1%               | 93,4%  | <b>93,8%</b> |
| Références bibliographiques entrelacées | 70,2%               | 72,6%  | 71,4%        |

Tableau 3 – Résultats pour la détection des zones bibliographiques

Le tableau 3 présente les résultats obtenus suite à la détection des zones bibliographiques pour chacune des sous-catégories. Nous observons que les résultats les plus faibles sont obtenus par la catégorie «Références bibliographiques entrelacées» et les résultats les plus significatifs par la catégorie de «Références bibliographiques courtes». Ce phénomène peut être expliqué par la complexité de la structure de ces dernières mais également par un degré d'implicite plus important que les références courtes. En effet, les «Références bibliographiques courtes» ont une structure beaucoup plus formelle que les «Références bibliographiques entrelacées». Les références courtes sont généralement utilisées afin de citer, de rapporter les mots ou les phrases de quelqu'un. Ce type de références est structuré plus formellement et répond à des conventions plus strictes tandis que les références entrelacées se fondent dans le corps du texte. Nous notons également que les catégories «Références bibliographiques entrelacées» et «Notes de bas de pages particulières» obtiennent substantiellement les mêmes F-mesures avec cependant une précision légèrement supérieure pour la catégorie «Notes de bas de pages particulières». Ce phénomène s'explique par la structuration similaire des références présentes au sein de ces deux catégories ainsi qu'un degré similaire d'implicite. Cette analyse nous permet de rendre compte de la difficulté qui réside dans la détection des zones bibliographiques plus le degré d'implicite est important. En effet, le degré d'implicite des références bibliographiques engendre des structurations et des répartitions dans le texte très variées ce qui rend l'homogénéisation des traitements difficile.

## 5.2. Détection des champs bibliographiques

Dans cette section, nous présentons les résultats obtenus suite à l'identification des différents champs bibliographiques. Pour cette expérimentation nous choisissons de présenter les résultats obtenus d'une part, sur un apprentissage effectué sur des modèles directement établi en fonction de chacune des sous-catégories et d'autre part, sur un apprentissage effectué sur un modèle établi sur le premier corpus<sup>21</sup> réalisé par le laboratoire OpenEdition. Pour rappel, le premier corpus d'OpenEdition est exclusivement destiné à l'apprentissage de modèles pour les références bibliographiques présentes dans les zones bibliographiques. Nous choisissons d'établir cette comparaison afin de constater l'impact des performances d'un modèle appris sur des données fortement structurées et de son application sur des données de nature différente. Cette analyse nous permet également de comparer notre approche aux approches qu'emploient les outils basés sur des CRFs présentées précédemment dans la section 2. Nous précisons également que cette expérimentation est faite indépendamment des résultats obtenus lors de la détection des zones bibliographiques. Les corpus utilisés ne découlent donc pas directement des résultats présentés lors de l'expérimentation précédente. Nous présentons des expérimentations basées sur 10 validations croisées composées de 70% de corpus d'apprentissage et de 30% de corpus de test afin d'éviter un sur-apprentissage dû à une quantité trop importante de données d'apprentissage.

| Nom corpus                                          | Corpus de référence |        |              |
|-----------------------------------------------------|---------------------|--------|--------------|
|                                                     | précision           | rappel | F-mesure     |
| Notes de bas de pages particulières                 | 86,2%               | 80,1%  | 82,9%        |
| Références bibliographiques courtes                 | 89,1%               | 88,8%  | <b>89,1%</b> |
| Références bibliographiques entrelacées             | 79,7%               | 65,6%  | 71,8%        |
| Notes de bas de pages particulières<br>Corpus 1     | 69,1%               | 62,4%  | 65,5%        |
| Références bibliographiques courtes<br>Corpus 1     | 69,0%               | 73,8%  | 71,2%        |
| Références bibliographiques entrelacées<br>Corpus 1 | 34,2%               | 27,5%  | 30,5%        |

Tableau 4 – Résultats pour la détection des champs bibliographiques

Le tableau 4 nous permet d'observer une nette dégradation des performances suite à l'apprentissage effectué sur le premier corpus d'OpenEdition. Ceci s'explique par la structuration très formelle que l'on retrouve dans les références bibliographiques présentes dans les zones bibliographiques. En effet, ce type de références répondent à des conventions très strictes. Il est intéressant de noter que les références courtes

21. Dans le tableau 4 se corpus est présenté avec l'extension Corpus 1

obtiennent les meilleures performances suite à l'apprentissage sur le premier corpus d'OpenEdition. Nous expliquons ce phénomène par le fait que ce type de références est structuré plus formellement et répond à des conventions plus strictes à la différence des deux autres catégories. Cette expérience nous permet de mettre en évidence une perte importante de performance lorsque l'on change le contexte applicatif sur lesquels les modèles ont été appris. Concernant les performances obtenues suite à l'apprentissage effectué sur des modèles directement établi en fonction de chacune des sous-catégories, les résultats les plus faibles sont obtenus pour la catégorie «Références bibliographiques entrelacées». Ce phénomène peut s'expliquer par la diversité des champs bibliographiques que l'on peut trouver dans cette catégorie associé à un degré d'implicite important. À l'inverse, la catégorie «Références bibliographiques courtes» obtient les meilleurs résultats ce qui est due à la présence de champs bibliographiques plus récurrent. En effet, nous dénombrons une moyenne de cinq champs bibliographiques différents pour la catégorie «Références bibliographiques courtes» tandis que pour la catégorie «Références bibliographiques entrelacées» nous recensons douze champs bibliographiques différents.

## 6. Conclusion

En conclusion, nous avons pu observer des résultats très variables concernant la classification supervisée. Ce phénomène est induit par les quantités de données d'apprentissage très inégales selon les sous-catégories suite à l'échantillonnage aléatoire des revues pour établir les différents corpus. Concernant la détection des zones bibliographiques, nous avons pu rendre compte de la difficulté qui réside dans la détection des zones bibliographiques plus le degré d'implicite est important. Nous avons pu noter que le degré d'implicite des références bibliographiques engendre des structurations et des répartitions dans le texte très variées ce qui rend l'homogénéisation des traitements difficile. Concernant la détection des champs bibliographiques, les expériences effectuées ont permis de mettre en évidence une perte importante de performance lorsque l'on change le contexte applicatif sur lesquels les modèles ont été appris. Lors de cette même étude, nous avons pu observer que les champs bibliographiques qui composent les références bibliographiques sont très variés selon les sous-catégories. Les différentes expériences menées sur les références bibliographiques diffuses nous permettent de constater d'une part, que l'identification des zones bibliographiques est une tâche complexe selon le degré d'implicite de la référence et d'autre part que, la nature des champs bibliographiques qui composent ces références bibliographiques diffuses est très variés.

Dans nos futurs travaux, nous comptons trouver une autre granularité que les paragraphes afin de réduire le champ d'action des CRFs. Nous envisageons d'établir des corpus mieux proportionnés mais également plus représentatifs de tous les cas possibles. Nous comptons choisir des caractéristiques spécifiques à chaque sous-catégorie pour améliorer les performances lors de l'annotation que ce soit pour améliorer la détection des zones bibliographiques que pour l'annotation de leurs contenus. À court terme, notre but s'oriente sur la réalisation d'une chaîne de traitement complète dans

laquelle nous allons intégrer un SVM multiclasse qui guidera directement le choix du premier CRF. Nous rappelons que la finalité de ce travail consiste en la mise en œuvre d'un système de recommandation de lectures dans lequel nous comptons utiliser les références bibliographiques afin d'établir des liens entre les documents.

## 7. Bibliographie

- Anzaroot S., McCallum A., « A new dataset for fine-grained citation field extraction », *ICML Workshop on Peer Reviewing and Publishing Models*, 2013.
- Bellis N. D., *Bibliometrics and citation analysis : from the science citation index to cybermetrics*, Scarecrow Press, 2009.
- Chen C.-C., Yang K.-H., Chen C.-L., Ho J.-M., « Bibpro : A citation parser based on sequence alignment », *Knowledge and Data Engineering, IEEE Transactions on*, vol. 24, n° 2, p. 236-250, 2012.
- Councill I. G., Giles C. L., Kan M.-Y., « ParsCit : an Open-source CRF Reference String Parsing Package. », *LREC*, p. 28-30, 2008.
- Cronin B., *The Citation Process : The Role and Significance of Citations in Scientific Communication*, Taylor Graham, London, 1984.
- Deconinck S., « Artificial intelligence a modern approach », 2010.
- Giles C. L., Bollacker K. D., Lawrence S., « CiteSeer : An automatic citation indexing system », *Proceedings of the third ACM conference on Digital libraries*, ACM, p. 89-98, 1998.
- Huang I.-A., Ho J.-M., Kao H.-Y., Lin W.-C., « Extracting citation metadata from online publication lists using BLAST », *Advances in Knowledge Discovery and Data Mining*, Springer, p. 539-548, 2004.
- Kim Y.-M., Bellot P., Faath E., Dacos M., « Automatic annotation of bibliographical references in digital humanities books, articles and blogs », *Proceedings of the 4th ACM workshop on Online books, complementary social media and crowdsourcing*, ACM, p. 41-48, 2011.
- Kim Y.-M., Bellot P., Faath E., Dacos M., « Annotated Bibliographical Reference Corpora in Digital Humanities. », *LREC*, p. 329-340, 2012a.
- Kim Y.-M., Bellot P., Tavernier J., Faath E., Dacos M., « Evaluation of BILBO reference parsing in digital humanities via a comparison of different tools », *Proceedings of the 2012 ACM symposium on Document engineering*, ACM, p. 209-212, 2012b.
- Lavergne T., Cappé O., Yvon F., « Practical very large scale CRFs », *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, p. 504-513, 2010.
- Lopez P., « GROBID : Combining automatic bibliographic data recognition and term extraction for scholarship publications », *Research and Advanced Technology for Digital Libraries*, Springer, p. 473-474, 2009.
- Rezaei B. A., Muntz A. H.-Y. M., « System and method for context-based knowledge search, tagging, collaboration, management, and advertisement », February 19, 2013. US Patent 8,380,721.
- Thorsten J., Making large scale SVM learning practical, Technical report, Universität Dortmund, 1999.





Papier absent

Papier absent

*Romain Giot, Romain Bourqui, Nicholas Journet, Vialard Anne*

Papier absent

Papier absent

*Romain Giot, Romain Bourqui, Nicholas Journet, Vialard Anne*

Papier absent



---

# Was it better before? Automated Quotation Detection in Ancient Texts

*Evaluating today's approaches on yesterday's content*

**Samuel Gesche, Előd Egyed-Zsigmond, Sylvie Calabretto**

Univ Lyon  
INSA de Lyon, LIRIS  
7 av Capelle  
69100 Villeurbanne  
Email: {prénom.nom}@liris.cnrs.fr

---

*RESUME.* Cet article s'intéresse à l'application des travaux en recherche de citations dans le contexte de documents anciens en langue grecque. La notion de citation est définie dans le contexte de ces documents, et les approches automatiques –statistiques et sémantiques– qui permettent de les découvrir sont évaluées à la lumière de cette définition et des ressources disponibles. Nous étudions également les effets spécifiques à notre corpus sur les métriques de performance.

*ABSTRACT.* This work evaluates quotation finding approaches in the context of ancient Greek documents. We explore the notion of quotation as relevant to the first centuries of our era, and we discuss the efficiency and usability of unsupervised approaches, both statistical and semantic, used in more modern cases. We also discuss some characteristics of our corpus relatively to performance metrics.

*MOTS-CLES :* détection de citations, corpus antique, rappel, évaluation

*KEYWORDS:* quotation detection, ancient corpus, recall, evaluation

---

## 1. Introduction

Detecting quotations and references has become the focus of numerous projects, and the applications of this field of research are as various as detecting plagiarism, studying the influences between authors or artists, or even predicting news spread across the Internet.

This work stems from the encounter between two fields: Patristics, which studies ancient documents and the quotation relationships between them, and Information Retrieval, which uses digital algorithms to match queries and documents. The story may have been that simple: the ancient documents have been digitalized, they are in a relatively homogeneous format, and the issue is a known one: detecting quotation between them. Moreover, the quotations were mostly *already* detected, so the issue was merely to find their exact boundaries.

In this paper, we present the challenges brought by quotations in ancient documents. We discuss the definition of a quotation in the context of Antiquity as opposed to the modern definition, and we try to map the modern approaches towards the ancient material. We also present the challenges brought by the corpus itself, and how it impacts the usual performance metrics such as precision, recall and f-measure. Our work has been mainly directed towards statistical linguistics, but we also discuss statistical semantics based on a few results.

## 2. Related Work

### 2.1. Related fields

Quotation detection is a research field that grew increasingly important during the last years, benefiting from the more large-scale interest for text reuse and plagiarism detection that is crucial to fields such as the patent business or copyright enforcing.

One great asset of this field is that it can be considered as a specific subfield of information retrieval, using any part of a whole document as a query. Of course, using such large-scale querying means that specific measures must be taken to ensure that the answers can be found in a reasonable amount of time; however, both the approaches and the metrics (notably precision, recall and f-measure) can be reused to a certain extent, as long as the core of the problem remains similarity detection (a process for which (Lukashenko *et Al.*, 2007) or (Bao *et Al.*, 2006) recall most approaches and paradigms).

However, this means that plagiarism and quotation detection also suffer from the same difficulty as information retrieval, which is the translation of the query -here some part of a text- into a form that will allow finding relevant documents -those that quoted that part of text- in a large corpus. Therefore, if the detection of literal quotations is as easy as matching words, in most cases more advanced strategies are required. There is a gradation between literal, erroneous literal, literary or



paraphrased quotations and allusions (references that sometimes require large context awareness) that is not unlike the gradation between lexical, syntactic, semantic and pragmatic levels of analysis. Plagiarism detection has shown advances, for example in paraphrase detection ((Faisal *et Al.*, 2012), (Jo *et Al.*, 2007)), but even then relies more on knowledge of the actor's mindset -the psychological mechanisms of plagiarism and paraphrasing- than on crude text analysis heuristics. There is also the case where text reformulation merely serves the purpose of merging the quotation within a new discourse, and the work is far easier then. (Ernst-Gerlach *et Al.*, 2008) work on this kind of reformulation, including errors, word addition, deletion or change, and language evolution. Lastly, semantic approaches emerge as semantic resources are made available. For instance, (Nawab *et Al.*, 2012) uses synonymy to enhance n-grams overlap detection.

Automatically detecting allusions is still very difficult unless specific semantic or pragmatic resources are tailored to the task. *Meme* (recurrent ironic representation) tracking on the Internet (Leskovec *et Al.*, 2009) may be the beginning of an answer, even though it focuses on quick mapping of the spread of a widely-used, quickly mutating formula and not on detecting less widely used formulae over a large time frame.

## 2.2. Quotation detection in antique texts

(Ernst-Gerlach *et Al.*, 2008) proposes an approach for discovering references in a Latin text corpus. This work provides a typology for the text differences inherent to the mechanics of referencing (regular and irregular differences, deletions, insertions and substitutions). The proposed method assumes the availability of simple resources such as proper nouns (characters, authors, ethnic groups, places *etc.*) and number format variations (digits, letters, abbreviations). Their algorithm is based on the concept of sliding overlapping windows and is very tolerant, two identical words within a window being sufficient to assert a quotation. The method is tested on quotations taken from a Latin dictionary (basically using the examples of an entry as the quotations to find), and these quotations are looked for in the Perseus corpus<sup>1</sup>. The results have a good recall but a varying precision. Among the reasons for the lack in precision is the impact of short words or that of not taking word order into account. They suggest taking into account the amount and frequency of terms, their orders and stop-words. The expected results are also far worse on short quotations (especially one word quotations, which are more than half their corpus).

(Lee J., 2007) takes another approach to antique text and provides many elements on the specificities of these texts and their languages. Applying cosine similarity to verses of the Greek New Testament, and consolidating highly similar verse groups, this work is able to find many quotations within the three synoptic

---

<sup>1</sup> Perseus Digital Library, <http://www.perseus.tufts.edu>, accessed on 2013-02-07, updated in 2013. This is the digital library of Tufts University, containing 69M words worth of resources in Ancient Greek and Latin. The 'Perseus Hopper' provides not only the resources, but also text processing tools to work on them online.

Gospels. One of the main hypotheses is that quotations follow the same text order in both the quoted and the quoting texts. The approach is evaluated against nine quotation tables obtained from nine different experts, and the block approach is thus validated empirically.

(Büchler M. *et Al.*, 2010) studies text reuse for the purpose of discussing the available versions of the texts. This work explains the difference in the practice of quotation between Antiquity and nowadays (and in particular the absence of the explicit reference). The work also adds a visual analysis over the text processing aspect, which is enhanced itself using the Perseus lemmatizer. The described algorithm is based on n-grams and prefix filtering and aims at finding similar areas in the texts. It then performs a more semantic analysis using significant words co-occurrence in sentences. The approach is able to take into account language evolution and dialect change, as well as word omission and substitution. However, it cannot discriminate between quotations and large idiomatic expressions, and is not good at finding the exact quotation boundaries.

Recently there have been methods using machine learning and neural networks to create numerical (vector) representations of words (Mikolovet *al.*, 2013-1; Mikolovet *al.*, 2013-2) and short texts (Le *et al.*, 2014) based on a large corpus. We are studying them.

### 3. Project Context

The purpose of our research is to find references of a work within another. Our focus is on ancient (*Koiné* Greek) documents that reference both the Tanakh and the New Testament (which will eventually be known together as the Bible). We do not delve deeply into the issue of versioning, although it is a really relevant problem in this research field, and one we had to clear beforehand (we still present it among the challenges we had to overcome).

```
Quis dives salvetur, by Clement of Alexandria
...
chapter 42
paragraph 1
line 1: Ἵνα δὲ ἐπιθαρρήσῃς οὕτω μετανοήσας ἀληθῶς ὅτι σοὶ μένει
line 2: σωτηρίας ἐλπίς ἀξιοχρεῶς ἄκουσον μῦθον οὐ μῦθον ἀλλὰ ὄντα
line 3: λόγον περὶ Ἰωάννου τοῦ ἀποστόλου παραδεδομένον καὶ μνήμη πεφυλαγμένον
paragraph 2
line 1: ἐπειδὴ γὰρ τοῦ τυράννου τελευτήσαντος ἀπὸ τῆς Πάτμου
...
```

Figure 1 - Example of a patristic structured text (beginning of chapter 42 of 'Quis Dives Salvetur' by Clement of Alexandria)

Our research corpus is composed of the Septuagint (the *Koiné* translation of the Tanakh that was available at the time), the Greek New Testament, and around 700 works, mainly of Church Fathers, which amounts to approximately ten million words.

Documents are structured and therefore can be referenced easily. Biblical texts are structured as books/chapters/verses, and patristic texts as work/chapter/paragraph/line (we discarded the page structure because it depends too much on which edition is used). This allows a single word to be located unambiguously. Figure 1 shows an example of patristic text with its logical structure.

## **4. Challenges**

### **4.1. *Koiné* Greek**

*Koiné* Greek is a language that gained relevance following the conquests of Alexander the Great, went global during the hellenistic period and the Roman empire, was supplanted by Latin around 300 A.D., and developed further within the Byzantine empire, becoming the much different Medieval Greek in the process.

*Koiné* Greek was therefore the global language during the first centuries in the Roman Empire, which means that it was constantly evolving to meet new needs, new markets and new concepts, much like English nowadays. This means that *koiné* Greek is really different from both the former Classical Greek and, of course, the Modern Greek. The consequence is that semantic resources tailored to either classical or modern Greek cannot be used easily, if at all.

There are, however, some things that remained constant:

- The alphabet did not evolve much: as a matter of fact, the Unicode standard for polytonic Greek works for both classical and *koiné* Greek (we will point out however that there are still other encoding standards, which still causes conversion issues);

- The language itself remains based on inflection, which means that the word order within a sentence bears no importance, at least concerning the meaning of the sentence (this means in particular that a quotation can reorder words); unfortunately, we have to add here that in most antique manuscripts, there is no visual indication of sentence beginning or end. Depending on the digital source, we thus may or may not be able to split the text into sentences.

### **4.2. *Quotation in the Antiquity***

Quotation finding, and more generally finding any kind of text similarities between documents (including, of course, plagiarism), is a research field that still mostly applies to quotation, and text similarity, as it is defined today. We quickly realized that, in order to work on ancient documents, we had to define what a quotation was at the time.

Quoting other people -be it written text, vocal speech, gestures, or other means of communication- is intrinsically a part of the communication process. While the

core concept stayed mostly the same (which is, recalling or telling what someone has expressed), the actual process of quotation, and the final result, has evolved because of several tremendous revolutions. Ancient texts come after the development of written language (with oral tradition still being very strong and as likely to be quoted as written text), but since then, the world has seen the development of the printing press, which made copy of written texts available for every scholar and every library, and the digital revolution, which made accessing texts, copying, quoting, transforming and broadcasting them trivial, to the point where the concept of document itself may not be clear anymore (Pédauque R. T., 2007).

In the Antiquity, texts (physically tablets, parchments, papyri or codexes) were not as readily available as nowadays. Copying was a long -and therefore expensive- process. However, during the first centuries of our Era, road were mostly secure and travel was quick along the commercial axes, so travelling to read a specific book - and to copy some parts of it- was possible, as was transporting a specific text (apostles' epistles, for instance, initially were transported from community to community to be read, while the Great Library of Alexandria, as well as numerous others, were busy with scholar visitors).

Taking this context into account, we can divide references into three types: literal quotations, non-literal quotations, and allusions.

- Literal quotations -today a portion of text put between brackets that must be lexically *strictly identical* to the original text- was at the time merely *exactly corresponding* to the original text, as a result of the writer either having the original text before his eyes, or having a faithful memory of the text. In other words, an ancient literal quotation says exactly what the original says, but not necessarily with the same words.

- In contrary, non-literal quotations are quotations that do not strictly correspond to the original text; either because of a partial memory, or by a deliberate change.

- References are not quotations *per se*, or they are implicit quotations -or cross-references. While lexically they do not contain any form of similarity to the original text, they do reference this text. Often, an analysis at semantic level is not sufficient to detect such a reference: the proper level is the pragmatic level. This is for example where the use of that very characteristic formula leads to considering that this author has been influenced by some work of this other author -which can be a material for thesis rather than automated digital process.

If we were to provide a single relevant number, it would be that 20% of the references we worked on do not have even a single lemma in common (yet lemmatization drops this number from an initial 43%).

What remains clear in that context is that searching for lexical similarities, while helpful, will not be nearly as successful as in a modern context. And since semantic resources that may be available for Modern Greek are irrelevant for *Koiné*, to use any approach by this angle we had to build them first.

### 4.3. Versioning

We will not present this challenge in details, since there is little we can do about it: these issues must be solved by the researchers in Humanities. They do, however, put a hard limit on our ability to find some references, so we will mention them briefly.

There are mainly two aspects of versioning:

- Present versioning, or the issue of the availability today of the documents that were quoted yesterday;
- Past versioning, or the knowledge of which of the then available versions of the document was quoted.

### 4.4. Available resources

#### 4.4.1. Lemma and morpho-syntactic analysis

The available resources were mainly lemma and morphological analysis for a number of word forms in ancient Greek. We used three main sources: Perseus, initially through the Archimedes Project Morphology Service<sup>2</sup>, then through their database dump available for download; BibleWorks<sup>3</sup>, a Bible-focused software who provided an export for morphologically analysed texts; finally, the lemma tables of Sources Chrétiennes<sup>4</sup>, built over the last years. In total, we got over 380 000 lemmatized ancient Greek forms, discounting homonymy and polysemy.

Without a morphological analyzer for our own texts, and without a morphological analysis accompanying the lemma tables of Sources Chrétiennes, we decided to limit our language processing to matching a term with a lemma from these lists. Of course, we were then vulnerable to the issues of homonymy and polysemy. We thus took a priority rule as follow: first, if the term is already a lemmatized form (that is, if one of the lists has the form identical to one of its possible lemmas), the form itself is kept. If it is not the case, we take the first lemma present in these tables for the form, prioritizing the tables according to their focus to Patristics:

- Sources Chrétiennes first, with 21 505 lemmatized terms in *koiné* Greek;
- then BibleWorks, with 115 498 lemmatized terms in *koiné* Greek;
- finally Perseus and its 318 584 lemmatized terms spread over multiple forms of ancient Greek language, including *koiné*;

All in all, we were able to find a lemma for 40% of the forms of our entire text corpus (however, several texts were entirely lemmatized, and we experimented on them), and 24% of the available lemmatized forms were present in the text corpus

---

<sup>2</sup> <http://archimedes.mpiwg-berlin.mpg.de/arch/doc/xml-rpc.html>

<sup>3</sup> <http://www.bibleworks.com/>

<sup>4</sup> <http://www.sources-chretiennes.mom.fr/>

(78% if we discount Perseus due to its language spread). Figure 2. <sup>(CORIA)</sup> presents the distribution of the lemmatized forms within the different sources, and the contribution of these sources to the lemmatization of the forms present in the corpus.

As a side effect, we got the list of around 5 000 proper nouns used in the Septuagint.

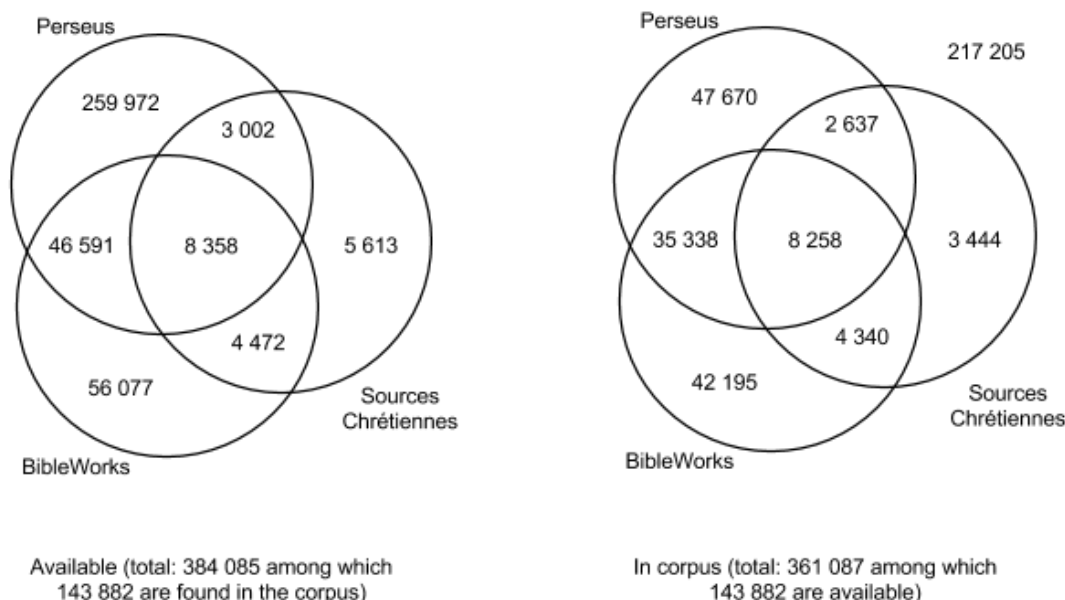


Figure 2. Distribution of the available and necessary lemmatized forms among the three sources Perseus, BibleWorks and Sources Chrétiennes (left: forms available for lemmatization; right: forms both available and actually present in our corpus - the outer number represents the amount of forms for which none of the sources provide a single lemma)

Automated lemmatization (Dimitrios *et Al.*, 2008) may greatly enhance these results but we have not delved into it currently, since it was not the focus of our research. The same is true for stemming, which we could not achieve with simple approaches (Greek is a not an easy language to stem), and for which we could not find lists of already stemmed forms.

#### 4.4.2. Stopwords and other recurring terms

We have a very narrow list of stopwords, the experts having determined that a larger list did not reflect the list of basically disposable terms. We have 16 lemma in this list (οὐν, τε, ό, καί, δε, γάρ, τίς, ή, μέν, μήν, δή, οὗς, γέ, ἄρα, αὐτός, πότε). Figures 3 to 6 show this list to be sufficient (there are not many common words left between the references in the control set).

Using a table of the most frequent terms, we built two more lists, a list of recurring terms in the language (111 lemma such as ὁσιος - hallowed), and a list of recurring terms in the corpus (870 lemma such as κύριος - lord). These lists allow us

to better adjust the sensibility of our algorithm, and better, to use each list at the relevant steps (for instance, a stopword can be eliminated from the start, but recurring words will typically be used to discard results that would only contain them).

On top of that, we listed recurring expressions as well (24 in the language and 202 in the corpus).

We called those recurring formulae, terms or expressions nonquotes. While they do contribute to the meaning of the text, and as such cannot be simply discarded, their presence cannot be used alone as an argument to support a quotation.

#### *4.4.3. Semantic resources*

Semantic approaches are greatly limited as long as they rely on resources that are highly language-dependent. However, statistico-semantical approaches remain usable as long as their underlying hypotheses on language are valid for koiné Greek. (Büchler *et Al.*, 2010) actually uses the statistico-semantical tool of significant terms co-occurrence, and (Ernst-Gerlach *et Al.*, 2008) uses names as hints for a greater chance of quotation. Other semantic analysis methods such as LSA may yield results too.

### **4.5. Corpus-specific challenges**

#### *4.5.1. Boundaries detection and precision*

Precision is defined as the proportion of found answers that are good answers, and recall as the proportion of good answers that are found. In our case, however, asserting whether a found quotation is a good result is difficult, as is asserting whether a quotation was found.

Firstly, we have a list of results to find -results that are taken from already published material. These results are, however, often approximate, to the point where one of our goals is to find the actual boundaries of the quotations within them. We know where these results begin, and often not where they end. So, how do we decide that a quotation is good? Fortunately, in our small test sample (329 references total, including allusions that do not have any word, lemma or otherwise in common), a manual work has been done to ensure that we know the boundaries at the granularity of words. But how is this number sufficient to assert the performance of our algorithm?

We eventually decided that if a quotation that we found overlaps with a quotation that must be found, even with a single word, we will take it as a good result. Obviously, overlapping must occur in both texts.

Secondly, another of our aims is to find new quotations -quotations that have not been detected, or documented, in the previous centuries. Digital processing allows being exhaustive, if biased by the limits of an algorithm, and we already found several new quotations during our tests. While recall does not suffer from this aspect

of the project, we have to first pass the results to a second process -in this case, a manual study- to assert whether a false result (as defined by the control list) is actually a true negative that should enrich the original list, thus irremediably corrupting the independence between the expected results and the algorithm that we want to test: we can only include the true negative if we find them, which leads to an artificial increase in performance that does not reflect an actual improvement. We cannot artificially enhance its performance by biasing the test sample.

We can of course tag these results as ‘good but not mandatory’, thus keeping the independence, and simply not count them. But it still means that we have to manually control the results before asserting a precise precision score.

#### 4.5.2. Self-quotation and recall

Lastly, we have a specific bias that comes from the very text we study: the Bible is a web of self-quotations, with sometimes more than 20 candidate passages for a single quotation. In this case, which of the texts is the right one? The one in the list of results is, obviously. But can we hold an algorithm -that basically matches strings- accountable for deconvolving the strata of successive quotations and finding the right author that was quoted, as opposed to those he quoted and those who quoted him? In many cases, a pragmatism analysis is necessary. In some extreme cases, manual research is necessary.

We first sought to differentiate real quoted text from mere recurring formulae - either in the language or in the specific corpus- and it brought us to expand the notion of stopwords to nonquotes as defined in Section 4.4.2. Then, we looked with the experts for a threshold for the amount of candidates; either there were less candidates, and they were all considered right, for needing a manual validation (we had actually quotations that were considered enriching by the experts, quotations that they wanted to have as results, even if not formally the right one); or the amount of candidates exceeded the threshold and they were all considered false, for having found the right answer for wrong reasons.

Intertwining both processes, searching for nonquotes in the numerous candidates, was very effective at building our lists of recurring expressions. The drawback, however, was the risk for them to become highly sensitive to the specific kind of text we were testing with, and to bias the results once more. This means that the search for recurring formulae must be integrated as a learning process in the algorithm itself.

We thus defined both a *raw* score -precision and recall using the amount of found answers- and an *efficient* score, factoring that last bias and the according threshold. The efficient score basically takes into account the fact that the results, in our case, will be studied by human experts.

It is worth noting that while in the first case a single number serves as the dividend of both precision and recall, in the refined version the dividend of precision is greater than the dividend of recall (factoring the numerous cases where more than one answer is deemed correct). Besides, the impact of this correction on the recall



depends on the amount of found answers (which normally only impacts precision). This impairs sets of parameters that increase the amount of candidates: in addition to being subject to low precision, they also get low recall. Empirically, we obtained a factor of 2/3 between raw recall and efficient recall with a threshold of 5.

## **5. Evaluation**

### **5.1. Algorithm**

In order to test the approaches that we presented in section 2, we designed a general algorithm based on the usual steps:

- Pre-processing, including documents import and encoding resolution, tokenizing, and optionally stopwords deletion and language processing.
- Main processing: matching word n-grams from both texts and computes the exact quotation boundaries.
- Post-processing: preparing the found quotations to match the quotation granularity of the control set, then computing the relevant metrics (both raw and efficient precision, recall and f-measure) as well as a more verbose report (allowing us to detect for instance which approach is better at finding some set of quotations).

During the main processing, we tested various approaches and heuristics, among which:

- (Ernst-Gerlach *et Al.*, 2008): overlapping windows algorithm, fundamentally equivalent to a n-gram approach where we search for 2 shared terms in a n-gram the size of the window, and then merge the overlapping found results; we also tested their heuristic of expanding the window whenever a proper noun is found; we did not test the impact of having different formats for numbers. We did test the impact of word order as discussed in their perspectives.
- (Lee J., 2007): merging two non-overlapping quotations if there were less than 50 words between each other. We did not however use the full consolidation process due to the difference between the corpuses (the method described in this work is especially effective in the studied case).
- (Büchler M. *et Al.*, 2010): using co-occurrence of significative words as a hint for a more semantic measure. We used the approach from (Mousselly-Sergie, H. *et Al.*, 2013) to infer the semantic similarity.

### **5.2. Experimental sets**

We defined two experimental sets to test the approaches. They differ both on their scale and on the accuracy of the control set of quotations.

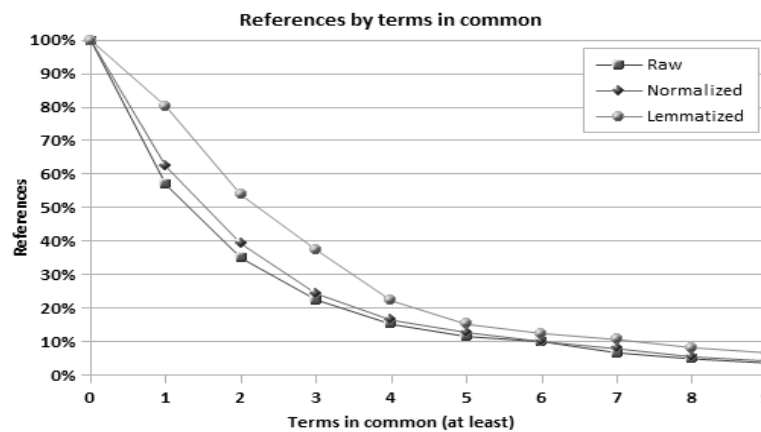


Figure 3. Amount of references to the Bible within *Quis Dives Salvetur* where both texts have at least a given amount of common terms (either the terms themselves, their normalized version, or their corresponding lemma). Stopwords are discounted. Raw: refers to the original text, Normalized to the normalized text, and Lemmatized to the text where words are reduced to their lemmatized form.

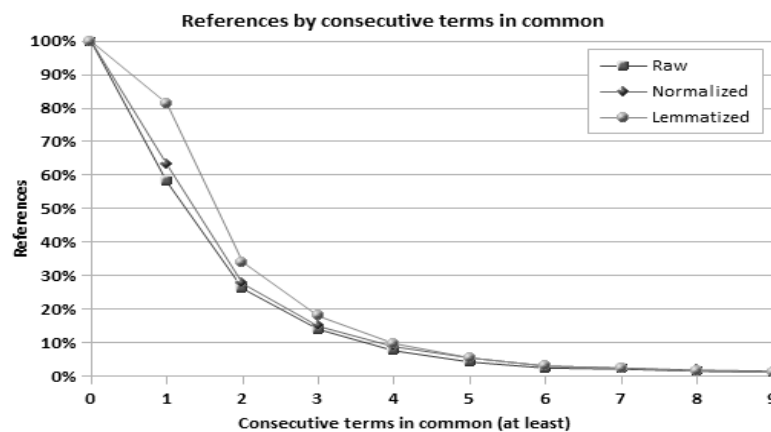


Figure 4. Amount of references to the Bible within *Quis Dives Salvetur* where both texts have in common at least a given amount of consecutive terms (either the terms themselves, their normalized version, or their corresponding lemma). Stopwords are discounted.

The first experimental set uses a single document, *Quis dives salvetur*, quoting the Bible as composed of the Septuagint and the Greek New Testament. To control our results, we have a narrow set of 329 references, the exact boundaries of which have been produced. These references include 19 allusions. The breakdown of the similarities between the quoting text and the quoted text are presented in Figure 3 (identical terms) and 4 (amount of consecutive identical terms).

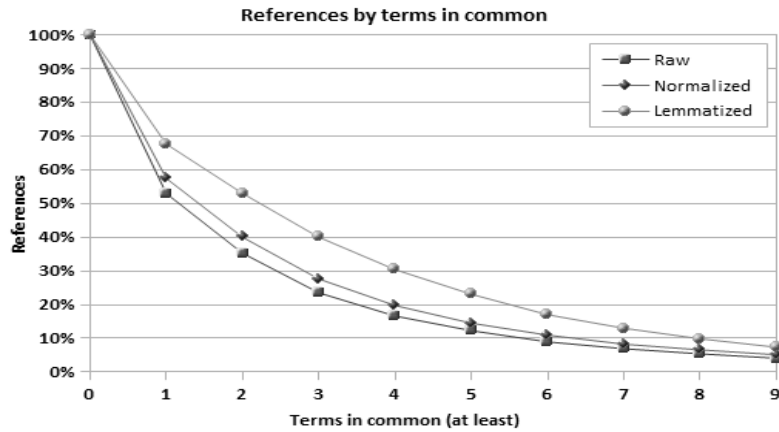


Figure 5. Amount of references to the Torah within the works of Philo of Alexandria where both texts have at least a given amount of common terms (either the terms themselves, their normalized version, or their corresponding lemma). Stopwords are discounted.

The second experimental set uses a whole author's publication (namely Philo of Alexandria), quoting the Septuagint. We only use the references to the Torah (the first five books of the Septuagint) as a control set, and we only know in which paragraph of the Philo texts the reference is. We do not know how many of them are allusions. The breakdown of the similarities between the quoting text and the quoted text are presented in Figure 5 (identical terms) and 6 (amount of consecutive identical terms). It is worth noting that this case shows even less similarities between the texts than the previous case, even though the quotations are more loosely defined.

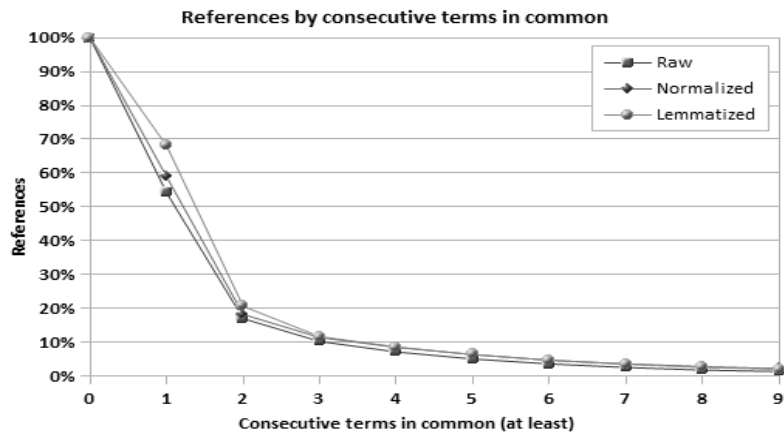


Figure 6. Amount of references to the Torah within the works of Philo of Alexandria where both texts have in common at least a given amount of consecutive terms (either the terms themselves, their normalized version, or their corresponding lemma). Stopwords are discounted.

### 5.3. Results

We run the algorithm on those sets several hundreds of times with varying parameters. Overall, we were able to find up to 30% of the references (without resorting to extremely loose settings). The results were comparable between both sets of experiments.

Some of the parameters had massive impact:

- Discarding stopwords had the most noticeable impact, always bringing better results (especially in the case of our effective scores). We quickly stopped experimenting with stopwords allowed, so the following results all include stopword deletion.

- Merging found references that were less than 50 words distant (in both texts) greatly increased precision and effective recall (limiting the amount of candidates for a given reference). We also quickly included this action by default in the experiments.

- The size of the n-grams (how many similar words have to be found) and the tolerance (how many differences are allowed between similar fragments) had obviously a definite influence on both precision and recall.

- The former set a compromise between them, smaller n-grams bringing lower precision and higher recall. However, the increase in effective recall receded quickly when the size of the n-grams went lower than 3, so we kept 3-grams as an optimal size, from the original 5 taken from (Ernst-Gerlach *et Al.*, 2008). The F-measure was actually better for larger n-grams, but the amount of results was too low to be of use (on the first set, a mere 20 results for 5-grams compared to several hundred for 3-grams).

- Tolerance showed a sizeable increase in recall (averaging 60% all other conditions equal) without affecting precision nearly as much (a mere 20% loss). Interestingly, the numbers are the same in the case of both the usual recall and our more constrained effective recall. Our optimal setup was searching for 3 common terms within 5-grams.

Among the parameters that had either low or negligible impact:

- Nonquotes filtering had no significant impact outside of experiments which did not discard stopwords (in which case the filtering was not enough to warrant interesting results).

- Using proper nouns as a hint to widen the frame had no discernible impact, probably due to the fact that few of the references that were not found displayed a name.

- Discounting word order had a larger impact on lowering precision than on increasing recall, especially with lemmatized text.

- Text processing, rather than simply improving the results, moved the compromise between precision and recall. Figures 3 to 6 show the beneficial impact of lemmatization in references, but the same impact is true outside of them. In

average, the results with effective precision and recall were better with unprocessed or normalized text.

– We only tested a single semantic approach, but the results were along the lines of text processing and n-gram size: the better the measure was at merging terms, the farther the results moved along the precision versus recall axis (and in the case of effective scores, towards lower overall scores).

These results tend to show the following features of our corpus:

– There is, as we showed in section 4.3, a proportion of quotations in which there are common terms, but these terms do not follow one another exactly. In other words, there will often be an edit distance between a source text and the quoting text.

– The general heuristic that stopwords limit the efficiency of statistical text matching holds true. However, there is no need to have a large list (since filtering other common words does not have a significant impact).

– The general heuristic that merging terms that have a close meaning (through text processing, semantic measures and other means) increases recall at the cost of precision also holds true. In our case, the benefit is not apparent, especially because we have constrained the recall to take into account a manual processing of the results and the highly self-quoting nature of the source document.

– Most of the references still cannot be found, even with highly permissive parameters. This is to be expected when half of them do not even have two lemma in common.

## **6. Conclusion and perspectives**

We discussed the issue of detecting quotations within ancient documents. We took into account the specificities of language, culture (through the practices of quotation) and of the corpus itself. Using several approaches that were used in similar cases, we evaluated the main tools offered by statistical computation and found that even though they were not a sufficient solution, they worked as usual in the case of quotations that show text similarities.

This work leaves us three options to increase our performance at detecting quotations in ancient Greek texts. The first two revolve around the heuristic of merging similar terms using statistical methods. We can use our partnership with Greek experts to build a stemmer for this language, and we can use new semantic algorithms to improve our similarity measure. Increasing the amount of common terms may allow us to increase the similarity threshold of quotation detection and improve precision. The third option is finding heuristics within the metadata that can be obtained on the texts themselves. For example, if we know that some text is a commentary of some book –and there are many such cases, we can rely on this information to resolve most cases of multiple source candidates.

Another perspective is the study of the efficiency and scalability of statistico-semantic methods. We are currently working on the implementation of methods

such as word2vec (Mikolov, *et al.*, 2013) and the improvement of memory efficiency (CORIA) of the algorithm of (Mousselly-Sergieh *et al.*, 2013) based on stream processing methods.

## 7. References

- Bao J.P. *et Al.* (2006). A fast document copy detection model. *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, vol. 10, n° 1, p. 41-46.
- Büchler M. *et Al.* (2010). Unsupervised Detection and Visualisation of Textual Reuse on Ancient Greek Texts. *Journal of the Chicago Colloquium on Digital Humanities and Computer Science*, vol. 1, n° 2, p. 1-17.
- Coulie B. (1996). La lemmatisation des textes grecs et byzantins : une approche particulière de la langue et des auteurs. *Byzantion : revue internationale des études byzantines*, vol. 66, p. 35-54.
- Dimitrios P. L. *et Al.* (2008). Applying similarity measures for automatic lemmatization: a case study for modern Greek and English?. *Proceedings of Int. J. Artif. Intell. Tools 2008*.
- Ernst-Gerlach A. *et Al.* (2008). Identifying quotations in reference works and primary materials. *Research and Advanced Technology for Digital Libraries*, vol. 5173, p. 78-87.
- Faisal A. *et Al.* (2012). Analysis and extraction of sentence-level paraphrase sub-corpus in CS education. *Proceedings of ACM SIGITE 2012*.
- Jo C. *et Al.* (2007). New Functions for Unsupervised Asymmetrical Paraphrase. *Journal of Software*, vol. 2, n° 4, p. 12-23.
- Le, Q., & Mikolov, T. (2014). Distributed Representations of Sentences and Documents. In *Proceedings of The 31st International Conference on Machine Learning* (pp. 1188–1196). Retrieved from <http://jmlr.org/proceedings/papers/v32/le14.html>
- Lee J. (2007). A computational model of text reuse in ancient literary texts. *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*.
- Leskovec J. *et Al.* (2009). Meme-tracking and the dynamics of the news cycle. *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*.
- Lukashenko R. *et Al.* (2007). Computer-Based Plagiarism Detection Methods and Tools : An Overview. *Proceedings of the International Conference on Computer Systems and Technologies 2007*.
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *Computation and Language*. Retrieved from <http://arxiv.org/abs/1301.3781>
- Mikolov, T., Sutskever, I., 0010, K. C., Corrado, G. S., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. In *NIPS* (pp. 3111–3119). Retrieved from <http://dblp.uni-trier.de/db/conf/nips/nips2013.html#MikolovSCCD13>
- Mousselly-Sergieh, H. *et Al.* (2013). Tag Similarity in Folksonomies. *Proceedings of the XXXI INFORSID congress*, p 319-334.
- Nawab R. *et Al.* (2012). Detecting Text Reuse with Modified and Weighted N-grams. *Proceedings of the ACM First Joint Conference on Lexical and Computational Semantics 2012*.
- Pédauque R. T. (2007). *La redocumentarisation du monde*. Paris : Éditions Cepadues.



CORIA – CIFED 2016

Documents graphiques  
CIFED

Président : *XX*





Papier absent

Extraction de patterns successifs dans des images de document à partir  
de la combinaison d'histogrammes de forces et de droites discrètes

Papier absent

# Papier absent

Extraction de patterns successifs dans des images de document à partir  
de la combinaison d'histogrammes de forces et de droites discrètes

Papier absent

# Papier absent



Papier absent

Papier absent



# Papier absent

Papier absent

# Papier absent



---

# Using BLSTM for Interpretation of 2D Languages – Case of Handwritten Mathematical Expressions

**Ting ZHANG\*** — **Harold MOUCHERE\*** — **Christian VIARD-GAUDIN\***

\* *LUNAM/IRCCyN/IVC - UMR CNRS 6597*  
*Université de Nantes*  
*Nantes, France*

---

*RÉSUMÉ. Nous proposons une extension à l'utilisation classique des réseaux de type BLSTM pour leur permettre de traiter des données provenant de langages graphiques bidimensionnels tels que les formules mathématiques manuscrites. La solution proposée repose sur un parcours respectant l'ordre temporel des traits. Il en résulte une séquence alternant les étiquettes de symboles et les étiquettes des relations spatiales. Dans le cas des expressions purement linéaires (1-D), nous utilisons l'étiquette « Right » pour permettre la segmentation entre les symboles. Pour une extension au cas des expressions véritablement bidimensionnels (2-D), nous utilisons autant de nouvelles étiquettes qu'il y a de relations spatiales différentes entre les sous-expressions. Il en résulte que les réseaux BLSTM permettent de résoudre à la fois la tâche de reconnaissance de symboles et celle de segmentation. Une telle approche est nouvelle dans le domaine de la reconnaissance des expressions mathématiques.*

*ABSTRACT. In this work, we study how to extend the capability of BLSTM networks to process data which are not only text strings but graphical two-dimensional languages such as handwritten mathematical expressions. The proposed solution aims at transforming the mathematical expression description into a sequence including at the same time symbol labels and relationship labels, so that classical supervised sequence labeling with recurrent neural networks can be applied. For simple one-dimensional (1-D) expression, we use the Right label to segment one symbol from the next one, as with the standard blank label for regular text. For genuine two-dimensional (2-D) expressions, we introduce additional specific labels assigned to each of the different possible spatial relationships that exist between sub-expressions. As a result, BLSTM network is able to perform at the same time the symbol recognition task and the segmentation task, which is a new perspective for the mathematical expression domain.*

Using BLSTM for Interpretation of 2D Languages – Case of  
Handwritten Mathematical Expressions

*MOTS-CLÉS : Reconnaissance d'expressions mathématiques, écriture manuscrite, réseau récurrent, BLSTM.*

*KEYWORDS: Handwritten mathematical expression, online handwriting, recurrent neural network, BLSTM.*

---

## 1. Introduction

Online Handwritten Mathematical Expressions (MEs) recognition has been an active research field for years, especially after the rapid development of touch screen devices. There are two main sources of handwritten ME : offline (images of manuscript documents or video of blackboards) and online. We focus on online documents which come from sensitive screens (interactive whiteboards, smaller touch screens or tablets, etc.). An online document consists of one or more strokes which are sequences of points sampled from the trajectory of the writing tool between a pen-down and a pen-up. On-line handwritten ME recognition involves the automatic interpretation of these temporal sequence of 2-D points into structured sets of symbols. To realize these, both a large set of symbol classes and symbol arrangements in a two-dimensional layout need to be recognized. ME recognition still exhibits a big challenge to researchers.

Generally, ME recognition involves three interdependent tasks (Zanibbi et Blostein, 2012) : (1) Symbol segmentation, which consists in grouping strokes that belong to the same symbol ; (2) symbol recognition, the task of labeling the symbol to assign each of them a symbol class ; (3) structural analysis, its goal is to identify spatial relations between symbols and with the help of a grammar to produce a mathematical interpretation. These three problems can be solved sequentially or jointly. In the first case, the tasks are processed sequentially which means the error from the previous stage will be propagated to the next stage. The alternative solutions run these three tasks concurrently and aim at making a final decision of the formula using global information (Awal *et al.*, 2014 ; Álvaro *et al.*, 2014b ; Álvaro *et al.*, 2016). These approaches seem reasonable and exhibit out performances as these three problems are highly interdependent by nature. Human beings recognize symbols with the help of structure, and vice versa.

Current solutions process each task separately but need to consider context. Context is included to classical classifier (MLP, SVM, etc.) with specific features or language model (grammar). Bidirectional Long Short-term Memory (BLSTM) network naturally takes this context into account because it can access the contextual information from both the future and the past in an unlimited range. The advanced recurrent neural network BLSTM has been proved to outperform other classifiers in tasks like handwritten text recognition (Graves *et al.*, 2009), handwritten math symbol recognition (Álvaro *et al.*, 2013 ; Álvaro *et al.*, 2014a), printed text recognition (Ray *et al.*, 2015) and keyword spotting (Frinken *et al.*, 2014). However, the move from text recognition to mathematical expression recognition is far from being straightforward since ME has the structure of 2 dimensions. In this study, we explored ME recognition using BLSTM model, not simply as a symbol classifier exploiting a pre-segmented fragment as proposed in (Álvaro *et al.*, 2014a), but as a system able to label a global sequence, solving at the same time the segmentation and the recognition tasks.

This research domain has been boosted by the Competition on Recognition of Handwritten Mathematical Expressions (CROHME), which began as part of the International Conference on Document Analysis and Recognition (ICDAR) in 2011.

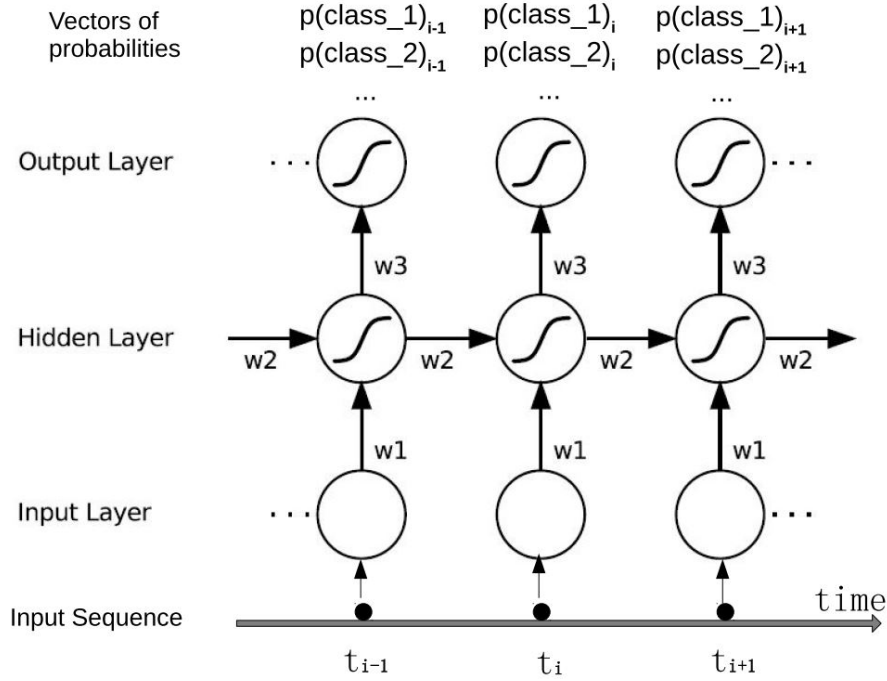


**Figure 1.** An unfolded single-directional recurrent network.

It provides a platform for researchers to test their methods and compare them, and then facilitate the progress in this field. It attracts increasing participation of research groups from all over the world. In this work, the provided data and tools will be used and results will be compared to participants.

The remainder of the paper is organized as follows. Section II introduces BLSTM architecture briefly. Section III describes the representation of ME. Section IV presents recognition strategies at stroke level. Features for on-line ME are showed in Section V. In Section VI, we report the experiments and results. Finally, conclusions and future works are presented.

## 2. BLSTM

Recurrent neural networks (RNNs) can use contextual information and therefore are suitable for sequence labeling (Graves *et al.*, 2012). In order to visualise the RNN and understand how it operates, we unfolded it along the input sequence and illustrated a part of the unfolded one in Figure 1. Here, each node represents a layer of network units at a single time-step. The input at a single time-step is a feature vector ; for the whole process, the input is a time sequence of feature vectors. The output at a single time-step is a probability vector of which each element is the probability of belonging to some class ; the overall output is a sequence of probability vectors. As can be seen, the output of each step depends on both the current input and the information from the



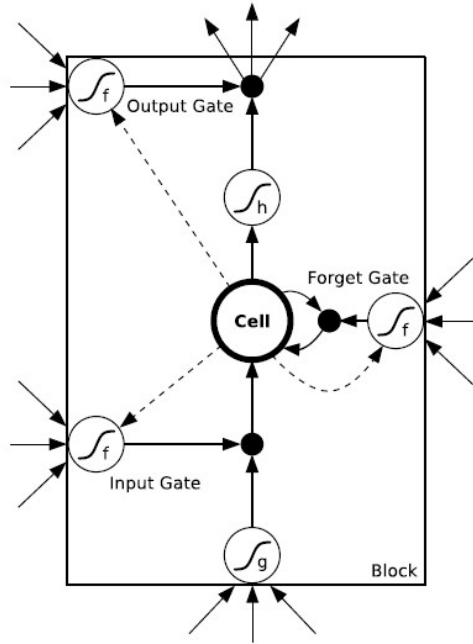


**Figure 2.** *LSTM memory block with one cell, figure extracted from (Graves et al., 2012).*

previous time-step. The same weights ( $w_1$ ,  $w_2$ ,  $w_3$ ) are reused at every time-step. The training process of RNN requires the ground-truth for each time-step because it needs to compute the error to do back propagation. With respect to recognition process, since the network only outputs local classifications, some form of post-processing need to be done.

An important benefit of RNNs is their ability to use contextual information. Unfortunately, for standard RNN architectures, the range of context that can be accessed is quite limited. Long short-term memory (LSTM) (Hochreiter et Schmidhuber, 1997) was proposed to address this weakness. An LSTM network is the same as a standard RNN, except that the summation units in the hidden layer are replaced by memory blocks, as shown in Figure 2. Each block contains one or more self-connected memory cells and three multiplicative units (the input, output and forget gates). The three gates collect activation from inside and outside the block and control the activation of the cell via multiplications. The input and output gates multiply the input and output of the cell while the forget gate multiplies the cell's previous state. The only output from the block to the rest of the network emanates from the output gate multiplication.

Standard RNNs process sequences in temporal order, as presented in Figure 1, which means they can only access the past context, ignoring the information from the future. However, future context is vital for recognition task likewise. An elegant solution to this problem is bidirectional recurrent neural networks (BRNNs) (Schuster et

Paliwal, 1997), which presents every training sequence forward and backward to two separate recurrent hidden layers. Replacing the summation units in the hidden layer of BRNN generates BLSTM (Graves et Schmidhuber, 2005). Combining the advantages of BRNN and LSTM, BLSTM offers access to long range context in two directions. In the nature of things, this advanced architecture outperformed other models in several tasks (Graves *et al.*, 2009 ; Álvaro *et al.*, 2013 ; Álvaro *et al.*, 2014a).

### 3. The Representation of ME

The recurrent neural network BLSTM processes each training sequence in temporal order (forward and backward). That means if we want to use the BLSTM model directly into ME recognition, each expression should be stored as a 1-D sequence. The proposed solution is to extract a reference path in a graph describing the ME. As a graph, we present in the next section how to model a ME with a primitive (stroke) Label Graph (LG).

#### 3.1. Primitive Label Graph

Structures can be depicted at three different levels : symbolic, object and primitive (Zanibbi *et al.*, 2013). In the case of handwritten ME, the corresponding levels are expression, symbol and stroke. It is possible to describe a ME at the symbol level using a Symbol Relation Tree (SRT), while if needed to go down at the stroke level, a primitive Label Graph can be derived from the SRT. In SRT, nodes represent symbols and labels on the edges indicate the relationships between symbols. Examples can be found in Figure 3. LG contains the same information as SRT but at the stroke level. In LG, nodes represent strokes, while labels on the edges encode either segmentation information or layout information. As relationships are defined at the level of symbols, all strokes in a symbol have the same input and output edges. Consider the simple expression '2 + 2' written using four strokes (two strokes for '+' ) in Figure 4a. The corresponding SRT and LG are shown in Figure 4b and Figure 4c respectively. As Figure 4c illustrates, nodes are labeled with the class of the symbol. A dashed edge corresponds to segmentation information ; it indicates that a pair of strokes belongs to the same symbol. In this case, the edge label is the same as the common symbol label. On the other hand, the non-dashed edges define spatial relationships between nodes and are labeled with one of the different possible relationships between symbols.

The spatial relationships as defined in the CROHME competition are : *Right*, *Above*, *Below*, *Inside* (for square root), *Superscript*, *Subscript*. For the case of nth-Roots, like  $\sqrt[n]{x}$ , we define that the symbol 3 is *Above* the square root and  $x$  is *Inside* the square root. The limits of an integral and summation are designated as *Above* or *Superscript* and *Below* or *Subscript* depending on the actual position of the bounds. In addition, the label ' \_ ' is used to denote that there is no relation between two symbols.

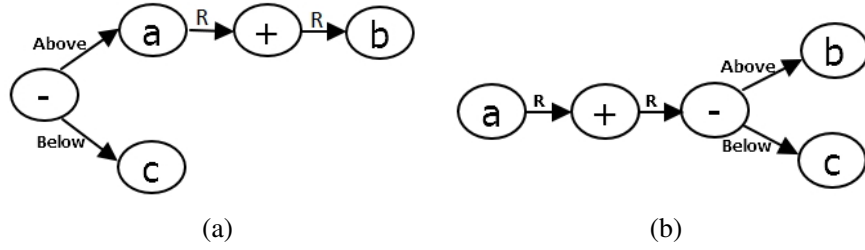


**Figure 3.** (a) the symbol relation tree of  $\frac{a+b}{c}$ ; (b) the symbol relation tree of  $a + \frac{b}{c}$ . 'R' is for left-right relationship.

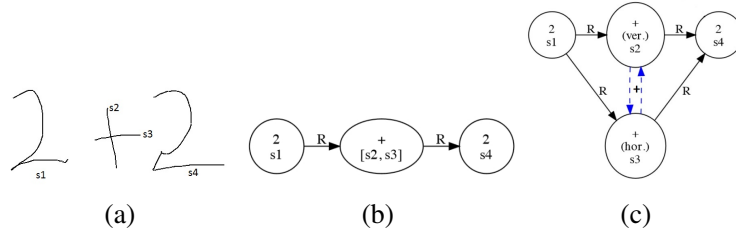


**Figure 4.** (a) '2 + 2' written with four strokes; (b) the symbol relation tree of '2 + 2'; (c) the label graph of '2 + 2'. The four strokes are indicated as s1, s2, s3, s4 in writing order. (ver.) and (hor.) are added to differentiate the vertical and the horizontal strokes for '+'. 'R' is for left-right relationship.

### 3.2. Storing expression as 1-D sequence

As mentioned, each expression should be stored into a 1-D sequence in order to be suitable for BLSTM. There are several solutions to generate a 1-D sequence from a tree but none of them allows to rebuild systematically the tree structure except Euler string which requires additional brackets. We propose to generate a 1-D sequence following the time sequence of strokes in LG to represent an expression. This time sequence is not always the best solution, but the simplest and the most intuitive by nature. From Figure 4c, the corresponding 1-D sequence is  $\{s1, edge_{s1\_s2}, s2, edge_{s2\_s3}, s3, edge_{s3\_s4}, s4\}$  labeled as  $\{2, R, +, +, +, R, 2\}$ , where  $edge_{s_i\_s_j}$  denotes the edge from  $s_i$  to  $s_j$ . This sequence alternates the node labels  $\{2, +, 2\}$  and the edge labels  $\{R, +, R\}$ . This proposal works well on linear expressions. For example, given  $\{2, R, +, +, +, R, 2\}$ , we can regenerate the correct LG with adding the edges from  $s1$  to  $s3$  and from  $s2$  to  $s4$  (following the rule that all strokes in a symbol have the same input and output edges), also the edge from  $s3$  to  $s2$ . It can also deal with a part of 2-D expressions. With  $P^{eo}$  shown in Figure 5a, the sequence is  $\{P, P, P, Superscript, e, R, o\}$ . All the spatial relationships are covered in it and naturally a correct LG can be regenerated. However, this kind of sequence

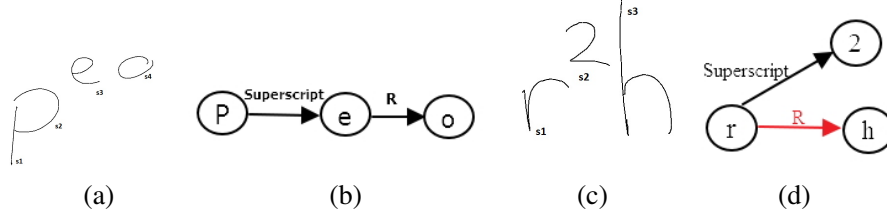


**Figure 5.** (a)  $P^{eo}$  written with four strokes ; (b) the SRT of  $P^{eo}$  ; (c)  $r^2h$  written with three strokes ; (d) the SRT of  $r^2h$ , the red edge can not be generated by the time sequence of strokes.

fails on a number of 2-D expressions. Figure 5c presents a failed case. According to time order, 2 and  $h$  are neighbors but there is no edge between them as can be seen on Figure 5d. In this case, we write the sequence as  $\{r, Superscript, 2, \_, h\}$ . The *Right* relationship existing between  $r$  and  $h$  drawn with red color in Figure 5d is missing in the previous sequence. For the training process, it means missing a training sample for relationship. For the recognition part, it means this expression can never be recognized fully. Being aware of this limitation, the 1D time sequence of strokes is used to train the BLSTM and the outputted sequence of labels during recognition will be completed to generate a LG graph as much as possible.

Expressions can be regarded as the arrangements of math symbols in two-dimensional layout according to some grammars. To feed the inputs of the BLSTM, it is important to scan as well the points belonging to the strokes themselves (on-paper points) and also the points separating one stroke from the next one (in-air points). We expect that the on-paper points will be labeled with corresponding symbol labels and that the in-air points will be assigned with one of the possible edge labels. Thus, besides re-sampling points from strokes, we also re-sample points from the straight line which links two strokes. In the rest of this paper, *strokeD* and *strokeU* are used to indicate a re-sampled pen-down stroke and a re-sampled pen-up stroke for convenience.

Given each expression, we first re-sampled points both from strokes and between two successive strokes in the writing order. 1-D unlabeled sequence can be described as  $\{strokeD_1, strokeU_2, strokeD_3, strokeU_4, \dots, strokeD_K\}$  with  $K$  is the number of re-sampled strokes. Note that if  $s$  is the number of visible strokes,  $K = 2 * s - 1$ . Ground-truth of each point is required for BLSTM training process. The labels of the points from  $strokeD_i$  should be assigned with the label of the corresponding node in LG ; The label of the points from  $strokeU_i$  should be assigned with the label of the corresponding edge in LG. If no corresponding edge exists, the label is ' $\_$ '.



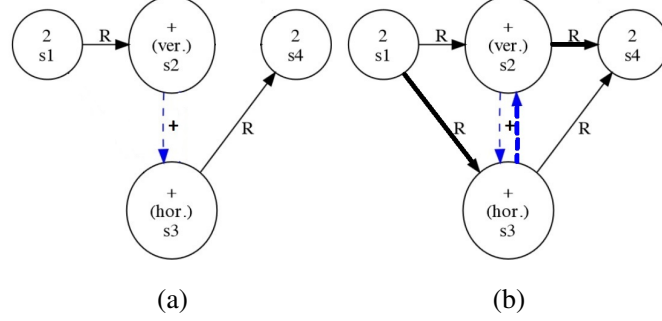


**Figure 7.** (a) The correct recognized sequence of '2 + 2' written with four strokes ;  
(b) the rebuilt LG of '2 + 2', added edges are depicted as bold.

## 5. Features

A stroke is a sequence of points sampled from the trajectory of a writing tool between a pen-down and a pen-up at a fixed interval of time. Then an additional re-sampling is performed with a fixed spatial step to get rid of the writing speed. The number of re-sampling points depends on the size of expression. For expression which has 3 or more strokes, we re-sample with  $10 \times (\text{length}/\text{avrheight})$  points ; otherwise, we re-sample with  $10 \times (\text{length}/\text{avrheight})$ . Here, *length* refers to the length of all the strokes (including distance between successive strokes) and *avrheight* (*avrheight*) refers to the average height (diagonal) of the bounding boxes of all the strokes in an expression. Then, the whole expression is re-scaled, preserving the aspect ratio, into the normalized rectangle :  $[-w/2, w/2] \times [-h/2, h/2]$  to be robust with regard to size variation, where  $w = \text{width}/\text{avrwidth}$  and  $h = \text{height}/\text{avrheight}$ , while *width* (*height*) refers to the width (height) of the bounding box of the entire expression.

Subsequently, we compute five features per point, which are quite close to the state of art (Álvaro *et al.*, 2013 ; Awal *et al.*, 2014). For every point  $p(x, y)$  we obtained 5 features in the following format :

$$[\sin\theta, \cos\theta, \sin\phi, \cos\phi, \text{PenUD}]$$

with :

- $\sin\theta, \cos\theta$  are the sine and cosine directors of the tangent of the stroke at point  $p(x, y)$  ;
- $\phi = \Delta\theta$ , defines the change of direction at point  $p(x, y)$  ;
- PenUD refers to the state of pen-down or pen-up.

## 6. Experiments

We use the RNNLIB library (Graves, 2013) for training a BLSTM model. For each training process, the network having the best classification error on validation data set is saved. Then, we test this network on the test data set. The Label Graph Evaluation library (LgEval) (Mouchère *et al.*, 2014) is adopted to evaluate the recognition output.

Three experiments are performed in this paper. Experiment 1 only focus on expressions which do not include 2-D spatial relations. In the second one, we begin to test our method on 2-D expressions the depth of which is limited to 1. It imposes that two sub-expressions having a spatial relationship (*Above*, *Below*, *Inside*, *Superscript*, *Subscript*) should be left-right expressions. Finally, all kinds of expressions available are included in experiment 3.

Network architecture and configuration are as following :

- The input layer size : 5
- The output layer size : the number of class
- The hidden layers : 2 layers, the forward and backward, each contains 100 single-cell LSTM memory blocks
- The weights : initialized uniformly in  $[-0.1, 0.1]$
- The momentum : 0.9

This configuration has obtained good results in both handwritten text recognition (Graves *et al.*, 2009) and handwritten math symbol classification (Álvaro *et al.*, 2013 ; Álvaro *et al.*, 2014a).

### 6.1. Experiment 1

We select the expressions with only left-right relations from CROHME 2014 training and test data. 2609 expressions are available for training, about one third of the full set ; 265 expressions for testing. In this case, there are 91 classes of symbols. Next, we split the training set into the new training set and validation set, 90% for training and 10% for validation. The output layer size in this experiment is 93 (91 symbol classes + *Right* + *NoRelation*). In left-right expressions, *NoRelation* will be used for indicating delayed strokes.

### 6.2. Experiment 2

In this part, according to the rule described before, 5820 expressions are selected for training from CROHME 2014 train set ; 674 expressions for test from CROHME

2014 test set. Also, we divide 5820 expressions into the new training set and validation set, 90% for training and 10% for validation. The output layer size in this experiment is 100 (93 symbol classes + 6 relationships + *NoRelation*).

### 6.3. Experiment 3

We use the complete data set from CROHME 2014, 8834 expressions for training and 982 expressions for test. Also, we divide 8834 expressions for training (90%) and validation (10%). The output layer size in this experiment is 108 (101 symbol classes + 6 relationships + *NoRelation*).

### 6.4. Discussion

The evaluation results on symbol level for the 3 experiments are provided in Table 1 including recall ('Rec.') and precision ('Prec.') rates for symbol segmentation ('Segments'), symbol segmentation and recognition ('Seg+Class'), spatial relationship classification ('Tree Rels.'). A correct spatial relationship between two symbols requires that both symbols are correctly segmented and with the right relationship label.

As can be seen, the results in 'Segments' and 'Seg+Class' of experiment 1 are a bit lower compared to experiment 2 and 3. It is a reasonable phenomenon given that the training data set in experiment 1 is not large enough. The results in 'Segments' and 'Seg+Class' do not present a big difference in experiment 2 and 3. The results for 'Tree Rels.' are declining in these 3 experiments. It is understandable as the number of missed relationships grows with the complexity of expressions with the limitation of our method. The results of experiment 3 are comparable to the results of CROHME 2014 because the same training and testing data sets are used. The second part of Table 1 gives the symbol level evaluation results of the top 4 participants in CROHME 2014 sorting by the recall rate for correct symbol segmentation. The best 'Rec.' of 'Segments' and 'Seg+Class' reported by CROHME 2014 are 98.42% and 93.91% respectively. Ours are 92.14% and 82.82%, both ranked 3 out of 8 systems (7 participants in CROHME 2014). Our solution presents competitive results on symbol recognition task and segmentation task even though the symbols with delayed strokes were missed. However, our proposal, at that stage, shows poor performances compared to the 7 participants on relationship detection and recognition task which is not surprising because the existing of limitation that some relationships were missed.

Table 2 shows the recognition rates at the global expression level with no error, and with at most one to three errors in the labels of LG. This metric is very strict. For example one label error can happen only on one stroke symbol or in the relationship between two one-stroke symbols; a labeling error on a 2-strokes symbol leads to 4 errors (2 nodes labels and 2 edges labels). The recognition rate with no error on CROHME 2014 test set is 12.22%. The best one and worst one reported by CROHME



**Table 1.** The symbol level evaluation results on CROHME 2014 test set, including the experiment results in this work and CROHME 2014 participant results (Top 4 by recall of Segments).

| exp.     | Segments (%)                            |              | Seg + Class (%) |              | Tree Rels. (%) |              |
|----------|-----------------------------------------|--------------|-----------------|--------------|----------------|--------------|
|          | Rec.                                    | Prec.        | Rec.            | Prec.        | Rec.           | Prec.        |
| 1        | 90.44                                   | 80.81        | 78.69           | 70.31        | 80.58          | 72.26        |
| 2        | 92.71                                   | 85.86        | 82.47           | 76.38        | 64             | 70.51        |
| <b>3</b> | <b>92.14</b>                            | <b>84.68</b> | <b>82.82</b>    | <b>76.12</b> | <b>59.52</b>   | <b>67.53</b> |
| system   | CROHME 2014 participant results (Top 4) |              |                 |              |                |              |
| III      | 98.42                                   | 98.13        | 93.91           | 93.63        | 94.26          | 94.01        |
| I        | 93.31                                   | 90.72        | 86.59           | 84.18        | 84.23          | 81.96        |
| VII      | 89.43                                   | 86.13        | 76.53           | 73.71        | 71.77          | 71.65        |
| V        | 88.23                                   | 84.20        | 78.45           | 74.87        | 61.38          | 72.70        |

2014 are 62.68% and 15.01%. With regard to the recognition rate with  $\leq 3$  errors, 4 participants are between 27% and 37% and our result is 28.51%.

**Table 2.** The expression level evaluation results on CROHME 2014 test set, including the experiment results in this work and CROHME 2014 participant results (Top 2)

| exp.   | correct (%)                             | $\leq 1$ error | $\leq 2$ errors | $\leq 3$ errors |
|--------|-----------------------------------------|----------------|-----------------|-----------------|
| 1      | 30.57                                   | 47.92          | 52.45           | 53.21           |
| 2      | 16.32                                   | 25.22          | 32.64           | 37.39           |
| 3      | 12.22                                   | 19.56          | 24.95           | 28.51           |
| system | CROHME 2014 participant results (Top 2) |                |                 |                 |
| III    | 62.68                                   | 72.31          | 75.15           | 76.88           |
| I      | 37.22                                   | 44.22          | 47.26           | 50.20           |

We present a correctly recognized sample and an incorrectly recognized sample in Figure 8 and Figure 9 respectively.

## 7. Conclusion

The capability of BLSTM networks to process graphical two-dimensional languages such as handwritten mathematical expressions is explored in this study. We represent the mathematical expression with a 1-D sequence derived from a path following the stroke writing order and using the labels extracted from the LG, so that classical supervised sequence labeling with recurrent neural networks can be applied. Our solution presents competitive results with CROHME 2014 on symbol recognition task and segmentation task. Proposing a global solution to perform segmentation, recognition and interpretation, with no dedicated stages, is a major advantage of the proposed solution. To some extent, at the present time, it fails on relationship recog-

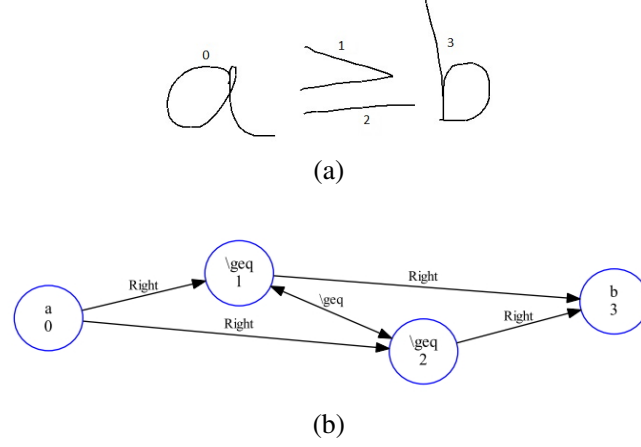


**Figure 8.** (a)  $a \geq b$  written with four strokes ; (b) the rebuilt LG of  $a \geq b$  according to the recognition result, all labels read correct.

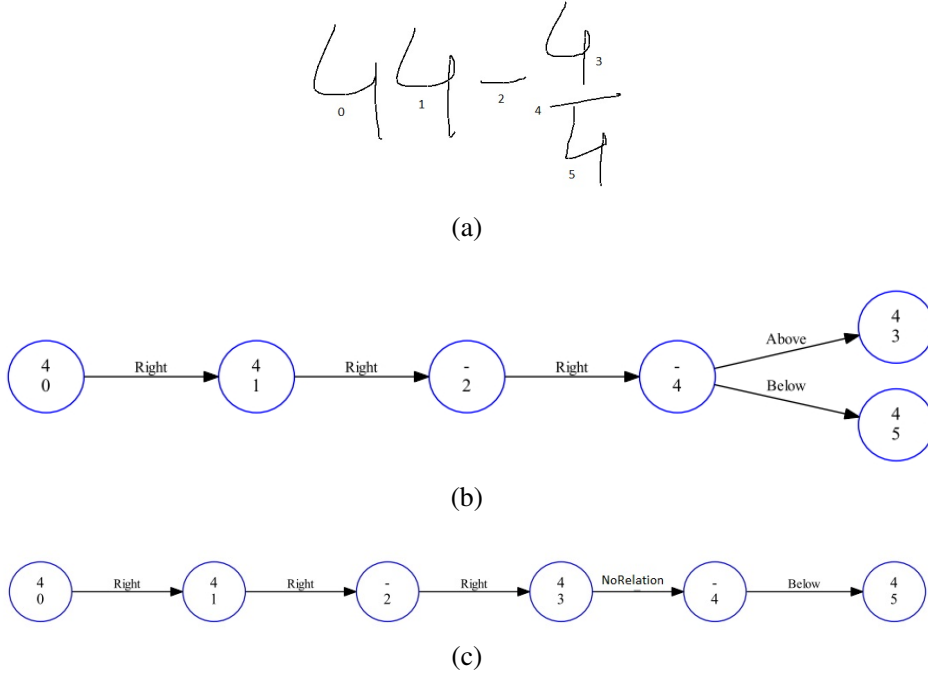


**Figure 9.** (a)  $44 - \frac{4}{4}$  written with six strokes ; (b) the ground-truth LG ; (c) the rebuilt LG according to the recognition result. Three edge errors occurred : the Right relation between strokes 2 and 4 was missed and the relation between strokes 3 and 4 was wrongly recognized, also, the edge between strokes 2 and 3 should be labeled as NoRelation.

dition task. This is primarily due to the proposed expression description. Currently, a single path following the time sequence of strokes in LG is used to represent the expression. Actually, some important relationships are discarded. In future, a better description solution will be proposed such that less relationships will be missed. Several paths can be merged to build a more robust decision which takes into account most of possible stroke configurations in time and space.

## Remerciements

We would like to thank the China Scholarship Council for the financial support for Ting ZHANG PhD studentship at Nantes university.

## 8. Bibliographie

- Álvaro F., Sánchez J.-A., Benedí J.-M., « Classification of on-line mathematical symbols with hybrid features and recurrent neural networks », *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*, IEEE, p. 1012-1016, 2013.
- Álvaro F., Sánchez J.-A., Benedí J.-M., « Offline Features for Classifying Handwritten Math Symbols with Recurrent Neural Networks », *Pattern Recognition (ICPR), 2014 22nd International Conference on*, IEEE, p. 2944-2949, 2014a.
- Álvaro F., Sánchez J.-A., Benedí J.-M., « Recognition of on-line handwritten mathematical expressions using 2d stochastic context-free grammars and hidden markov models », *Pattern Recognition Letters*, vol. 35, p. 58-67, 2014b.
- Álvaro F., Sánchez J.-A., Benedí J.-M., « An integrated grammar-based approach for mathematical expression recognition », *Pattern Recognition*, vol. 51, p. 135-147, 2016.
- Awal A.-M., Mouchère H., Viard-Gaudin C., « A global learning approach for an online handwritten mathematical expression recognition system », *Pattern Recognition Letters*, vol. 35, p. 68-77, 2014.
- Frinken V., Fischer A., Baumgartner M., Bunke H., « Keyword spotting for self-training of BLSTM NN based handwriting recognition systems », *Pattern Recognition*, vol. 47, n° 3, p. 1073-1082, 2014.
- Graves A., « Rnnlib : A recurrent neural network library for sequence learning problems », 2013.
- Graves A. *et al.*, *Supervised sequence labelling with recurrent neural networks*, vol. 385, Springer, 2012.
- Graves A., Fernández S., Gomez F., Schmidhuber J., « Connectionist temporal classification : labelling unsegmented sequence data with recurrent neural networks », *Proceedings of the 23rd international conference on Machine learning*, ACM, p. 369-376, 2006.
- Graves A., Liwicki M., Fernández S., Bertolami R., Bunke H., Schmidhuber J., « A novel connectionist system for unconstrained handwriting recognition », *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, n° 5, p. 855-868, 2009.

- Graves A., Schmidhuber J., « Framewise phoneme classification with bidirectional lstm networks », *Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on*, vol. 4, IEEE, p. 2047-2052, 2005.
- Hochreiter S., Schmidhuber J., « Long short-term memory », *Neural computation*, vol. 9, n° 8, p. 1735-1780, 1997.
- Mouchère H., Viard-Gaudin C., Zanibbi R., Garain U., « Icfhr 2014 competition on recognition of on-line handwritten mathematical expressions (crohme 2014) », *Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on*, IEEE, p. 791-796, 2014.
- Ray A., Rajeswar S., Chaudhury S., « Text recognition using deep BLSTM networks », *Advances in Pattern Recognition (ICAPR), 2015 Eighth International Conference on*, IEEE, p. 1-6, 2015.
- Schuster M., Paliwal K. K., « Bidirectional recurrent neural networks », *Signal Processing, IEEE Transactions on*, vol. 45, n° 11, p. 2673-2681, 1997.
- Zanibbi R., Blostein D., « Recognition and retrieval of mathematical expressions », *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 15, n° 4, p. 331-357, 2012.
- Zanibbi R., Mouchère H., Viard-Gaudin C., « Evaluating structural pattern recognition for handwritten math via primitive label graphs », *IS&T/SPIE Electronic Imaging*, International Society for Optics and Photonics, p. 865817-865817, 2013.



## CORIA – CIFED 2016

### Expansion de requêtes CORIA

Président : *Philippe Mulhem*

Using BLSTM for Interpretation of 2D Languages – Case of  
Handwritten Mathematical Expressions

---

# Query Expansion by Local Context Analysis

Liana Ermakova\* — Josiane Mothe\*\*

\* *Institut de Recherche en Informatique de Toulouse, Perm State National Research University*

\*\* *Institut de Recherche en Informatique de Toulouse*

---

**RÉSUMÉ.** *La tâche de notre recherche est de fournir le contexte de recherche à un moteur de recherche, i.e. étendre une requête. Nous proposons une méthode d'expansion automatique de requêtes basée sur la proximité des termes. Elle estime l'importance des termes candidats selon la proximité des termes de la requête dans les premiers documents retrouvés. L'évaluation sur les collections TREC indique que la performance de la recherche est améliorée significativement.*

**ABSTRACT.** *Query expansion (QE) aims at improving information retrieval (IR) effectiveness by enhancing the query formulation. Because users' queries are generally short and because of the language ambiguity, some information needs are difficult to answer. Query reformulation and QE methods have been developed to face this issue. Relevance feedback (RF) is one of the most popular QE techniques. In its manual version, the system uses the information on the relevance - manually judged - of retrieved documents in order to expand the initial query. Rather than using users' judgment on the document relevance, blind RF considers the first retrieved documents as relevant. Generally speaking, RF methods consider the terms that cooccur with query terms within positive feedback documents as candidates for the expansion. Rather than considering feedback documents in their all, it is possible to analyze local information. This paper presents a new method that uses local context from feedback documents for QE. The method uses POS information as well as the remoteness from query terms within feedback documents. We show that the method significantly improves precision on TREC collections.*

**MOTS-CLÉS :** *Recherche d'information, expansion automatique de requêtes, retour de pertinence, contexte local.*

**KEYWORDS:** *Information Retrieval, Query Expansion, Relevance Feedback, Local Context.*

---

## 1. Introduction

IR aims at retrieving the relevant documents according to a user's need. Concretely, a search engine computes a similarity between the user's query and the indexed documents ; the documents that contain the query terms are retrieved and ordered according to their decreasing similarity with the query. Because real queries are short and because natural language is ambiguous such matches can be wrong or incomplete. Matching is first difficult because the terms used by the authors of documents and by the users of search engines to represent a concept may be different. It is also difficult because users express their needs using just a few words, making the query difficult to "understand" by the system. The average query length remains between 2.4 and 2.7 words (Gabrilovich *et al.*, 2009). Various approaches have been developed to face these challenges. Semantic indexing and search aim at tackling the problem of term ambiguity. Some solutions rely on knowledge resources such as ontologies to use concepts rather than terms or stems, both during indexing and matching (Ermakova, 2015). Term ambiguity has also been treated with positive results as a classification or clustering problem, in which documents that share the same sense with the query terms are retrieved whereas documents that use the query terms but in a different meaning are filtered out (Schütze, 1998). On the other hand QE has driven many works in IR (see Carpineto's survey (Carpineto et Romano, 2012)). QE aims at adding new terms to the initial query based on some knowledge, either extracted from the term collection distribution, from the user or user's profile, or from feedback information. The initial query can be expanded using term cooccurrences in the documents (Amati, 2003 ; Xu et Croft, 1996) or based on Word-Net definition (Voorhees, 1994). The former type of approaches has the advantage of taking into account the document collection and thus the capability of the collection to contain the relevant information whereas the latter is collection independent. The use of relevance information was suggested by Rocchio (Rocchio, 1971) who defines the RF principle. Users are supposed to judge in some ways some of the retrieved documents and this feedback information is used in turn either to reweight query terms or to expand the query with the most important terms from relevant documents. To avoid users' judgment that can be difficult to collect and to make the process fully automatic, Buckley et al. (Buckley, 1995) suggested to consider the first initially retrieved documents as relevant. Many studies have shown that this method is efficient in general even if it can lower the performance for some queries (Amati *et al.*, 2004 ; Cronen-Townsend et Croft, 2002).

Selecting the most appropriate terms from the relevant -or considered as such- documents is a challenge. While weighting the term candidates considering their frequency or their weight calculated during the indexing phase is an intuitive and widely used approach, we suggest that a deeper analysis of document content can be useful. Searching for the best-fit terms in our approach is based on the local context analysis for ranking of terms and sentences. We use the term *local context analysis* for the analysis of neighborhood of a sentence or a term. Our first hypothesis is that terms that occur closely to query terms within the documents should be good candidates for QE ; the closer the better candidate. The second hypothesis is that natural language



considerations should help to decide the best candidate terms, that is to say that some types of terms should be better candidates (e.g. noun being better than adverbs). To study these hypotheses, we propose a method that considers a term window surrounding query terms from feedback documents. In addition, our method considers Part Of Speech (POS) information to weight differently the QE term candidates. The remainder of the paper is organized as follows. Section 2 presents related works. Section 3 details the QE method we promote. Section 4 presents the experimental framework as well as the collections and performance measures we used. Section 5 provides the results. Finally, section 6 concludes the paper and draws up some future works.

## 2. Literature Review

Automatic methods for QE were firstly proposed by Maron and Kuhns in 1960. QE based on RF makes the hypothesis that relevant documents are key components to decide which terms are important to formulate an enhanced query regardless to an information need. Using the vector space model, Rocchio defined a way to re-weight query terms and thus to add new terms to the initial query - terms that were initially associated with a null value. The term weights are re-computed so that the terms that occur in relevant documents contribute positively to the new query whereas the weight of the terms that occur in non-relevant documents are lowered (Rocchio, 1971). A balance between the initial query and feedback information is involved in the weighting. While improving the effectiveness of search, the method however implies that document relevance is collected. Buckley et al. (Buckley, 1995) went a step further by assuming the top-retrieved documents are relevant. The so-called blind or pseudo RF is now commonly used in IR evaluation campaigns. Some current studies focus on the when QE is useful. Indeed it has been shown that if RF successfully improves the system performance in average (Voorhees et Harman, 1998), in some cases, QE worsens the quality of the retrieval (Amati *et al.*, 2004). Some approaches considers selective QE : the system decides whether or not QE should applied, based on some query features (Cronen-Townsend et Croft, 2002). The query features are either pre-retrieval or post-retrieval query features that are used to cluster queries. A training phase builds the model from queries for which the best decision is know ; then the model is applied to any new query. Another range of works focuses on selecting the best feedback information. Cao et al. propose a term classification method to predict the usefulness of expansion term candidates (Cao *et al.*, 2008). Xu et al. (Xu *et al.*, 2009) suggested that top documents should not be considered in a blind way but rather the system should distinguish relevant from non-relevant documents. Their idea is that non relevant documents should cluster as relevant documents do. In addition, they consider that query terms should occur in the relevant document cluster and that some documents from the non-relevant cluster do not contain any of the query terms. One can consider term cooccurrence by applying latent semantic analysis hypothesizing that related words cooccur in similar context (Landauer *et al.*, 1998). Term co-occurrence may be discovered by a cluster algorithm, e.g. the Naive-Bayes maximizing the classification maximum likelihood criterion, where each word is pre-

sented as a vector with the components corresponding to the number of occurrences of the word in each document (Amini *et al.*, 2007). Singh and Sharan combined co-occurrence and semantic similarity of terms (Singh et Sharan, 2015). Xu and Croft went a step further (Xu et Croft, 1996). They use a feature selection based on co-occurrence of terms, considering that the best terms are the ones that co-occur with as many query terms as possible within the top-ranked documents or document passages. In addition, they consider nouns and noun phrases as the expansion terms. Xu and Croft’s co-occurrence measure is not a probability in the strict sense, while mutual information shows the joint probability of terms to co-occur within a text. Wan *et al.* suggested to combine ontology-based methods with the proximity heuristics (Wan *et al.*, 2012). Miao *et al.* proposed an extension of the Rocchio’s approach by introducing a concept of proximity-based term frequency that focuses on the proximity of terms rather than positional information unlike the positional relevance model (Miao *et al.*, 2012). They provide 3 approaches to estimate the proximity-based term frequency, namely (1) moving window ; (2) kernel-based and (3) Hyperspace Analogue to Language (HAL) methods. In contrast to (Wan *et al.*, 2012 ; Miao *et al.*, 2012), we estimate the distance in term of sentences and we evaluate the sentences that are the sources of the candidate terms.

The method we propose also considers feedback information for QE. We consider the co-occurrence of query terms with QE term candidates from pseudo-relevant documents. Moreover, we make use of the surrounding terms of query terms. Our hypothesis is that the closer a term to a query term, the better as a QE term candidate. POS information is used in order to weight differently QE candidates.

### 3. Method Description

In this paper a *token* is viewed as a word occurrence within a document. Moreover, we will use *token* and *word* as synonyms. A term is a normalized representation of tokens in a dictionary (Manning *et al.*, 2008) (i.e. stems or lemmas). The key idea of the proposed method is to search the most appropriate terms for QE in the top ranked documents. Searching for the best-fit terms is based on ranking of terms and sentences. Both ranking procedures include local context analysis, i.e. analysis of neighboring sentences. Our approach is underlain by the following hypotheses :

- 1) Not always an entire document is relevant to a query, but it can contain one or several relevant passages. Term candidates should be selected from these passages.
- 2) Terms for QE come from appropriate sentences (in general, this hypothesis is similar to those of RF). The measure of sentence appropriateness is called sentence score and referred to  $score(S)$  in the rest of the paper.
- 3) Good terms should have appropriate POS and high *IDF*. Not all POS are suitable for QE (e.g. functional words). Moreover, the most frequent terms are nouns. However, in some cases adjectives, verbs and numbers are indispensable. A good term should well distinguish documents from each other. POS weight and *IDF* may be

considered as a query-independent term score.

4) The terms lying in the neighborhood of query terms are closer related to them than the remote ones.

Thus, all candidate terms are ranked according to the following metric :

$$w_{total}(t) = f(score(S), w_{pos}(t), IDF(t), importance(t, Q)) \quad [1]$$

where  $score(S)$  is score of the sentence  $S$  containing  $t$ ,  $w_{pos}(t)$  is the weight of the POS of  $t$ ,  $IDF(t)$  is the inverse document frequency of the candidate term,  $importance(t, Q)$  is a function of (1) the distance to the query  $Q$  terms, (2) their weights, and (3) the likelihood of the candidate term to cooccur not by chance with the query terms in the top ranked documents.  $importance(t, Q)$  allows to find terms occurring in the neighborhood of important query terms. These elements are described in the following sections.

### 3.1. Sentence Scoring

Sentence scoring method is an elaboration of RF. We strengthen the criteria of provenance of good terms for QE used in RF. Sentence scoring method presented in this paper is the adaptation of the method initially developed for query-biased multi-document summarization (Ermakova, 2015). We assume that appropriate terms come from appropriate sentences. Right sentences should be of high quality and moreover they should match the query. Hence, sentence score is estimated as the function of sentence weight  $ws(S)$  and sentence quality measure  $SntQual(S)$  which is used to avoid trash passages from real web collections :

$$score(S) = \sigma(ws(S), SntQual(S)) \quad [2]$$

We define it as the function of the lexical diversity  $LexDiv(S)$ , meaningful word ratio  $Meaning(S)$  and punctuation score  $PunctScore(S)$  :

$$SntQual(S) = \phi(LexDiv(S), Meaning(S), PunctScore(S)) \quad [3]$$

**Lexical diversity** allows avoiding sentences that do not contain terms except those from a query. Lexical diversity in our approach is defined as the number of different lemmas used within a sentence divided by the total number of tokens in this sentence. **Meaningful word ratio** is also aimed to penalized sentences that either have no sense at all or are not comprehensible without large context. Meaningful word ratio is the number of non-stop words within a sentence over the total number of tokens in this sentence. Besides unreadable passages, many symbols usually used as punctuation marks can be found in emoticons. Emoticons represents humans' attitude towards something. However, they are not relevant for informative, navigational nor transactional queries. Hence,  $PunctScore(S)$  penalizes sentences containing many punctuation marks. **Punctuation score** is estimated by the formula :

$$PunctScore(S) = 1 - \frac{PunctuationMarkCount(S)}{TokenCount(S)} \quad [4]$$

where  $PunctuationMarkCount(S)$  is a total number of punctuation marks in the sentence, and  $TokenCount(S)$  – is a total number of tokens in  $S$ . Thus,  $PunctScore(S)$  shows the ratio of tokens which are not punctuation marks.

Thus, we believe that a good sentence should have high ratio of different meaningful words and reasonable ratio of punctuation. Sentence quality is query-independent, while sentence weight  $ws(S)$  shows how well a sentence matches a query.

We assume that relevant sentences come from relevant documents. However, in real world search engines we do not know which documents are actually relevant. Therefore in our method sentence weight depends on pseudo-relevance  $DocRel(d)$  of the corresponding document  $d$  assigned by a search engine (i.e. document rank, score or their combination), and computed smoothed sentence relevance  $R(S)$  :

$$ws(S) = \omega(DocRel(d), R(S)) \quad [5]$$

**Smoothed sentence relevance** estimation  $R(S)$  in (5) is based on the first-stage local context analysis. In contrast to (Yang *et al.*, 2011), we believe that a context does not provide redundant information, but allows to precise and extend sentence meaning. Neighboring sentences influence the sentence of interest, but this influence decreases as the remoteness of the context increases. We choose the simplest dependence model, namely the linear function. In this case  $R(S)$  is calculated by the formulas :

$$R(S) = \sum_{i=-k}^k w_i \times r_i \quad [6]$$

$$w_i = \begin{cases} \frac{1-w_t}{k+1} \times \frac{k-|i|}{k} & 0 < |i| \leq k \\ w(S), & i = 0 \\ 0, & |i| > k \end{cases} \quad [7]$$

$$\sum_{i=-k}^k w_i = 1 \quad [8]$$

where  $w(S)$  is the weight of the sentence  $S$ ,  $w_i$  and  $r_i$  are respectively the weights and the prior scores of the sentences from the context of  $S$  of  $k$  length. In this formula weights of context linearly decrease with the growth of the distance from the sentence  $S$ . If the sentence number in left or right context is less than  $k$ , their weights are added to the target sentence weight  $w(S)$ . This allows keeping the sum equal to one. That is important since otherwise a sentence with a small number of neighbors (e.g. the first or last sentences) would be penalized (even if first sentences of a document are often considered to be very informative). Prior scores of sentence  $r_i$  correspond to their prior similarity to the query  $sim_{total}(S, Q)$  which is a function of the cosine similarity measure  $sim_{uni}$  between the terms of the sentence and the query and the similarity of named entities  $sim_{NE}$  :

$$r_i = sim_{total}(S, Q) = \zeta(sim_{uni}, sim_{NE}) \quad [9]$$

### 3.2. Term Weighting

The next step of our method is to compute the importance of all terms in all sentences from RF :

$$importance(t, Q) = \theta(wd(t, Q), cooccurrence(t, Q)) \quad [10]$$

$wd(t, Q)$  is a function of the distance from the candidate terms to the query  $Q$  and their weights, and  $cooccurrence(t, Q)$  shows the likelihood of the candidate term to occur not by chance with the query terms in the top documents ranked according to the initial query. The concrete functions are given in the evaluation section.

The term score is combined with the corresponding sentence score. Thus, we used a two-step local context analysis : for sentence scoring and for estimation of term importance. In previous works local context was viewed as a single document and it was opposed to the entire collection analysis (global context) (Carpineto et Romano, 2012 ; Xu et Croft, 2000). In this paper we consider local context in a stricter way, precisely we look not only to the whole document statistics, but also for terms surrounding the query terms.

## 4. Experimental Framework

In this section the experimental framework is described. Firstly, we present the details of the implemented system. Then we provide the data sets we used and the evaluation metrics. The last subsection describes the systems used for comparison.

### 4.1. Details of the Implemented System

Our approach requires RF. In order to obtain preliminary ranking we used the Terrier platform<sup>1</sup>, an open-source search engine developed by the School of Computing Science, University of Glasgow. This platform considers documents as bags of words. It implements various weighting and retrieval models. We applied a default retrieval model in Terrier *InL2*. *InL2* is a DFR (Divergence from Randomness) model based on  $TF - IDF$  measure with  $L2$  term frequency normalization (Amati, 2003). The DFR models are based on the assumption that informative words are relatively more frequent in relevant documents than in others (Amati, 2003). In the DFR models the weight  $weight(t, d)$  of the term  $t$  in the document  $d$  is estimated as follows :

$$weight(t, d) = \frac{1}{tf + 1} \times Inf_1(tf) \quad [11]$$

$$Inf_1(tf) = tf \times \log_2 \frac{N + 1}{n + 0.5} \quad [12]$$

---

1. terrier.org

where  $tf$  is the initial frequency of the term  $t$  in the document  $d$ ,  $N$  is the total number of documents in the collection, and  $n$  is the number of documents containing the term  $t$ .  $L2$  normalization of the term frequency  $tf_n$  is computed as :

$$tf_n = tf \times \log_2 1 + c \frac{avg_l}{l}, \quad c > 0 \quad [13]$$

where  $l$  is the length of the document  $d$ ,  $avg_l$  is the average document length, and  $c$  is the normalization parameter. Thus,  $weight(t, d)$  determined by  $InL2$  is :

$$weight(t, d) = \frac{tf_n}{tf_n + 1} \times \log_2 \frac{N + 1}{n + 0.5} \quad [14]$$

Stemming was performed by Porter's algorithm. The 10 top documents retrieved for each query were processed by Stanford CoreNLP which integrates POS tagger based on the Penn Treebank tag set and named entity recognizer<sup>2</sup>. The parsed documents were treated as described in the section 3.

For the parameter combination one can use various functions : sum, weighted sum, product, exponential function or more complex non-linear functions. Exponential function may be reduced to product. In our experiments we mostly used product function to combine parameters. It was caused by several reasons. Firstly, the product of parameters is biased to the smallest one. It allows penalizing candidates that are not suitable according to one of the score elements, e.g. the sums of  $0.9 + 0.1 = 1$  and  $0.5 + 0.5 = 1$  are equal, but the products  $0.9 \times 0.1 = 0.09$  and  $0.5 \times 0.5 = 0.25$  differ much. The same refers to weighted sum. The only difference is the impact of each parameter in the result. Moreover, multiplication does not require normalization. In our future work we will try more complex functions to combine parameters. Hence, the total term score was estimated as the product of the score of the sentence  $S$  containing the term  $t$ , its POS weight,  $IDF$  and the likelihood of  $t$  to cooccur not by chance with the query terms :

$$w_{total}(t) = score(S) \times w_{POS}(t) \times IDF(t) \times importance(t, Q) \quad [15]$$

The weights  $w_{POS}(t)$  of different POS were set as parameters.

Sentence score was computed as the product of the sentence weight  $ws(S)$  and its quality  $SntQual(S)$  :

$$score(S) = ws(S) \times SntQual(S) \quad [16]$$

Sentence weight  $ws(S)$  was calculated by the formula :

$$ws(S) = DocRel(d) \times R(S) \quad [17]$$

Sentence quality measure  $SntQual(S)$  was estimated as the product of the lexical diversity  $LexDiv(S)$ , meaningful word ratio  $Meaning(S)$  and punctuation score  $PunctScore(S)$  :

$$SntQual(S) = LexDiv(S) \times Meaning(S) \times PunctScore(S) \quad [18]$$

2. <http://stanfordnlp.github.io/CoreNLP/>

The similarity between a sentence and a query was computed as follows :

$$sim_{total}(S, Q) = sim_{uni} \times sim_{NE} \quad [19]$$

$$sim_{NE} = \frac{NE_{common} + NE_{weight}}{NE_{query} + 1} \quad [20]$$

where  $NE_{weight}$  is positive floating point parameter,  $NE_{common}$  is the number of NE appearing in both query and sentence,  $NE_{query}$  is the number of NE appearing in the query.  $NE_{weight}$  allows not to reject sentence without NE which can be still relevant. We add 1 to the denominator to avoid division by zero.

$$importance(t, Q) = wd(t, Q) \times cooccurrence(t, Q) \quad [21]$$

$$wd(t, Q) = \max_{Q_i} (IDF(Q_i) \times d(t, Q_i)) \quad [22]$$

$$d(t, Q_i) = \frac{1}{\max(1, dist(t, Q_i) - 2)} \quad [23]$$

where  $dist(t, Q_i)$  is the distance from the term to the nearest term appearing in the query. We believe that terms cooccurring in the window of the length 4 are strongly interconnected, thus we consider them to have the same weight by applying the formula  $\max(1, dist(t, Q_i) - 2)$ . Here we use the same hypothesis as for sentence scoring, namely the interconnection between text parts decreases as the distance between them grows. The likelihood of the candidate term to occur not by chance with the query terms in the top documents ranked according to the initial query can be calculated by different models (see the detailed description of Xu and Croft's approach and DFR models in subsection 4.4). In our approach it was estimated similarly to Bose-Einstein 2 model (Amati, 2003). We consider only terms occurring at least in 2 top ranked documents. In contrast to (Amati, 2003), we apply grammatical filters to the candidate terms, i.e. we consider only some POS (e.g. only nouns or all meaningful words), while others are ignored (e.g. numbers, pronouns, adverbs etc.). Another modification is that we filter words before scoring.

## 4.2. Data Sets

The evaluation was performed on two kinds of datasets : TREC Ad Hoc Track data sets and WT10G<sup>3</sup>. Documents are tagged by SGML. The collected documents are not normalized and may contain spelling or other errors. TREC Ad Hoc Track data sets are "pure" collections since the documents have almost the same format and there is no spam. In contrast, WT10G is a snapshot of the web with real documents in HTML format, some of which are spam.

3. <http://trec.nist.gov/>

We used TREC Ad Hoc Track data sets (Text Retrieval Conference) for three years (TREC) : 1997 (6), 1998 (7) and 1999 (8). TREC 6-8 are driven on the data on Disks 4 and 5 and contain 150 topics in total. There are 4 sources of documents : the news articles from (1) The Financial Times, 1991-1994 (FT) - 564MB, 210 158 documents ; (2) Federal Register, 1994 (FR94) - 395MB, 55 630 documents ; (3) Foreign Broadcast Information Service (FBIS) - 470MB, 130 471 documents ; and (4) The LA Times - 475MB, 131 896 documents. Each of TREC 6-8 has 50 topics. A topic represents an information need and contains 4 fields : (1) topic number, (2) title (very short description of a topic – about three words), (3) description (a “normal” sentence description of a topic), and (4) narrative (description of the information that should be presented at relevant documents). The pools of relevant documents (q-rels) were merged from the top 100 documents per topic retrieved in each submitted run and assessed by humans. In total for 150 queries 14 013 documents were considered to be relevant.

WT10G was used at TREC Web track 2000-2001. It is 10GB subset of the web snapshot of 1997 from Internet Archive. WT10G contains 1 692 096 documents from 11 680 servers (minimum 5 documents per server). There were 50 topics in 2000 and 2001 (total 100 topics). In total 5 953 were judged as relevant.

#### 4.3. Evaluation Measures

TREC q-rels may be used by trec\_eval. The trec\_eval software enables to evaluate ranked retrieval results and implements the following measures :

- Mean Average Precision over all queries ;
- R-Precision ;
- Normalized discounted cumulative gain ;
- Precision at 5, 10, 15, 20, 30, 50, 100, 200, 500 and 1000 ;
- 11-point interpolated average precision (at 0%, 10%, ..., 100% of recall) which allows to draw precision-recall curve.

Precision (P) is the fraction of retrieved documents that are relevant (Manning *et al.*, 2008) :

$$P = \frac{\#RelevantRetrievedItems}{\#RetrievedItems} \quad [24]$$

Precision at k ( $P@k$ ) the fraction of the top k retrieved documents that are relevant. Interpolated average precision is the ration of relevant retrieved documents over the number of documents that gives a certain percentage of recall. Recall (R) shows the fraction of relevant retrieved documents over all relevant documents :

$$R = \frac{\#RelevantRetrievedItems}{\#RelevantItems} \quad [25]$$

R-Precision is the proportion of relevant retrieved documents over the size of the set of relevant documents.



Mean average precision is calculated as follows :

$$MAP = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} Precision(R_{jk}) \quad [26]$$

where  $|Q|$  is the number of queries,  $m_j$  is the number of relevant documents for the  $j$ -th query, and  $R_{jk}$  is the set of ranked retrieval results from the top result until the document  $d_k$  is achieved.

Discounted cumulative gain  $DCG$  is a measure of IR effectiveness that penalizes highly relevant documents appearing lower in a search result (Manning *et al.*, 2008). The graded relevance value is discounted logarithmically proportional to the position of the result :

$$DCG_k(Q_j) = \sum_{i=1}^k \frac{2^{rel_i^{(j)}} - 1}{\log_2(i + 1)} \quad [27]$$

Normalized discounted cumulative gain  $NDCG$  is normalized over Ideal DCG  $IDCG$ , i.e. the maximum possible DCG till the position  $k$  :

$$NDCG_k(Q_j) = \frac{DCG_k(Q_j)}{IDCG_k(Q_j)} \quad [28]$$

$NDCG$  can be averaged over all queries and all positions :

$$NDCG(k) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} NDCG_k(Q_j) \quad [29]$$

$$NDCG = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} NDCG_k(Q_j) \quad [30]$$

#### 4.4. Systems Used for Comparison

For evaluation purpose we used several RF methods, namely Xu and Croft's cooccurrence model and DFR models implemented in Terrier. During QE the best-scored terms from the top-ranked documents are extracted. Terms are ranked using one of the DFR weighting model.

In the **Kullback-Leibler** model QE is performed by ordering the candidate terms by their information content given the query  $Q$  (Amati, 2003) :

$$Inf(t | Q) = Inf_{D_Q}(t) = -\log P(t | Q) \quad [31]$$

where  $t$  is the candidate term. If  $P(t | Q)$  is viewed as binomial distribution :

$$\begin{aligned} P(t | Q) &= B(C_{D'(t)}, C_{D'}, \frac{C_D(t)}{C_D}) \\ &= \binom{C_{D'}}{C_{D'(t)}} \left( \frac{C_D(t)}{C_D} \right)^{C_{D'(t)}} \left( 1 - \frac{C_D(t)}{C_D} \right)^{C_{D'} - C_{D'(t)}} \end{aligned} \quad (32)$$

where  $D'$  is a subset of the original collection  $D$ ,  $C_{X(t)}$  is the number of time the term  $t$  occurs in  $X$ ,  $C_X$  – the total number of terms in  $X$ . In the case of the approximation via the divergence function the information content of the term  $t$  is proportional to :

$$Inf(t | Q) \sim TF_{D'}(t) \times \log \frac{TF_{D'}(t)}{TF_D(t)} \quad (33)$$

The information content of the term  $t$  in the **Chi-square** model is estimated as (Amati, 2003) :

$$\begin{aligned} Inf(t | Q) &\sim TF_{D'}(t) \times TF_{D'} \times \left( \log \frac{TF_{D'}(t)}{TF_D(t)} + \log \frac{1 - TF_{D'}(t)}{1 - TF_D(t)} \right) \\ &+ 0.5 \times (2\pi \times TF_{D'} \times (1 - \frac{TF_{D'}(t)}{TF_D(t)})) \end{aligned} \quad (34)$$

**Bose-Einstein 1 (Bo1)** and **2 (Bo2)** models are DFR models implemented in Terrier (Amati, 2003). By default they are parameter-free, but Rocchio's QE mechanism can be also applied.

$$Bo1 = TF_{D'}(t) \times \log \frac{1 + f1}{f1} + \log (1 + f1) \quad (35)$$

$$f1 = \frac{TF_D(t)}{|D|} \quad (36)$$

$$Bo2 = TF_{D'}(t) \times \log \frac{1 + f2}{f2} + \log (1 + f2) \quad (37)$$

$$f2 = \frac{TF_{D'}(t) \times TF_{D'}}{TF_D} \quad (38)$$

We implemented the **Xu and Croft's** method described in (Xu et Croft, 2000) both for the entire documents and documents split at blocks of 300 words. The Porter's algorithm was used for stemming. The cooccurrence  $cooccurrence(t, Q)$  measure shows the likelihood of a term to occur not by chance with the query terms in the top ranked document set.  $cooccurrence(t, Q)$  is estimated as follows :

$$cooccurrence(t, Q) = \prod_{Q_i} (\delta + co_{degree}(t, Q_i))^{IDF(Q_i)} \quad (39)$$

$$co_{degree}(t, Q_i) = \log\left(\sum_{d \in D'} TF(t, d) \times TF(Q_i, d) + 1\right) \times \frac{IDF(t)}{\log n} \quad [40]$$

The parameter  $\delta$  was set to 0.05.  $IDF$  was estimated as :

$$IDF(t) = \min\left(1.0, \frac{\log_1 0 \frac{|D|}{|D'|}}{5}\right) \quad [41]$$

where  $D'$  is the set of the top-ranked documents from the collection  $D$ .

## 5. Results

We compared our system (LC) with those described in the previous section (see tables 1 and 2), namely :

- Kullback-Leibler divergence model implemented in Terrier (KL) ;
- Chi-square divergence model from Terrier (CS) ;
- Bose-Einstein 1 model from Terrier (Bo1) ;
- Bose-Einstein 2 model from Terrier (Bo2) ;
- Baseline presented by InL2 model without any QE (InL2) ;
- Xu and Croft's approach.

All systems used InL2 model for RF, 3 documents from which 10 best scored terms were extracted. The results obtained by applying Xu and Croft's approach were much lower than the baseline. Therefore, we do not present them.

Table 1 provides information about the number of relevant retrieved documents (RRD), MAP, R-precision (R-P) and NDCG. MAP and R-precision may be viewed as the main measures since MAP has very good discrimination and stability and R-precision represents the break-even point and highly correlates with MAP (Manning *et al.*, 2008). On both data sets by all metrics our systems showed the best results, which are much higher than the baseline. Table 2 contains the values of precision at  $k$  for the systems mentioned above. In most cases users prefer to look for relevant documents on the first page of search results. The standard size of search engine result pages is 10. Therefore, precision at 5 and 10 seems to be very important. At this level our system showed the best results for both data sets as well as for P@100. Table 3 provides paired difference Student t-test results performed on average precisions. Our method had the following characteristics Mean = 0.253620, Std.Dv. = 0.217502 and Mean = 0.220422, Std.Dv. = 0.20096 at TREC 6-8 and Web data sets respectively. At the level  $p < 0.05$  the differences between the approach proposed in this paper and all other evaluated methods are significant for TREC 6-8 data. On Web data at the level  $p < 0.05$  the differences between the our approach and Bo2 is insignificant, as well as the differences between LC and KL. The insignificant difference between LC and Bo2 can be caused by the fact that the design of our system which implies modified Bo2-score computation.

**Tableau 1. General results**

|          |      | LC            | Bo2    | KL     | Bo1    | CS     | InL2   |
|----------|------|---------------|--------|--------|--------|--------|--------|
| TREC 6-8 | RRD  | <b>8242</b>   | 8184   | 8034   | 8027   | 7913   | 7225   |
|          | MAP  | <b>0.2536</b> | 0.2491 | 0.2400 | 0.2391 | 0.2312 | 0.2105 |
|          | R-P  | <b>0.291</b>  | 0.2868 | 0.2797 | 0.2818 | 0.2611 | 0.2627 |
|          | NDCG | <b>0.5221</b> | 0.5168 | 0.5093 | 0.5087 | 0.4921 | 0.4682 |
| WT10G    | RRD  | <b>3964</b>   | 3935   | 3938   | 3970   | 3724   | 3810   |
|          | MAP  | <b>0.2204</b> | 0.219  | 0.212  | 0.2094 | 0.2068 | 0.1894 |
|          | R-P  | <b>0.2485</b> | 0.241  | 0.2375 | 0.2355 | 0.242  | 0.2304 |
|          | NDCG | <b>0.4844</b> | 0.4795 | 0.4816 | 0.4816 | 0.4515 | 0.4624 |

**Tableau 2. Precision at  $k$** 

|          |        | LC     | Bo2    | KL     | Bo1    | CS     | InL2   |
|----------|--------|--------|--------|--------|--------|--------|--------|
| TREC 6-8 | P@5    | 0.496  | 0.4933 | 0.4800 | 0.4773 | 0.4547 | 0.4413 |
|          | P@10   | 0.4427 | 0.4413 | 0.4333 | 0.4327 | 0.4107 | 0.4180 |
|          | P@100  | 0.2235 | 0.2190 | 0.2133 | 0.2119 | 0.2048 | 0.1953 |
|          | P@1000 | 0.0549 | 0.0546 | 0.0536 | 0.0535 | 0.0528 | 0.0482 |
| WT10G    | P@5    | 0.3816 | 0.3796 | 0.3714 | 0.3694 | 0.3551 | 0.3286 |
|          | P@10   | 0.3378 | 0.3296 | 0.3255 | 0.3163 | 0.3102 | 0.2816 |
|          | P@100  | 0.1599 | 0.1589 | 0.1523 | 0.1510 | 0.1480 | 0.1387 |
|          | P@1000 | 0.0404 | 0.0402 | 0.0402 | 0.0405 | 0.0380 | 0.0389 |

## 6. Conclusion

In this paper we proposed the method for QE based on cooccurrence measure as well as importance estimated by analyzing local context. In contrast to previous works we treated not only entire documents, but also text passages surrounding query terms. We evaluated our method on TREC Ad Hoc 6-8 collection as well as on WT10g. Our approach obtained the best results among 6 systems, namely the baseline and DFR models for QE implemented in Terrier platform. Our system outperformed others according to MAP, NDCG and R-precision for both data sets. For TREC Ad Hoc 6-8 it also showed the best results according to the number of relevant retrieved documents.

**Tableau 3. T-test. Significant results are marked by \***

|          |          | InL2    | Bo1     | Bo2     | CS      | KL      |
|----------|----------|---------|---------|---------|---------|---------|
| TREC 6-8 | Mean     | 0.2105  | 0.2391  | 0.249   | 0.2311  | 0.24    |
|          | Std.Dv.  | 0.1961  | 0.2092  | 0.2152  | 0.2143  | 0.21    |
|          | Diff.    | 0.0431  | 0.0145  | 0.0045  | 0.0224  | 0.0135  |
|          | t        | 6.0666  | 3.9325  | 2.3173  | 3.8832  | 3.6129  |
|          | p        | 0.0*    | 0.0001* | 0.0218* | 0.0001* | 0.0004* |
|          | Cnf.-95% | 0.029   | 0.0072  | 0.0006  | 0.011   | 0.0061  |
|          | Cnf.+95% | 0.057   | 0.0217  | 0.0084  | 0.0338  | 0.021   |
| WT10G    | Mean     | 0.1893  | 0.2094  | 0.219   | 0.2067  | 0.2119  |
|          | Std.Dv.  | 0.1865  | 0.1973  | 0.2019  | 0.2014  | 0.2007  |
|          | Diff.    | 0.031   | 0.011   | 0.0013  | 0.0136  | 0.0084  |
|          | t        | 3.7126  | 2.1668  | 0.3564  | 3.2594  | 1.6283  |
|          | p        | 0.0003* | 0.0326* | 0.7222  | 0.0015* | 0.1066  |
|          | Cnf.-95% | 0.0144  | 0.0009  | -0.0063 | 0.0053  | -0.0018 |
|          | Cnf.+95% | 0.0476  | 0.021   | 0.009   | 0.0219  | 0.0187  |

For the Web data by this measure our system was right after the Bo1 model. Our system showed the best results for both data sets for P@5, P@10, and P@100. As it was showed in (Savenkov *et al.*, 2011), users trust search engine ranking and they click more frequently on top-ranked documents. Thus, P@5 and P@10 seems to be quite crucial. The differences between the our approach and other evaluated methods are significant at the level  $p < 0.05$  for TREC Ad Hoc 6-8 collection. On Web data at  $p < 0.05$  the differences with Bo2 and KL models is insignificant. The insignificant difference between LC and Bo2 could be explained by the fact that our method is partially based on Bo2.

## 7. Bibliographie

- Amati G., *Probability Models for Information Retrieval Based on Divergence from Randomness : PhD Thesis*, University of Glasgow, 2003.
- Amati G., Carpineto C., Romano G., « Query Difficulty, Robustness, and Selective Application of Query Expansion », *Advances in Information Retrieval* p. 127–137, 2004.
- Amini M. R., Tombros A., Usunier N., Lalmas M., « Learning-based summarisation of XML documents », *Inf. Retr.*, vol. 10, n<sup>o</sup> 3, p. 233–255, 2007.
- Buckley C., « Automatic Query Expansion Using SMART : TREC 3 », *In Proceedings of The third Text REtrieval Conference (TREC-3). NIST Special Publication 500-226.*, National Institute of Standards and Technology (NIST), Gaithersburg, MD, p. 69–80, 1995.
- Cao G., Nie J.-Y., Gao J., Robertson S., « Selecting good expansion terms for pseudo-relevance feedback », *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, ACM, New York, NY, USA, p. 243–250, 2008.
- Carpineto C., Romano G., « A Survey of Automatic Query Expansion in Information Retrieval », *ACM Computing Surveys*, vol. 44, n<sup>o</sup> 1, p. 1–50, January, 2012.
- Cronen-Townsend S., Croft W. B., « Quantifying query ambiguity », p. 104–109, March, 2002.
- Ermakova L., « A Method for Short Message Contextualization : Experiments at CLEF/INEX », *Experimental IR Meets Multilinguality, Multimodality, and Interaction - 6th International Conference of the CLEF Association, CLEF 2015, Toulouse, France, September 8-11, 2015, Proceedings*, p. 352–363, 2015.
- Gabrilovich E., Broder A., Fontoura M., Joshi A., Josifovski V., Riedel L., Zhang T., « Classifying search queries using the Web as a source of knowledge », *ACM Trans. Web*, vol. 3, n<sup>o</sup> 2, p. 5 :1–5 :28, April, 2009.
- Gabrilovich E., Markovitch S., « Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis », *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI'07)* p. 1606–1611, 2007.
- Landauer T. K., Foltz P. W., Laham D., « Introduction to Latent Semantic Analysis », *Discourse Processes* p. 259–284, 1998. 25.
- Manning C. D., Raghavan P., Schütze H., *Introduction to Information Retrieval*, Cambridge University Press, 2008.

- Miao J., Huang J. X., Ye Z., « Proximity-based Rocchio's model for pseudo relevance », *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, ACM, p. 535-544, 2012.
- Rocchio J., « Relevance Feedback in Information Retrieval », *The SMART Retrieval System*, p. 313-323, 1971.
- Savenkov D., Braslavski P., Lebedev M., « Search snippet evaluation at yandex : lessons learned and future directions », *Proceedings of the Second international conference on Multilingual and multimodal information access evaluation*, CLEF'11, Springer-Verlag, Berlin, Heidelberg, p. 14-25, 2011.
- Schütze H., « Automatic Word Sense Discrimination », *Computational Linguistics*, vol. 24, n° 1, p. 97-123, 1998.
- Singh J., Sharan A., « Co-occurrence and Semantic Similarity Based Hybrid Approach for Improving Automatic Query Expansion in Information Retrieval », *Distributed Computing and Internet Technology. 11th International Conference, ICDCIT 2015, Bhubaneswar, India, February 5-8, 2015. Proceedings*, vol. 8956 of *Lecture Notes in Computer Science*, Springer International Publishing, p. 415-418, 2015.
- Voorhees E., Harman D., « Overview of the Seventh Text REtrieval Conference (TREC-7) », *Text REtrieval Conference (TREC) TREC-7 Proceedings*, Department of Commerce, National Institute of Standards and Technology, p. 1-23, 1998. NIST Special Publication 500-242 : The Seventh Text REtrieval Conference (TREC 7).
- Voorhees E. M., « Query expansion using lexical-semantic relations », *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '94, Springer-Verlag New York, Inc., New York, NY, USA, p. 61-69, 1994.
- Wan J., Wang W., Yi J., Chu C., Song K., « Query Expansion Approach Based on Ontology and Local Context Analysis », *Research Journal of Applied Sciences, Engineering and Technology*, vol. 4, n° 16, p. 2839-2843, 2012.
- Xu J., Croft W. B., « Query expansion using local and global document analysis », *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '96, ACM, New York, NY, USA, p. 4-11, 1996.
- Xu J., Croft W. B., « Improving the effectiveness of information retrieval with local context analysis », *ACM Trans. Inf. Syst.*, vol. 18, n° 1, p. 79-112, January, 2000.
- Xu Y., Jones G. J., Wang B., « Query dependent pseudo-relevance feedback based on wikipedia », *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, ACM, New York, NY, USA, p. 59-66, 2009.
- Yang Z., Cai K., Tang J., Zhang L., Su Z., Li J., « Social context summarization », *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, ACM, Beijing, China, p. 255-264, 2011.

---

# Dimensionnalité intrinsèque dans les espaces de représentation des termes et des documents

Vincent Claveau

IRISA-CNRS

Campus de Beaulieu, 35042 Rennes, France

`vincent.claveau@irisa.fr`

---

**RÉSUMÉ.** *L'examen des propriétés des espaces de représentation des documents ou des mots en RI (typiquement,  $\mathbb{R}^n$  avec  $n$  très grand) fournit de précieuses indications pour aider la recherche. Récemment, plusieurs travaux ont montré qu'il était possible d'étudier la dimensionnalité réelle des données, appelée dimensionnalité intrinsèque, en certains points de ces espaces (Houle et al., 2012a). Dans cet article, nous proposons de revisiter cette notion de dimension intrinsèque sous la forme d'un indice noté  $\alpha$  dans le cas particulier de la RI et d'étudier son utilisation pratique en RI. Plus précisément, nous montrons comment son estimation à partir de similarités de type RI, peut être utilisée dans les espaces de représentations des documents et les espaces de représentations de mots (Mikolov et al., 2013 ; Claveau et al., 2014). Ainsi, nous montrons d'une part que l'indice  $\alpha$  aide à caractériser les requêtes difficiles ; d'autre part, dans une tâche d'extension de requête, nous montrons comment cette notion de dimensionnalité intrinsèque appliquée à des mots permet de choisir au mieux les termes à étendre et leurs extensions.*

**ABSTRACT.** *Examining the properties of representation spaces for documents or words in IR (typically  $\mathbb{R}^n$  with  $n$  large) brings precious insights to help the retrieval process. Recently, several authors have studied the real dimensionality of the datasets, called intrinsic dimensionality, in specific parts of these spaces (Houle et al., 2012a). In this paper, we propose to revisit this notion through a coefficient called  $\alpha$  in the specific case of IR and to study its use in IR tasks. More precisely, we show how to estimate  $\alpha$  from IR similarities and to use it in representation spaces used for documents and words (Mikolov et al., 2013 ; Claveau et al., 2014). Indeed, we prove that  $\alpha$  may be used to characterize difficult queries; moreover we show that this intrinsic dimensionality notion, applied to words, can help to choose terms to use for query expansion.*

**MOTS-CLÉS :** *Dimensionnalité intrinsèque, fonctions RSV, thésaurus distributionnels, extension de requête.*

**KEYWORDS:** *Intrinsic dimensionality, RSV scores, distributional thesauri, query expansion.*

---

## 1. Introduction

L'examen des propriétés des espaces de représentation des documents ou des mots en recherche d'information (typiquement,  $\mathbb{R}^n$  avec  $n$  très grand) peut fournir de précieuses indications pour aider la recherche. Il est bien connu que la dimension de l'espace de représentation n'est pas représentative de celle des données ; dans un modèle vectoriel classique en RI, la dimension de l'espace de représentation est le nombre de mots différents dans la collection, mais il est souvent possible de représenter ces mêmes données dans un espace de plus petite dimension. C'est cette constatation qui est au cœur des approches de type LSI/LDA qui réduisent la dimension de l'espace initial (très creux) en un espace plus petit (et plus dense).

L'aspect qui nous intéresse dans cet article est la dimensionnalité intrinsèque des données, non pas du point de vue global de l'ensemble des données, mais localement sur une portion de l'espace. Pour cela, nous nous reposons sur les travaux récents de (Houle *et al.*, 2012a ; Amsaleg *et al.*, 2015) qui permettent de définir et d'estimer la dimensionnalité intrinsèque locale des données (section 2). Cette dernière dépendant de la façon dont le voisinage est défini, et donc de la métrique de distances utilisée, il est nécessaire de l'adapter aux particularités des fonctions de similarités RSV utilisée en RI (section 3). Nous montrons comment elle peut ensuite être utilisée pour analyser l'espace de représentation des documents en RI (section 4), par exemple pour repérer des requêtes difficiles. En section 5, nous menons le même travail sur les espaces de représentations des mots utilisés en sémantique distributionnelle (Claveau et Kijak, 2015). Enfin, nous montrons dans la section 6 une des nombreuses applications possibles de cette notion de dimension intrinsèque sur une tâche d'extension de requêtes à partir de lexiques distributionnels.

## 2. État de l'art

De nombreux travaux se sont intéressés à la caractérisation de la dimensionnalité intrinsèque d'ensemble de données dans des espaces de représentation de grande dimension. Ainsi, les méthodes de plongement (*embedding*) ou de projection construisent des espaces de plus petite dimension dans lesquels les données peuvent être projetés en respectant certaines contraintes de discriminabilité. C'est le cas des méthodes PCA/LSI/LDA (Deerwester *et al.*, 1990 ; Hoffman *et al.*, 2010) ou *manifold learning* (Scholkopf *et al.*, 1998 ; Roweis et Saul, 2000 ; Venna et Kaski, 2006). La dimension intrinsèque de l'ensemble du jeu de données est alors celle de ce sous-espace issu de la projection.

Récemment, (Houle *et al.*, 2012a) a proposé une mesure dite dimension d'expansion généralisée qui définit la dimensionnalité intrinsèque localement en examinant l'augmentation du nombre d'objets rencontrés autour d'un point de l'espace lorsqu'on augmente la distance à ce point. Pour illustrer cela, considérons deux boules de rayons  $\epsilon_1$  et  $\epsilon_2$  et centrées en  $x_1$  et  $x_2$  dans un espace  $\mathbb{R}^m$ , comme illustré en figure 1 (d'après (Houle *et al.*, 2012a)). Notons que si  $\epsilon_2 = a \cdot \epsilon_1$  pour  $a > 1$ , alors le volume de



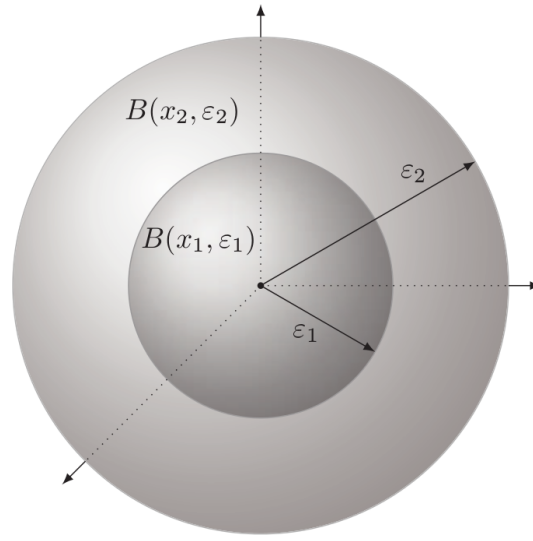


**Figure 1.** D'après (Houle et al., 2012a) ; deux volumes (voisinages de  $x_1$  et  $x_2$ ) dans un espace Euclidien à  $m$  dimensions, avec un centre commun ( $x_1 = x_2$ ).

la sphère extérieure est  $a^m$  fois plus grande que le volume de la sphère intérieure. Plus généralement, on peut s'intéresser au rapport entre ces volumes, qui s'exprime en fonction de la dimension  $m$  de l'espace :

$$\frac{\text{volume}(B(x, \epsilon_1))}{\text{volume}(B(x, \epsilon_2))} = \left(\frac{\epsilon_1}{\epsilon_2}\right)^m$$

De cela, on tire :

$$m = \frac{\ln(\text{volume}(B(x, \epsilon_1))) - \ln(\text{volume}(B(x, \epsilon_2)))}{\ln \epsilon_1 - \ln \epsilon_2}$$

L'idée au cœur de la dimensionalité intrinsèque est de détourner ce calcul de la dimension  $m$  en considérant non pas le volume, mais le nombre de points contenus dans ce volume comme un estimateur. En notant  $|B(x, \epsilon)|$  le nombre de points contenus dans le volume  $B(x, \epsilon)$ , on a donc :

$$\hat{m} = \frac{\ln |B(x, \epsilon_1)| - \ln |B(x, \epsilon_2)|}{\ln \epsilon_1 - \ln \epsilon_2}$$

La dimension mesurée n'est plus celle de l'espace de représentation, mais celle propre aux données. Par ailleurs, il est important de noter que cette estimation est locale en un point  $x$  (en prenant le centre des boules  $x_1 = x_2 = x$ ). Il ne s'agit donc pas d'une caractérisation de l'ensemble de points comme avec une PCA par exemple.

Ce modèle de dimensionalité intrinsèque a été utilisé récemment dans des applications d'analyse et de construction de structures d'index pour des recherches par

similarités (Beygelzimer *et al.*, 2006 ; Houle *et al.*, 2012b ; Houle et Nett, 2013) et pour la détection d'anomalies (de Vries *et al.*, 2012).

### 3. Utilisation dans un contexte de RI

L'intérêt de la dimensionnalité intrinsèque pour la RI est sa capacité à caractériser le voisinage d'une requête en terme de documents l'environnant. Si l'espace autour de la requête est de très haute dimensionnalité, c'est le signe qu'une petite variation de distance change énormément l'ensemble des documents jugés proches de la requête. Autrement dit, une haute dimensionnalité implique une haute indiscriminabilité des documents autour de la requête (Houle *et al.*, 2012a). C'est cette propriété que nous voulons donc utiliser dans nos travaux, à condition de pouvoir adapter le cadre précédent aux cas particuliers des similarités utilisées en RI.

#### 3.1. Limites

La définition de la dimensionnalité intrinsèque donnée précédemment vaut pour un espace dans laquelle la métrique utilisée est une distance, typiquement une distance L2 (euclidienne). C'est cette distance qui sert à définir les ensembles de points contenus dans les boules de différents diamètres. Par exemple, pour une boule centrée en  $x$  et de rayon  $r > 0$ , les points  $d_i$  considérés sont ceux à une distance L2 inférieure ou égale à  $r$  de  $x$  (soit encore de norme L2 inférieure ou égale à  $r$  si la boule est centrée sur l'origine).

En RI, la distance L2 est rarement utilisée comme fonction de pertinence (par la suite RSV, pour *Relevance Status Value*). En revanche, le cosinus a été largement utilisé comme fonction RSV dans le modèle vectoriel. Il permet de caractériser l'angle entre un vecteur requête et un vecteur document : deux vecteurs proches en terme d'angle (c'est-à-dire formant un angle petit) auront un cosinus élevé. Comme cela a été montré par (Houle *et al.*, 2012a), on peut adapter le calcul de dimensionnalité intrinsèque à ce cas des distances angulaires. L'idée est la même que précédemment : pour un vecteur requête donné, on examine le nombre de vecteurs dans ses voisinages à  $\epsilon_1$  et  $\epsilon_2$ , c'est-à-dire avec une distance angulaire inférieure à un seuil  $\epsilon_1$  et  $\epsilon_2$ .

Cependant, la plupart des fonctions RSV modernes peuvent s'écrire :

$$RSV(q, d) = \sum_{t \in q} w_q(t) \cdot w_d(t)$$

avec  $w_d(t)$  et  $w_q(t)$  qui sont respectivement les poids du terme  $t$  dans la requête  $q$  et dans le document  $d$ , comme illustré dans le tableau 1 (d'après (Lv et Zhai, 2011)).

|       | $w_q(t)$                           | $w_d(t)$                                                                                                                                                                                                                                                                                                         |
|-------|------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| BM25+ | $\frac{(k_3+1)c(t,q)}{k_3+c(t,q)}$ | $\left( \frac{(k_1+1)c(t,d)}{k_1(1-b+b \cdot dl(d)/avdl)+c(t,d)} + \delta \right) \cdot \log \frac{N+1}{df(t)}$<br>avec $k_1, k_3, b$ et $\delta$ des paramètres                                                                                                                                                 |
| PL2   | $c(t, q)$                          | $\frac{tfn(t,d) \cdot \log_2(tfn(t,d) \cdot \lambda_t) + \log_2 e \cdot (1/\lambda_t - tfn(t,d)) + 0.5 \log_2(2\pi \cdot tfn(t,d))}{tfn(t,d)+1}$<br>avec $tfn(t, d) = c(t, d) \cdot \log_2 \left( 1 + c \cdot \frac{avdl}{dl(d)} \right)$<br>$c > 0$ un paramètre de recherche et $\lambda_t = \frac{N}{c(t,C)}$ |
| Dir   | $c(t, q)$                          | $\log \left( \frac{\mu}{dl(d)+\mu} + \frac{c(t,d)}{(dl(d)+\mu)p(t C)} \right)$<br>$\mu > 0$ un paramètre de lissage                                                                                                                                                                                              |
| Piv   | $c(t, q)$                          | $\frac{1+\log(1+\log(c(t,d)))}{1-s+s \cdot dl(d)/avdl} \cdot \log \frac{N+1}{df(t)}$ si $c(t, d) > 0$ et 0 sinon<br>avec $s$ un paramètre                                                                                                                                                                        |

**Tableau 1.** Fonctions de pondération d'un terme dans une requête ou un document pour différents systèmes de RI de l'état-de-l'art : BM25+ (Robertson et al., 1998 ; Lv et Zhai, 2011), Divergence From Randomness PL2 (Amati et Rijsbergen, 2002 ; Fang et al., 2011), Modèle de langue avec lissage Dirichlet Dir (Zhai et Lafferty, 2001), Pivoted Normalization Piv (Singhal, 2001).

avec les notations suivantes :

- $c(t, d)$  nombre d'occurrences du terme  $t$  dans le document  $d$
- $c(t, q)$  nombre d'occurrences du terme  $t$  dans la requête  $q$
- $N$  nombre total de documents dans la collection
- $df(t)$  nombre de documents contenant le terme  $t$
- $dl(d)$  longueur du document  $d$
- $avdl$  longueur moyenne des documents
- $c(t, C)$  nombre d'occurrences du terme  $t$  dans la collection  $C$
- $p(t|C)$  probabilité d'un terme  $t$  pour un modèle de langue de la collection

Ces fonctions de pertinence peuvent alors s'interpréter comme un simple produit scalaire entre le vecteur document  $d$  et le vecteur requête  $q$ , noté  $\langle q, d \rangle$  par la suite. La seule différence par rapport au cosinus est donc de ne pas imposer une normalisation L2 des vecteurs. Cela peut se visualiser sur les figures 2 et 3 qui présente la distribution des normes L2 des documents de la collection Tipster (cf. section 4.1) et la rapport entre la longueur du document ( $dl(d)$ ) et sa norme L2, le tout avec une pondération de type BM25+ (version modifiée de BM25 selon (Lv et Zhai, 2011)).

L'absence de normalisation a une conséquence importante puisqu'il apparaît difficile d'utiliser le même principe que précédemment pour calculer la dimension intrinsèque. En effet, pour une requête  $q$  donnée et deux valeurs de produit scalaire

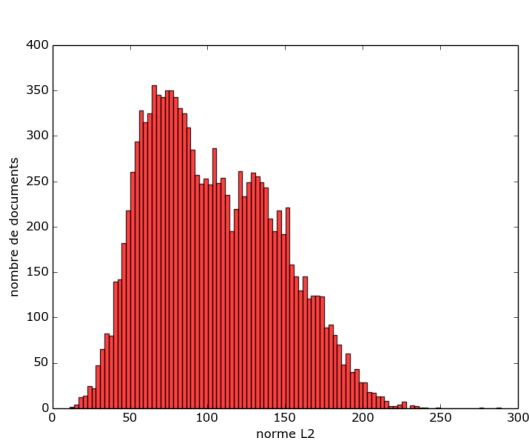


**Figure 2.** Distribution des normes L2 des documents de la collection Tipster avec une pondération BM25+ (version modifiée selon (Lv et Zhai, 2011)).

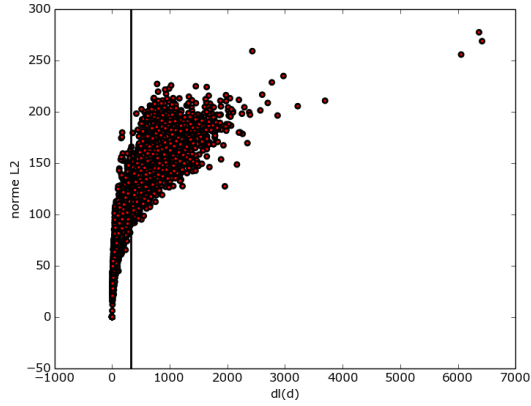


**Figure 3.** Norme L2 des documents de la collection Tipster selon leur longueur  $dl(d)$  avec une pondération BM25+ (version modifiée selon (Lv et Zhai, 2011)); la ligne verticale représente la longueur moyenne ( $avgdl$ ).

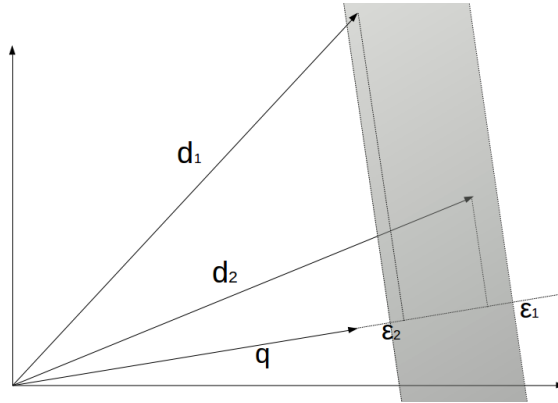


**Figure 4.** En gris : portion de l'espace défini par l'ensemble des points dont le produit scalaire avec un vecteur  $q$  normé est entre  $\epsilon_1$  et  $\epsilon_2$ .

$\epsilon_1$  et  $\epsilon_2$  ( $\epsilon_1 \geq \epsilon_2$ ), la portion de l'espace pouvant contenir des points  $d_i$  tels que  $\epsilon_1 \geq \langle d_i, q \rangle \geq \epsilon_2$  est infinie, comme illustrée en deux dimensions en figure 4.

### 3.2. Estimation par l'exposant d'une loi de puissance

Malgré la limite liée à l'utilisation de similarité sous forme de produits scalaires, on souhaite tout de même caractériser la dimension intrinsèque, ou du moins l'indis-

criminabilité, localement dans notre espace de représentation. Pour cela, nous nous inspirons des travaux de (Levina et Bickel, 2004 ; Amsaleg *et al.*, 2015) dans lesquels il a été montré qu’il était possible d’estimer la dimension intrinsèque à partir de la répartition des distances (par exemple L2 dans ces travaux) entre un vecteur requête et l’ensemble des autres vecteurs. Plutôt que de raisonner sur les rapports entre volumes, l’idée est de caractériser cette indiscriminabilité en examinant l’évolution du nombre de voisins (documents jugés proches de la requête) selon le score RSV. Cette courbe peut être interprétée (au sens des abscisses près) comme une fonction de répartition d’une variable aléatoire  $X$  représentant dans notre cas le score RSV entre le vecteur requête et un vecteur document. Plus précisément, puisqu’on s’intéresse au comportement local, on examine uniquement la portion de cette courbe RSV/nombre de voisins pour les plus grands scores RSV. L’hypothèse que nous faisons est que la distribution des RSV sur cette portion suit une loi de puissance, c’est-à-dire qu’elle s’exprime sous la forme :

$$f(x) = \lambda x^{-\alpha} \text{ avec } \lambda \text{ une constante et } \alpha > 1 \quad [1]$$

C’est cet exposant  $\alpha$  qui est caractéristique de la dimensionalité ou de l’indiscriminabilité des données.

Dans notre cas, nous disposons de  $n$  observations  $x_i$ , c’est-à-dire des valeurs de RSV entre une requête et ses  $n$  plus proches voisins ( $n$  meilleurs scores RSV). À partir de ces observations, il est possible d’estimer  $\alpha$ . Parmi les différentes méthodes proposées dans la littérature, celle reposant sur la log-vraisemblance a été montrée comme la moins biaisée (Clauset *et al.*, 2009). Soient les observations  $x_i$  toutes supérieures à un seuil  $x_{min}$  on peut alors estimer  $\alpha$  par :

$$\hat{\alpha} = 1 + n \cdot \sum_{i=1}^n \ln \frac{x_i}{x_{min}} \quad [2]$$

Dans les expériences rapportées dans les sections suivantes, pour estimer  $\alpha$  nous considérons  $n = 100$  observations  $x_i$ , c’est-à-dire les 100 plus grands scores RSV.

#### 4. Espace des documents

Dans cette section, nous souhaitons étudier l’utilisation de l’indice de dimensionalité  $\alpha$ , tel que défini dans la section précédente, dans un cadre classique de RI. Que les documents soient explicitement contenus dans des espaces euclidiens, comme avec le modèle vectoriel, ou que ce soit implicitement, au travers de fonctions de similarité pouvant s’interpréter comme des opérations dans des espaces euclidiens, nous montrons comment l’indice  $\alpha$  peut-être utilisé pour caractériser les documents au voisinage d’une requête.

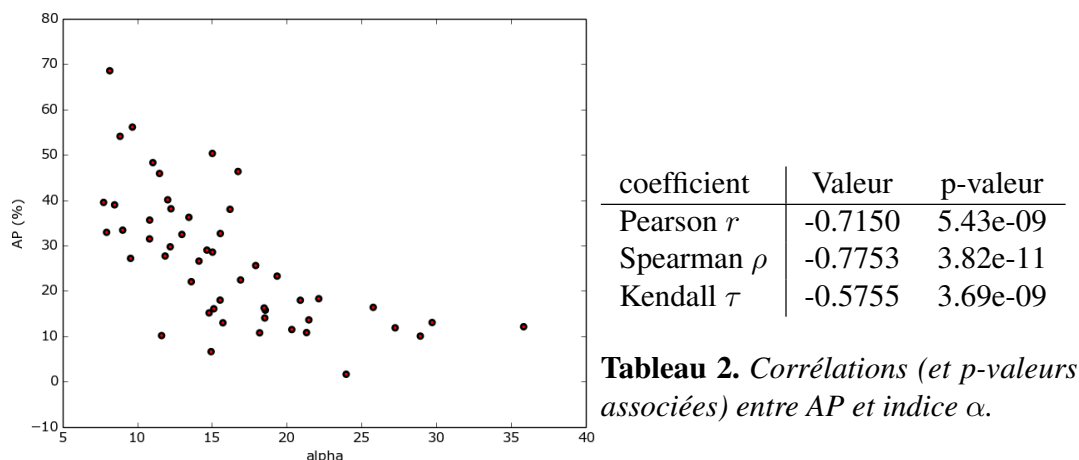


**Figure 5.** Performances (AP) des requêtes selon leur indice  $\alpha$  avec un modèle BM25+.

#### 4.1. Données et mesures d'évaluation

La collection de RI que nous utilisons est celle développée pour le projet Tipster et utilisée dans le cadre de TREC. Elle contient plus de 170 000 documents et cinquante requêtes. Ces requêtes sont composées de plusieurs champs (la requête à proprement parler, un champ narratif détaillant les critères de pertinence) ; dans les expériences rapportées ci-dessous, nous n'utilisons que le champ requête. Les performances pour cette tâche de RI sont également classiquement mesurées en précision à différents seuils (P@x), R-prec, MAP.

#### 4.2. Indice de requête difficile

La distribution des documents autour de la requête, ou plus précisément, la distribution des distances entre la requête et ses plus proches documents (selon le score RSV), peut aider à caractériser la difficulté de la requête. Pour vérifier cela, nous mettons en rapport l'indice  $\alpha$  estimé (cf. équation 2) dans le voisinage de chaque requête avec l'Average Precision (AP) obtenue pour cette requête. Cela est illustré dans la figure 5 sur la collection Tipster, la fonction RSV utilisée est BM25+ (Lv et Zhai, 2011).

Le nuage de points obtenu illustre bien la dépendance attendue entre l'indice  $\alpha$ , dérivé de la dimension intrinsèque, et la performance. Pour mesurer précisément cette relation, nous rapportons dans le tableau 2 les corrélations de Pearson, Spearman et Kendall (et les p-valeurs associées) entre la liste des requêtes ordonnées par AP et la même liste ordonnée par  $\alpha$ . La corrélation inverse entre  $\alpha$  et les performances obtenues pour la requête apparaît nettement : plus la dimension intrinsèque autour d'une

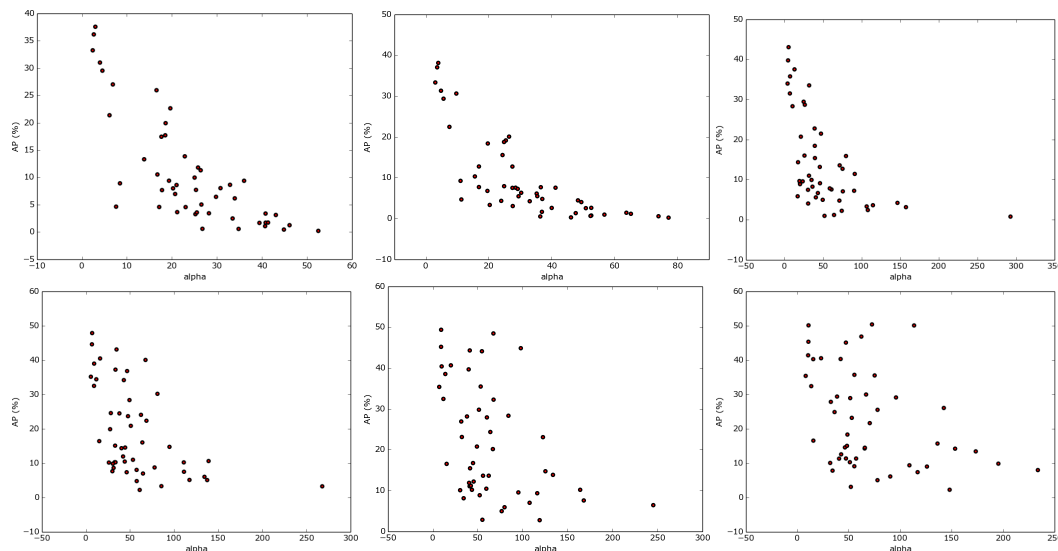


**Figure 6.** Performances (AP) des requêtes selon leur indice  $\alpha$  avec un modèle Dirichlet et dans le sens de lecture  $\mu = 1, 25, 250, 1000, 2500, 4000$ .

requête est grande, moins bonne est la performance du moteur de recherche sur cette requête.

Les mêmes observations sont valables avec d'autres fonctions RSV. Dans les figures 6, nous montrons l'évolution  $\alpha/AP$  en fonction du coefficient  $\mu$  d'un modèle de langue avec lissage Dirichlet. Ces figures permettent d'observer un des effets du lissage : quand celui-ci est élevé, les documents ont tendance à avoir des valeurs similaires pour les termes de la requête et se trouvent donc tous à des distances comparables de la requête, ce qui a pour effet d'augmenter la dimensionalité intrinsèque et donc l'indice  $\alpha$ . Quand  $\mu$  est faible, la corrélation (inverse) est donc très nette mais a tendance à disparaître quand  $\mu$  augmente. Cela peut s'observer directement dans la figure 7.

## 5. Espace des mots

Depuis quelques années, de nombreux travaux se sont appliqués à projeter les mots dans des espaces de représentations continues, comme les espaces vectoriels. Il est ainsi possible de calculer, au sein de ces espaces, des proximités sémantiques à l'aide de distances/similarités vectorielles. Pour ce faire, ces techniques dites de plongement (*embedding*) reposent sur une hypothèse classique de sémantique distributionnelle : les mots proches partagent des contextes proches. Ainsi en examinant le contexte d'apparition des mots (c'est-à-dire les mots précédents et suivants), et en calculant des proximités entre ces contextes, ces méthodes d'analyse distributionnelle induisent des proximités entre les mots représentés par ces contextes.

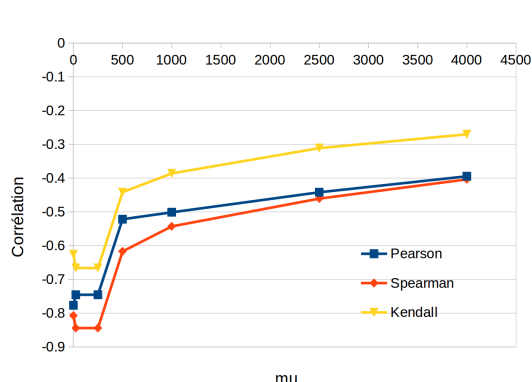


**Figure 7.** Évolutions des valeurs de corrélations ( $r$  de Pearson,  $\rho$  de Spearman et  $\tau$  de Kendall) selon le coefficient de lissage  $\mu$  dans le modèle de langue Dirichlet.

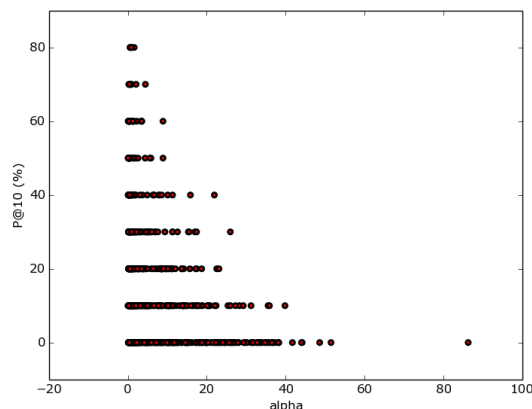


**Figure 8.** Performances ( $P@10$  sur WN+Moby) des mots-requêtes selon leur indice  $\alpha$  avec un modèle BM25 selon la méthode de (Claveau et Kijak, 2015).

Dans des travaux récents, (Claveau et Kijak, 2015) ont montré que les techniques de RI pouvaient être utilisées pour construire ces représentations de mots. En calculant par exemple des similarités de type Okapi-BM25 entre les ensembles de contextes de deux mots, ils ont montré qu'il était possible de construire des thésaurus distributionnels de qualité supérieure à l'état-de-l'art. C'est cette approche que nous reprenons ici. Comme nous l'avons fait pour les documents, nous nous intéressons dans cette partie à la dimension intrinsèque dans cet espace de représentation des mots. Puisqu'il repose lui-aussi sur des similarités RSV, nous reprenons la technique d'estimation de  $\alpha$  pour caractériser localement l'indiscriminabilité en tout point de l'espace.

### 5.1. Données et évaluation

Les données que nous utilisons pour nos expériences de construction sont celles utilisées dans plusieurs travaux sur la sémantique distributionnelle. Cela va nous permettre de comparer nos résultats à ceux publiés. Le corpus utilisé pour collecter les contextes est le corpus AQUAINT-2 ; il est composé d'articles de presse en anglais et compte 380 millions de mots. Parmi eux, les mots que nous considérons pour entrées de notre lexique sont les noms communs apparaissant au moins 10 fois dans le corpus, soit 25 000 noms différents. Les contextes de toutes les occurrences de ces mots sont donc collectés ; dans les expériences rapportées ci-dessous, le contexte d'un mot est composé des deux mots à droite et deux mots à gauche du nom visé, en gardant leur position. Par exemple, dans l'extrait : "... all forms of restriction on freedom of expression, threats ...", les mots restriction-2, on-1, of+1, expression+2 sont ajoutés à l'ensemble des contextes freedom.



Comme dans les travaux de (Claveau et Kijak, 2015), nous utilisons conjointement WordNet (Miller, 1990) et Moby (Ward, 1996) pour l'évaluation intrinsèque des thésaurus produits. Ces deux ressources offrent des caractéristiques complémentaires : WordNet recense des liens sémantiques forts (synonymes ou quasi-synonymes) alors que Moby recense une plus grande variété de liens (hyperonymes, méronymes, co-hyponymie...). Une description détaillée des liens considérés par ces ressources est donnée par (Ferret, 2013) ou (Claveau *et al.*, 2014). Ainsi, WN propose en moyenne 3 voisins pour 10 473 des noms du corpus AQUAINT-2 et Moby 50 voisins en moyenne pour 9 216 noms. Combinées, ces deux ressources couvrent 12 243 noms du corpus avec 38 voisins en moyenne. Le nombre de noms dans les listes de référence et la variété des relations sémantiques considérées font de ces données un jeu d'évaluation très complet par rapport à d'autres *benchmarks* parfois utilisés tels que WordSim 353 (Gabrilovich et Markovitch, 2007).

Pour cette tâche de construction lexicale sémantique, les mesures d'évaluation sont celles habituellement utilisées en RI.

## 5.2. Indice de requête difficile

Comme en section 4.2, nous examinons l'éventuel rapport entre l'indice  $\alpha$  d'un mot et les performances obtenues en recherchant ses voisins sémantiques. Cela est illustré en figure 8 ; nous utilisons cette fois P@10 comme mesure de performance du fait de l'utilisation que nous faisons des voisins en section suivante. Le tableau 3 rapporte les valeurs de corrélation correspondantes.

| coefficient     | Valeur  | p-valeur | coefficient     | Valeur  | p-valeur  |
|-----------------|---------|----------|-----------------|---------|-----------|
| Pearson $r$     | -0.3607 | 4.46e-24 | Pearson $r$     | -0.2268 | 4.65e-10  |
| Spearman $\rho$ | -0.4451 | 3.69e-37 | Spearman $\rho$ | -0.8371 | 9.79e-195 |
| Kendall $\tau$  | -0.3335 | 8.75e-42 | Kendall $\tau$  | -0.6389 | 1.53e-148 |

**Tableau 3.** Corrélations (et *p*-valeurs associées) entre P@10 et indice  $\alpha$

**Tableau 4.** Corrélations (et *p*-valeurs associées) entre nombre d'occurrences et indice  $\alpha$

La corrélation négative entre  $\alpha$  et P@10 est présente mais relativement faible. En toute généralité, l'indice  $\alpha$  peut être utile mais n'est pas fiable à lui seul pour prédire la qualité du thésaurus distributionnel produit.

## 5.3. Analyse fréquentielle

Un autre élément d'analyse est l'étude du lien entre l'indice  $\alpha$  et la fréquence du mots requête dans le corpus à partir duquel ses contextes sont collectés (et donc équivalent au  $dl(d)$  dans cette tâche). En effet, plusieurs travaux précédents

(Ferret, 2013 ; Claveau *et al.*, 2014) ont montré l'influence de ce nombre d'occurrences sur la performance à la tâche de recherche des voisins sémantiques. Comme précédemment, nous reportons dans le tableau 4 les nombres d'occurrences des mots requêtes en fonction de leur indice  $\alpha$ .

La corrélation est forte (et non linéaire puisque  $r$  est faible). Cela s'explique par le fait que le faible nombre d'occurrences, et donc le faible nombre de contextes collectés pour représenter un mot, se traduit par une description vectorielle insuffisamment détaillée du mot pour laquelle l'indiscriminabilité des voisins a donc tendance à être importante.

## 6. Extension de requête

Les lexiques sémantiques, tels que ceux générés par les techniques distributionnelles décrites précédemment, peuvent être utilisées dans beaucoup d'applications. En recherche d'information, ils peuvent notamment servir à étendre les requêtes : les voisins sémantiques des mots de la requête sont ajoutés à celle-ci. Depuis le travail sémi-nal et les conclusions négatives de (Voorhees, 1994) dans lequel WordNet était utilisé, de nombreux travaux ont au contraire montré l'intérêt de l'extension de requête, que ce soit avec des lexiques construits manuellement ou générés automatiquement. Dernièrement, (Claveau et Kijak, 2015) ont montré que les lexiques distributionnels qu'ils génèrent apportait des gains très substantiels en terme de MAP à la recherche, et que cette tâche de RI pouvait même être avantageusement utilisée comme une procédure d'évaluation des lexiques. Dans cette section, nous reprenons ce cadre applicatif en examinant comment notre indice  $\alpha$  dérivé de la dimensionnalité intrinsèque peut être utilisé dans ce cadre.

### 6.1. Mise-en-œuvre

A des fins de comparaison, nous adoptons le même cadre expérimental que (Claveau et Kijak, 2015) :

- la collection de RI que nous utilisons est la même que précédemment, à savoir Tipster.
- le lexique distributionnel est celui étudié en section 5, identique à celui utilisé dans (Claveau et Kijak, 2015).
- le système de recherche d'information que nous utilisons est Indri (Metzler et Croft, 2004 ; Strohman *et al.*, 2005), connu pour offrir des performances état-de-l'art. Ce système probabiliste implémente une combinaison de modèle de langue (Ponte et Croft, 1998), tel que vu précédemment, et de réseaux d'inférence (Turtle et Croft, 1991) permettant d'utiliser des opérateurs tels que ET OU... Dans les expériences rapportées ci-dessous, nous l'utilisons avec des réglages standard, à savoir un lissage de Dirichlet ( $\mu = 2500$ ). Ce système de RI offre l'avantage de disposer d'un langage de requête complexe qui nous permet d'inclure les mots du lexique distributionnel

| Extension                    | MAP    | R-Prec | P@5   | P@10  | P@50   | P@100  |
|------------------------------|--------|--------|-------|-------|--------|--------|
| Sans                         | 21,78  | 30,93  | 92,80 | 89,40 | 79,60  | 70,48  |
| avec extension               | +13,80 | +9,58  | +2,16 | +4,03 | +5,58  | +8,26  |
| avec extension et filtrage 1 | +15,73 | +9,27  | +2,22 | +4,96 | +9,63  | +14,41 |
| avec extension et filtrage 2 | +22,83 | +13,00 | +2,56 | +6,31 | +14,10 | +21,39 |

**Tableau 5.** *Gains relatifs de performance (%) par extension de requête sans filtrage et avec filtrage selon  $\alpha$  sur les mots de la requête (filtrage 1) et sur les mots en extension (filtrage 2).*

en exploitant au mieux le modèle par réseau d'inférence à l'aide de l'opérateur dédié '#syn' qui permet d'agréger les comptes des mots considérés comme synonymes (voir la documentation d'Indri pour plus de détails). Pour supprimer les effets de flexions (pluriel) sur les résultats, les formes pluriel et singulier des noms de la requêtes sont ajoutées, que ce soit dans les requêtes non étendues avec les synonymes ou celles étendues par les voisins sémantiques.

– nous évaluons les performances avec les mesures classiques de RI, en comparant les résultats avec et sans extension.

## 6.2. Expériences

Beaucoup d'utilisations des indices  $\alpha$  des requêtes elles-mêmes ou des mots de notre lexique distributionnel sont possibles. Nous rapportons ci-dessous les résultats de deux expériences où nous utilisons l'indice  $\alpha$  des mots pour filtrer les extensions, selon deux mises-en-œuvre. Dans la première mise-en-œuvre, notée filtrage 1, nous calculons  $\alpha$  pour chacun des mots de la requête originale, et nous ne proposerons des extensions (via le lexique distributionnel) que pour ceux inférieur à un certain seuil (fixé au  $\alpha$  de tous les mots de toutes les requêtes dans l'expérience). Dans la seconde mise-en-œuvre, notée filtrage 2, le filtrage 1 est opéré, et en plus, nous filtrons également les mots ajoutés en extension selon leur indice  $\alpha$ . Les résultats sont donnés dans le tableau 5 ; y sont rapportés les performances des requêtes originales, et les gains par rapport à ces performances obtenues sans extensions. Des tests de significativité statistique (wilcoxon avec  $p = 0.05$ ) ont été calculés : la version avec extension est comparée à la version sans extension, les versions avec filtrage (1 et 2) sont comparées avec la version avec extension (sans filtrage). Les résultats non significatifs sont en italique. De ces expériences, il ressort que les bons résultats de l'extension de requêtes par lexique distributionnel, déjà noté dans (Claveau et Kijak, 2015), est légèrement amélioré par un filtrage sur les mots à étendre (filtrage 1) mais sans que la différence soit statistiquement significative. En revanche, quand les extensions sont en plus elles-mêmes filtrées, le gain sur la version avec extension simple est alors plus important et statistiquement significatif. En pratique, un examen des requêtes étendues filtrées montre que les mots dont l'indice  $\alpha$  dépasse le seuil sont effectivement ceux polysémiques ou généraux tels que : *choice*, *term*, *use*, *young*... Les bons résultats du filtrage

vont donc à l'encontre des faibles corrélations vues en 5.2 entre  $\alpha$  et la précision sur les thésaurus de référence. Cette différence entre les résultats de l'évaluation intrinsèque (par thésaurus) et extrinsèque (via une tâche de RI dans notre cas) est en ligne avec les conclusions de (Claveau et Kijak, 2015).

## 7. Conclusion

Dans cet article, nous avons montré comment adapter la notion de dimensionnalité intrinsèque (Houle *et al.*, 2012a) aux similarités RSV utilisées en RI. En nous inspirant des travaux de (Amsaleg *et al.*, 2015), nous avons défini l'indice  $\alpha$  que nous avons ensuite utilisé pour caractériser l'indiscriminabilité des voisins d'un point quelconque de l'espace de représentation. Appliqué à une tâche classique de RI (une requête pour trouver des documents), nous avons montré le lien entre cet indice calculé pour une requête et les performances du système de RI pour cette même requête. Nous avons appliqué la même approche en sémantique distributionnelle sur un espace contenant cette fois-ci des mots, mais utilisant les mêmes fonctions RSV (Claveau et Kijak, 2015). Nous avons alors montré l'intérêt de cet indice  $\alpha$  pour améliorer les techniques d'extension de requête s'appuyant sur de tels lexiques distributionnels.

Beaucoup de perspectives sont ouvertes par ce travail. Tout d'abord, d'un point de vue théorique, la caractérisation de la dimensionnalité intrinsèque dans le cas de mesures de similarité (comme celles utilisées en RI) plutôt que de distances soulève plusieurs problèmes. Dans cet article, par analogie avec le travail de (Amsaleg *et al.*, 2015), nous avons utilisé l'indice  $\alpha$  en supposant une distribution en loi de puissance. Même si cette distribution est vérifiée expérimentalement, le lien précis avec la dimensionnalité intrinsèque telle que définie par (Houle *et al.*, 2012a) et  $\alpha$  est à préciser. D'autre part, on souhaite avoir une caractérisation locale de la dimensionnalité, mais l'estimation de  $\alpha$  requiert un nombre minimal d'observations (RSV des plus proches voisins) pour être fiable. Ces deux contraintes sont parfois contradictoires pour certaines portions de l'espace dans lesquelles les plus proches voisins sont très éloignés.

D'un point de vue applicatif, on peut imaginer beaucoup d'utilisation de cette notion de dimensionnalité intrinsèque. Elle peut permettre de proposer des représentations de type LSI, LDA mais adaptées à la complexité locale. Plus pragmatiquement, lors d'une recherche en ligne, lors de la phase de rédaction de la requête, elle peut aussi aider à trouver les mots précis dans la requête qui provoque la plus forte montée d' $\alpha$  et demander à l'utilisateur d'utiliser un autre mot ou une autre formulation, ce qui permettra probablement de trouver des résultats plus pertinents, comme nous l'avons vu en section 4.

## 8. Bibliographie

- Amati G., Rijsbergen C. J. V., « Probabilistic models of information retrieval based on measuring the divergence from randomness », *ACM Trans. Inf. Syst.*, vol. 20, p. 357-389, October, 2002.
- Amsaleg L., Ouassama C., Furon T., Girard S., Houle M. E., Kawarabayashi K.-I., « Estimating Local Intrinsic Dimensionality », *21st Conf. on Knowledge Discovery and Data Mining, KDD2015*, Sidney, Australia, August, 2015.
- Beygelzimer A., Kakade S., Langford J., « Cover trees for nearest neighbors », *Proc. of International Conference on Machine Learning (ICML)*, p. 97-104, 2006.
- Clauset A., Shalizi C. R., Newman M. E. J., « Power-Law Distributions in Empirical Data », *SIAM Review*, vol. 51, n° 4, p. 661-703, 2009.
- Claveau V., Kijak E., « Thésaurus distributionnels pour la recherche d'information et vice-versa », *Revue des Sciences et Technologies de l'Information - Série Document Numérique*, 2015.
- Claveau V., Kijak E., Ferret O., « Improving distributional thesauri by exploring the graph of neighbors », *International Conference on Computational Linguistics, COLING 2014*, Dublin, Irlande, August, 2014.
- de Vries T., Chawla S., Houle M. E., « Density-preserving projections for large-scale local anomaly detection », *Knowledge Information Systems*, vol. 32, n° 1, p. 25-52, 2012.
- Deerwester S., Dumais S. T., Furnas G. W., Landauer T. K., Harshman R., « Indexing by Latent Semantic Analysis », *Journal of the American Society for Information Science*, 1990.
- Fang H., Tao T., Zhai C., « Diagnostic evaluation of information retrieval models », *ACM Trans. Inf. Syst.*, 2011.
- Ferret O., « Identifying Bad Semantic Neighbors for Improving Distributional Thesauri », *51<sup>st</sup> Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, Sofia, Bulgaria, p. 561-571, 2013.
- Gabrilovich E., Markovitch S., « Computing semantic relatedness using wikipedia-based explicit semantic analysis », *20<sup>th</sup> International Joint Conference on Artificial Intelligence (IJCAI 2007)*, p. 6-12, 2007.
- Hoffman M., Bach F. R., Blei D. M., « Online Learning for Latent Dirichlet Allocation », in J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, A. Culotta (eds), *Advances in Neural Information Processing Systems 23*, Curran Associates, Inc., p. 856-864, 2010.
- Houle M. E., Kashima H., Nett M., « Generalized expansion dimension », *Proc. of the 12th IEEE International Conference on Data Mining Workshops (ICDMW)*, p. 587-594, 2012a.
- Houle M. E., Ma X., Nett M., Oria V., « Dimensional testing for multi-step similarity search », *Proc. of the 12th IEEE International Conference on Data Mining (ICDM)*, p. 299-308, 2012b.
- Houle M. E., Nett M., « Rank cover trees for nearest neighbor search », *International Conference on Similarity Search and Applications (SISAP)*, p. 16-29, 2013.
- Levina E., Bickel P. J., « Maximum likelihood estimation of intrinsic dimension », *Advances in Neural Information Processing Systems (NIPS)*, 2004.

- Lv Y., Zhai C., « Lower-bounding Term Frequency Normalization », *Proc. of the 20th ACM International Conference on Information and Knowledge Management*, CIKM '11, ACM, New York, NY, USA, p. 7-16, 2011.
- Metzler D., Croft W., « Combining the Language Model and Inference Network Approaches to Retrieval », *Information Processing and Management Special Issue on Bayesian Networks and Information Retrieval*, vol. 40, n° 5, p. 735-750, 2004.
- Mikolov T., Yih W.-t., Zweig G., « Linguistic Regularities in Continuous Space Word Representations », *2013 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (NAACL HLT 2013)*, Atlanta, Georgia, p. 746-751, 2013.
- Miller G. A., « WordNet : An On-Line Lexical Database », *International Journal of Lexicography*, 1990.
- Ponte J. M., Croft W. B., « A language modeling approach to information retrieval », *Proc. of the 21st Annual international ACM SIGIR Conference on Research and Development in information Retrieval (SIGIR '98)*, p. 275-281, 1998.
- Robertson S. E., Walker S., Hancock-Beaulieu M., « Okapi at TREC-7 : Automatic Ad Hoc, Filtering, VLC and Interactive », *Proc. of the 7<sup>th</sup> Text Retrieval Conference, TREC-7*, p. 199-210, 1998.
- Roweis S. T., Saul L. K., « Nonlinear dimensionality reduction by locally linear embedding », *Science*, vol. 290, n° 5500, p. 2323-2326, 2000.
- Scholkopf B., Smola A. J., Muller K.-R., « Nonlinear component analysis as a kernel eigenvalue problem », *Neural Computation*, vol. 10, n° 5, p. 1299-1319, 1998.
- Singhal A., « Modern information retrieval : a brief overview », *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 2001.
- Strohman T., Metzler D., Turtle H., Croft W., Indri : A language-model based search engine for complex queries (extended version), Technical report, CIIR, 2005.
- Turtle H., Croft W., « Evaluation of an Inference Network-Based Retrieval Model », *ACM Transactions on Information System*, vol. 9, n° 3, p. 187-222, 1991.
- Venna J., Kaski S., « Local multidimensional scaling », *Neural Networks*, 2006.
- Voorhees E. M., « Query Expansion Using Lexical-semantic Relations », *Proc. of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '94, Springer-Verlag New York, Inc., New York, NY, USA, p. 61-69, 1994.
- Ward G., « Moby Thesaurus », , Moby Project, 1996.
- Zhai C., Lafferty J. D., « A study of smoothing methods for language models applied to ad hoc information retrieval », *Proc. of the SIGIR conference*, p. 334-342, 2001.

---

# Expansion de requêtes par apprentissage <sup>1</sup>

Ahlem Bouziri\* — Chiraz Latiri \* — Eric Gaussier\*\*

\* Université de la Manouba (ISAMM)- LIPAH (FST), Tunisie

\*\* Université Joseph Fourier-Laboratoire d'Informatique de Grenoble, France

---

**RÉSUMÉ.** Cet article propose une approche d'expansion automatique de requêtes par apprentissage. L'expansion de requêtes se fait par l'ajout de termes provenant de règles d'association entre termes. Le problème d'expansion de requêtes est modélisé comme un problème de classification supervisée qui vise à déterminer les règles d'association les plus adaptées pour enrichir une requête donnée. Un ensemble de données d'entraînement est construit en utilisant un algorithme d'exploration de règles d'association pertinentes, basé sur les algorithmes génétiques. La classification se fait par la méthode de l'arbre de décision et par la méthode Random Forest. Les expérimentations sont menées sur la collection de textes en français SDA95 de la campagne d'évaluation CLEF 2003. Les résultats montrent une amélioration des performances de la tâche RI

**ABSTRACT.** We propose in this paper a learning query expansion approach using association rules. The query expansion problem is modeled as a supervised classification problem which aims at identifying the appropriate set of association rules to expand a given query. A training data set is generated using a GA based exploring algorithm of the association rules space. Classification is made by the method of the decision tree and the Random Forest method. The experiments are conducted on the French texts SDA95 collection of CLEF evaluation campaign 2003. The results show an improvement in task performance IR.

**MOTS-CLÉS :** règles d'association, expansion de requêtes, apprentissage supervisé.

**KEYWORDS:** Association rules, query expansion, supervised learning.

---

---

1. Traduction étendue d'un article publié en anglais à KDIR 2015 (Bouziri *et al.*, 2015)

## 1. Introduction

La Recherche d'Information (RI) étudie le processus d'adéquation entre la requête d'un utilisateur et une collection de documents, dont le résultat est souvent un sous-ensemble de documents pertinents contenant les mêmes termes de la requête originale. Le modèle classique de RI (Salton et McGill, 1983) consiste à attribuer, à chaque document d'une collection, des termes d'indexation, dits index du document, limitant les requêtes à l'ensemble global des termes de l'index, et utilisant des mesures de correspondance entre les requêtes et les documents. Une des difficultés rencontrées au cours d'une session de recherche est liée aux choix des termes d'interrogation. En effet, dans bien des cas, les termes de la requête, exprimée par l'utilisateur, ne correspondent pas exactement aux descripteurs des documents retenus par le modèle d'indexation. De ce fait, afin d'avoir des documents pertinents, l'utilisateur est contraint d'utiliser le "vocabulaire de description du document" propre au système. Face à cette contrainte, il est possible de faire appel à la technique d'expansion de requêtes (Buckley *et al.*, 1994) afin d'améliorer la correspondance requête/document, et ce en étendant la requête par des termes additionnels, corrélés à ceux de la requête originale. Intuitivement, l'apport d'une telle technique ne se réduit pas à l'amélioration du rappel en récupérant des documents pertinents qui ne peuvent pas être trouvés par la requête utilisateur, mais également à améliorer la précision des documents trouvés en les plaçant en haut de la liste des documents pertinents.

Nous proposons dans cet article une approche d'expansion de requêtes par apprentissage qui se base sur les règles d'association entre termes. Un processus exploratoire basé sur les algorithmes génétiques explore l'espace des règles d'association entre termes à la recherche des meilleurs termes d'expansion et génère en parallèle des instances d'entraînement qui serviront pour construire un classifieur. Intuitivement, une règle d'association traduit la probabilité d'avoir les termes de la conclusion dans un document, sachant que ceux de la prémisse y sont. Ainsi, l'utilisation de telles dépendances dans un processus d'expansion de requêtes améliore sensiblement la pertinence d'un SRI, car elles reflètent des corrélations fortes et implicites découvertes à partir de la collection de documents. Toutefois, face au nombre très important de règles d'association entre termes qui peuvent être découvertes à partir d'une collection de documents, nous proposons un nouveau processus d'expansion automatique de requêtes moyennant la base générique minimale *MGB* déployée dans (Latiri *et al.*, 2012).

## 2. Expansion de requêtes : revue de la littérature

La problématique d'expansion de requêtes a été largement abordée par la communauté RI depuis deux décennies (Buckley *et al.*, 1994 ; Ruthven et Lalmas, 2003 ; Kumaran et Allan, 2008 ; Carpineto et Romano, 2012).

Certaines approches d'expansion de requêtes se basent sur des techniques de fouille de texte et utilisent des connaissances dérivées à partir des collections de textes. Elles observent généralement la régularité des termes dans un contexte déterminé



d'une collection de textes. Elles sont basées sur l'hypothèse qui stipule que l'emploi de deux termes en co-occurrence est l'expression d'une relation sémantique entre eux (Rijsbergen, 1979). L'avantage de ces approches est qu'elles sont faciles à mettre en oeuvre tout en étant indépendantes du corpus. Parmi les premiers travaux relatifs à cette classe d'approches, Grefensette (Grefenstette, 1992) contribue par une approche syntaxique d'extraction de contextes de mots à partir des corpus textuels pour produire la liste des mots reliés à n'importe quel mot du corpus. D'autres approches s'appuient sur une analyse statistique des collections par l'extraction de règles d'association entre termes, afin d'ajouter des termes voisins à la requête originelle (Tangpong et Rungsawang, 2000 ; Haddad *et al.*, 2000). Les associations sont généralement basées sur la co-occurrence des termes dans les documents (Rungsawang *et al.*, 1999). L'usage d'une telle technique a montré que les liens inter-termes renforcent la notion de pertinence des documents par rapport aux requêtes (Latiri *et al.*, 2012).

D'autres travaux se sont intéressés à l'optimisation de l'expansion en utilisant des métaheuristiques tels que les algorithmes génétiques. Dans (Boughanem et Tamine, 2002), les auteurs présentent une approche d'expansion fondée sur une utilisation combinée de la stratégie d'injection de pertinence et des techniques avancées de l'algorithme génétique. La population est représentée par un ensemble de niches, renouvelée à chaque génération. La fonction d'adaptation est basée à la fois sur le jugement de pertinence des utilisateurs et sur un modèle de fonction statistiquement corrélé aux mesures de taux de rappel/précision. Les opérateurs génétiques sont appliqués de manière restreinte aux niches et non pas de manière uniforme sur toute la population. La sélection adoptée dans cette approche est fondée sur l'espérance mathématique qui permet de générer pour chaque individu de la population un nombre de clones dépendant de sa valeur d'adaptation. Le croisement est basé sur le poids des termes. La mutation consiste essentiellement à exploiter les termes présents dans les documents pertinents afin d'ajuster les valeurs des gènes correspondants dans les requêtes sélectionnées pour la mutation. Par ailleurs, les auteurs de (Araujo *et al.*, 2010) proposent une méthode basée sur les algorithmes génétiques pour résoudre les problèmes relatifs à la sélection de termes adéquats pour la reformulation de requêtes utilisant la structure morphologique des requêtes.

Certains travaux s'intéressent à développer des approches sélectives qui choisissent la fonction de recherche ou d'expansion en fonction des requêtes. La plupart des approches sélectives utilisent un processus d'apprentissage sur des caractéristiques de requêtes passées et sur les performances obtenues. Chifu et Mothe (Chifu et Mothe, 2014) présentent une méthode d'expansion sélective qui se base sur des prédicteurs de difficulté des requêtes. Le modèle de décision, appris par un SVM, permet de prédire la nature de l'expansion selon la difficulté de la requête.

Dans un couplage entre RI et fouille de textes, une approche d'expansion automatique de requêtes basée sur les règles d'association a été proposée dans (Latiri *et al.*, 2012). Cette approche commence par dériver, dans un premier temps, la base générique *MGB* de règles d'association non-redondantes entre termes à partir d'une collection de documents, et de l'utiliser, dans un deuxième temps, pour étendre la re-

quête originelle de l'utilisateur. Chaque requête originelle de la collection est ainsi étendue en injectant tous les termes qui apparaissent dans les conclusions des règles d'association, qui ont dans leurs prémisses respectives les termes de la requête originelle. Les expérimentations menées dans le cadre de l'évaluation de cette approche montrent, pour les différentes collections testées, une amélioration de la pertinence système en terme de MAP, réalisée avec les requêtes étendues par les règles d'association de la base générique *MGB* (Latiri *et al.*, 2012).

Dans le présent article, nous nous intéressons au raffinement de cette approche par une meilleure sélection des règles d'association candidates à l'expansion. En effet, nous partons de l'hypothèse que si une expansion de requêtes par injection systématique des termes des conclusion de toutes les règles d'association qui ont dans leurs prémisses respectives au moins un terme de la requête originelle, s'est avérée efficace ; alors elle serait encore plus efficace si seules les règles d'association qui améliorent la pertinence de la recherche sont considérées.

### 3. Définitions et formalisme de base

#### 3.1. Les algorithmes génétiques

Les algorithmes génétiques, initiés dans les années 1970 par John Holland, sont des algorithmes d'optimisation s'appuyant sur des techniques dérivées de la génétique et des mécanismes d'évolution de la nature : croisement, mutation, sélection. Un algorithme génétique recherche le ou les extrema d'une fonction définie sur un espace de données. Pour l'utiliser, on doit disposer des cinq éléments suivants :

- *Un principe de codage de l'élément de population.* Cette étape associe à chacun des points de l'espace d'état une structure de données. Elle se place généralement après une phase de modélisation mathématique du problème traité. La qualité du codage des données conditionne le succès des algorithmes génétiques. Les codages binaires ont été très utilisés à l'origine. Les codages réels sont désormais largement utilisés, notamment dans les domaines applicatifs pour l'optimisation de problèmes à variables réelles.

- *Un mécanisme de génération de la population initiale.* Ce mécanisme doit être capable de produire une population d'individus non homogène qui servira de base pour les générations futures. Le choix de la population initiale est important car il peut rendre plus ou moins rapide la convergence vers l'optimum global.

- *Une fonction à optimiser.* Celle-ci retourne une valeur appelée *fitness* ou fonction d'évaluation de l'individu.

- *Des opérateurs permettant de diversifier la population au cours des générations et d'explorer l'espace d'état.* L'opérateur de croisement recompose les gènes d'individus existant dans la population, l'opérateur de mutation a pour but de garantir l'exploration de l'espace d'états.

– *Des paramètres de dimensionnement* : taille de la population, nombre total de générations ou critère d'arrêt, probabilités d'application des opérateurs de croisement et de mutation.

### 3.2. Les règles d'association entre termes

Dans ce travail, nous nous basons sur une base générique de règles d'association entre termes appelée *MGB* proposée dans (Latiri *et al.*, 2012). La spécificité de cette base est qu'elle est compacte, dans le sens où elle englobe un noyau minimal de règles d'association approximatives et exactes non-redondantes entre termes. La caractéristique clé des règles dérivées est qu'elles ont des prémisses minimales illustrées par les générateurs minimaux et des conclusions maximales. De ce fait, les règles d'association s'avèrent intéressantes dans le contexte d'expansion de requêtes étant donné que ces prémisses maximales offrent plus de termes candidats à l'expansion. Leur déploiement en expansion de requêtes en RI a été également détaillé dans (Latiri *et al.*, 2012).

Une règle d'association entre termes  $R$  est une implication de la forme  $R : T_1 \Rightarrow T_2$ , où  $T_1$  et  $T_2$  sont deux sous-ensembles distincts de l'ensemble de tous les termes distincts des documents de la collection, et  $T_1 \cap T_2 = \emptyset$ .  $T_1$  et  $T_2$  sont, respectivement, appelés la *prémisse* et la *conclusion* de  $R$ . La règle  $R$  est ainsi dérivée à partir de  $T_1 \cup T_2$  (Latiri *et al.*, 2012).

Le *support* de la règle  $R : T_1 \Rightarrow T_2$  représente le nombre de documents  $d$  de la collection  $C$  qui contiennent tous les termes  $t$  de  $T_1 \cup T_2$ . Il est défini comme suit :

$$Supp(R) = Supp(T_1 \cup T_2) = |\{d | d \in C \wedge \forall t \in T_1 \cup T_2 : t \in d\}| \quad [1]$$

Alors que sa *confiance* est calculée comme suit :

$$Conf(R) = \frac{Supp(T)}{Supp(T_1)} = \frac{Supp(T_1 \cup T_2)}{Supp(T_1)} \quad [2]$$

Une règle d'association  $R$  est dite *valide* si sa valeur de confiance, *i.e.*,  $Conf(R)$ , est supérieure ou égale à un seuil prédéfini noté *minconf*. Ce seuil de confiance minimal est utilisé pour exclure les règles dites *non valides*. Par ailleurs, le seuil de support minimal *minsupp* est utilisé pour écarter les règles d'association dérivées à partir du termset  $T$  et qui ne sont pas suffisamment fréquentes, *i.e.*, les règles ayant un support  $Supp(T) < minsupp$ .

Dans la littérature, deux types de règles d'association sont définis à savoir : *les règles exactes* (avec une confiance égale à 1) et *les règles approximatives* (avec une confiance strictement inférieure à 1) (Zaki, 2004).

## 4. Apprentissage de l'expansion de requêtes par règles d'association

### 4.1. Aperçu général de l'approche

Notre système est formé de deux composants principaux comme le montre la Figure 1. Un premier composant est dédié à la construction des exemples d'entraînement. Le deuxième composant est chargé de la construction d'un modèle de prédiction à partir des exemples d'apprentissages et en utilisant un algorithme de classification supervisée. Etant donné une requête  $q$  et l'ensemble des règles d'association qui lui correspondent  $RA_q$ , le modèle de prédiction identifie  $RA_q^+$  le sous ensemble de règles d'association qui sont à utiliser pour générer la requête étendue  $Eq$ .

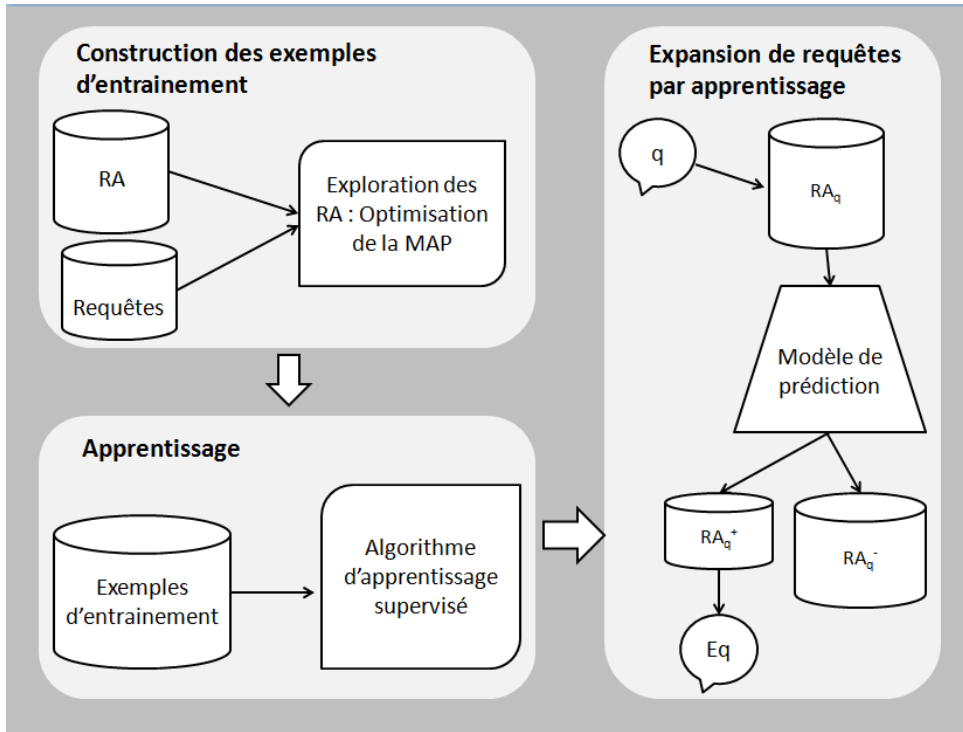


**Figure 1.** Schéma général de l'approche

### 4.2. Définition et modélisation du problème

Nous définissons le problème d'expansion de requêtes comme suit : étant donné une requête originelle  $q := \{t_1, t_2, \dots, t_n\}$ , et l'ensemble des règles d'association qui lui sont relatives  $AR_q$  ; il s'agit de trouver les règles d'association de  $AR_q^+$  dont les termes des conclusions sont les plus adaptés à enrichir  $q$  et restituer des documents qui répondent aux besoins de l'utilisateur.

Nous modélisons ce problème comme un problème de classification supervisée dans lequel il s'agit de prédire si une règle devrait être utilisée pour enrichir une re-

quête donnée. La classification consiste à définir une fonction  $f : \mathcal{X} \rightarrow \mathcal{Y}$  qui associe pour chaque observation  $x$  sa classe  $y$ .

Dans le contexte de l'expansion de requêtes utilisant les règles d'association :

- $\mathcal{X} \subseteq \mathbb{R}^d$  constitue l'ensemble des vecteurs à  $d$  dimension représentant l'espace des observations
- $x_{ij} \in \mathcal{X}$  est le vecteur des attributs d'une requête  $q_i$  et de la règle d'association  $AR_j$
- $\mathcal{Y} = \{0, 1\}$  est ensemble des classes possibles
- $y_{ij} \in \mathcal{Y}$  est la classe associée à  $x_{ij}$ .
  - $y_{ij} = 1$  si la règle d'association  $AR_j$  est sélectionnée par l'algorithme d'optimisation pour enrichir la requête  $q_i$
  - $y_{ij} = 0$  sinon.

Le vecteur d'attributs représenté par des attributs calculés à partir du texte de la requête et des termes de la conclusion de la règle d'association. Ces attributs sont au nombre de 10 et sont calculés à partir de mesures statistiques.

#### 4.2.1. Attributs basés sur la fréquence de documents

La fréquence de documents ( $DF$ ) d'un terme est une mesure statistique qui renseigne sur le nombre de documents qui contiennent le terme en question. Sa valeur pour une requête représente la moyenne du  $DF$  de tous les termes de la requête. La valeur  $DF(q)$  d'une requête  $q$  est calculée selon l'équation 3.

$$DF(q) = \frac{1}{|q|} \sum_{i=1}^{|q|} \frac{|\{d_j, t_i \in d_j\}|}{|C|} \quad [3]$$

Nous utilisons également

$$iDF(q) = \frac{1}{|q|} \sum_{i=1}^{|q|} \log \frac{|C|}{|\{d_j, t_i \in d_j\}|} \quad [4]$$

Pour une règle d'association, les  $DF$  et  $iDF$  sont calculés selon le même principe, et représentent la moyenne des  $DF$  (resp.  $iDF$ ) des termes de la conclusion de la règle.

#### 4.2.2. Attributs basés sur la fréquence des termes

Le deuxième type d'attributs que nous prenons en compte dans les instances d'entraînement concerne la fréquence d'un terme dans les documents de la collection. La fréquence ( $TF(t, d)$ ) correspond au nombre d'occurrences du terme  $t$  dans le document  $d$ . Pour chaque terme de la requête ou de la conclusion de la règle d'association, nous utilisons la moyenne des fréquences dans tous les documents et nous la notons

$ATF(t_i)$  pour le terme  $t_i$ . Pour une requête, la fréquence des termes est la moyenne des  $ATF$  de tous les termes de la requête et est calculée selon la formule 5.

$$TF(q) = \frac{1}{|q|} \sum_{i=1}^{|q|} ATF(t_i) \quad [5]$$

$$ATF(t_i) = \frac{1}{|C|} \sum_{j=1}^{|D|} TF(t, d_j)$$

Dans le calcul de la mesure  $TF(q)$ , nous nous basons sur une moyenne des fréquences de chaque terme dans tous les documents  $ATF(t)$ . Afin de tenir compte de la dispersion des différentes fréquences d'un terme dans chaque document par rapport à la moyenne  $ATF$ , nous introduisons la mesure  $VTF(q)$  qui représente la moyenne des variances des fréquences des termes de la requête. Le calcul de  $VTF(q)$  se fait comme l'indique la formule 6.

$$VTF(q) = \frac{1}{|q|} \sum_{i=1}^{|q|} \frac{1}{|C| - 1} \sum_{j=1}^{|C|} (TF(t_i, d_j) - ATF(t_i))^2 \quad [6]$$

#### 4.2.3. Attributs basés sur les caractéristiques des règles d'association

Nous utilisons 6 attributs dans le but de caractériser les règles d'association, à savoir :

- Le nombre de termes dans la conclusion de la règle,
- Le nombre de termes dans la prémisse de la règle,
- La confiance de la règle donnée dans l'équation (2),
- Le support de la règle donné dans l'équation (1).
- La proportion des termes de la requêtes qui se retrouvent dans la prémisse.
- Une mesure de l'importance des termes de la conclusion dans la base des règles d'association

### 4.3. Génération des exemples d'entraînement

Les exemples d'entraînement de la base d'apprentissage correspondent dans notre cas à des couples (requêtes, règles d'association). Pour chaque requête, les instances générées sont au nombre de règles d'association relatives à cette requête. Une instance est de classe positive si la règle d'association a été retenue pour l'expansion de la requête, elle est de classe négative sinon. Les règles d'association retenues pour l'expansion sont choisies lors d'un processus exploratoire qui se base sur un algorithme

génétique. Il prend en entrée une requête originelle et l'ensemble des règles d'association candidates à l'expansion pour cette requête. Il retourne une requête étendue optimisée et génère les instances d'apprentissage relatives à l'expansion optimale. Les principes de ce processus exploratoire sont décrits dans ce qui suit.

#### 4.3.1. Individu et Population

Un individu ou chromosome représente une requête étendue candidate.

Les gènes de l'individu sont les termes candidats à l'expansion issus à partir des règles d'association entre termes de la base générique  $\mathcal{MGB}$ . Pour ce faire, chaque terme de la requête originelle est traité individuellement en le cherchant dans les prémisses minimales des règles valides. Les termes des conclusions maximales de toutes les règles ayant comme prémisses le ou les termes de la requête originelle sont retenus comme candidats à l'expansion. La longueur d'un chromosome correspond au nombre de termes candidats à l'expansion.

Étant donnée une requête originelle  $Q = \{t_1, \dots, t_n\}$ , l'ensemble des termes candidats à l'expansion obtenu à partir des règles d'association de la base  $\mathcal{MGB}$ , noté  $TC$ , est exprimé comme suit (Latiri *et al.*, 2012) :

$$\begin{aligned} \forall R : T_1 \Rightarrow T_2, \text{une règle non-redondante} \in \mathcal{MGB}; & \quad [7] \\ \text{si } T_1 \subseteq Q & \\ \text{alors } TC = TC \cup T_2. & \end{aligned}$$

L'équation (7) signifie que si la prémisse de  $R$  est contenue dans  $Q$ , les termes de la conclusion sont candidats à l'expansion.

Nous adoptons un codage binaire qui renseigne sur l'apparition ou non d'un terme dans la requête étendue candidate.

#### 4.3.2. Population initiale

Chaque individu de la population initiale correspond à une requête étendue formée par les termes de la requête originelle auxquels sont ajoutés les termes des conclusions d'une règle d'association de la base  $\mathcal{MGB}$  dont les termes de la prémisse sont des termes de la requête originelle. Pour construire la population initiale, on commence par filtrer la base générique  $\mathcal{MGB}$  pour ne garder que les règles d'association dont les termes des prémisses sont des termes de la requête originelle.

Le nombre de ces règles détermine la taille de la population initiale. Le nombre de gènes des individus correspond au nombre de termes différents trouvés dans les conclusions de ces règles. Ces termes sont candidats à l'expansion.

#### 4.3.3. Fonction d'évaluation

L'objectif derrière l'expansion de requête est d'améliorer la pertinence de la tâche RI. La force d'un individu représente la capacité de la requête étendue correspondante à produire des documents pertinents. Nous utilisons la mesure d'évaluation des SRI, la MAP comme fonction d'évaluation. Pour ce faire, chaque individu/requête est soumis à un SRI expérimental (Terrier) pour en évaluer la force de l'individu.

#### 4.3.4. Opérateurs génétiques

##### 4.3.4.1. Sélection

Cet opérateur permet aux individus d'une population de survivre, de se reproduire ou de mourir. En règle générale, la probabilité de survie d'un individu sera directement reliée à son efficacité relative au sein de la population. Dans cette implémentation, nous optons pour une sélection élitiste qui consiste à sélectionner les  $n$  individus parents dont on a besoin pour la nouvelle génération en prenant les  $n$  meilleurs individus de la population courante après l'avoir triée de manière décroissante selon la valeur la fonction d'évaluation de ses individus.

##### 4.3.4.2. Croisement

Le rôle fondamental de l'opérateur de croisement est de permettre la recombinaison des informations présentes dans le patrimoine génétique de la population. L'opérateur de croisement favorise l'exploration de l'espace de recherche. Le croisement permet dans notre cas de produire de nouvelles requêtes étendues et enrichies avec de nouveaux termes à partir des requêtes/individus de la population courante. Nous optons pour un opérateur de croisement qui produit un seul enfant à partir de deux parents. La requête/ individu enfant hérite les termes d'expansion de ses deux parents.

##### 4.3.4.3. Remplacement

A chaque itération de l'AG, les nouveaux individus remplacent leur parents.

## 5. Évaluation expérimentale de l'approche d'expansion

### 5.1. Cadre d'évaluation

L'évaluation expérimentale a été menée avec le SRI TERRIER, en utilisant la collection de textes français SDA-95 de la campagne CLEF 2003. Quelques caractéristiques statistiques de la collection français *French SDA 95*, notée dans la suite de l'article SDA-95 sont données dans le Tableau 1.

Tous les champs de la requête, *i.e.*, titre, champs descriptifs et narratifs, sont utilisés lors du processus d'expansion. Nous avons testé notre approche d'expansion avec le schémas de pondération OKAPI BM25 qui est une méthode d'ordonnancement de la méthode OKAPI, la plus connue des méthodes probabilistes, et ayant pour but de



| Collection          | SDA95   |
|---------------------|---------|
| Nombre de documents | 42615   |
| Nombre de phrases   | 523320  |
| Nombre de termes    | 6664000 |
| Nombre de requêtes  | 60      |

**Tableau 1.** *Caractéristiques de la collection FRENCH SDA 95 (SDA-95) de CLEF 2003.*

construire un modèle probabiliste qui prend en compte la fréquence des termes ainsi que la taille des documents (Jones *et al.*, 2000).

## 5.2. Pré-traitements et bases comparatives

Nous avons procédé à un ensemble d'évaluations afin de déterminer des bases de comparaison pour évaluer la performance du processus d'expansion automatique de requêtes par apprentissage proposé dans cet article.

### 5.2.1. Évaluation des résultats pour les requêtes sans expansion

Il s'agit de déterminer la base d'évaluation comparative (baseline). La pertinence des documents retournés est estimée selon les mesures d'évaluation suivantes :

- Précision de la requête originelle (Q) à 11 points de rappel (P@11).
- Les précisions à P@5, P@10, P@15, et P@30 documents pertinents restitués.
- La précision moyenne MAP (Mean Average Precision). Comme la courbe de la précision à 11 points de rappel, la MAP définit la performance globale d'un SRI.

### 5.2.2. Évaluation des résultats de recherche pour les requêtes étendues avec PRF

La pseudo injection de pertinence (ou PRF de l'anglais Pseudo Relevance Feed Back) est une méthode d'expansion complètement automatique largement utilisée en recherche d'information. Elle considère les premiers documents retrouvés via la requête initiale comme pertinents et utilise les informations issues de ces documents pour l'expansion. Nous avons utilisé la méthode PRF implémentée dans Terrier avec les paramètres par défaut (T=10, D=3) pour établir une comparaison avec une méthode d'expansion basique en RI.

### 5.2.3. Génération de la base générique des règles d'association $\mathcal{MGB}$

Lors de la génération de la base générique  $\mathcal{MGB}$  (Latiri *et al.*, 2012), nous avons fait varier les seuils minimaux de support et de confiance, *i.e.*,  $\minsupp$  et  $\minconf$ . Rappelons que ces seuils sont définis pour éliminer, respectivement, les règles très rares et celles qui ne sont pas valides. Nous avons choisi de générer une base géné-

rique  $\mathcal{MGB}$  avec un large nombre de règles d'association entre termes afin de diversifier les termes candidats à l'expansion et de garantir plus d'efficacité au processus d'expansion. Pour ce faire nous avons opté, moyennant une étude de la distribution de Zipf des termes du vocabulaire, pour une valeur de  $minsupp=0.003$  et pour une valeur de  $minconf=0.1$ .

#### 5.2.4. Expansion de requêtes en utilisant $\mathcal{MGB}$

Afin de montrer l'intérêt de notre approche, nous avons procédé à une expansion "aveugle" des requêtes en utilisant la base générique des règles d'association  $\mathcal{MGB}$ . Dans cette expansion, nous ajoutons les termes des conclusions de toutes les règles de  $\mathcal{MGB}$  ayant au moins un terme de la prémisse dans la requête.

#### 5.2.5. Génération d'exemples d'entraînement

Le processus d'optimisation de l'expansion de requêtes basé sur les algorithmes génétique (c.f, section 4.3) est utilisé pour générer à la fois les requêtes étendues optimisées et les instances d'entraînement correspondantes aux 60 requêtes de la collection.

Il est important de noter que le nombre total d'instances générées pour les 60 requêtes est de 14495 avec 14058 instances négatives contre seulement 437 instances positives.

### 5.3. Protocole expérimental

Nous avons procédé à une validation croisée à 4 échantillons de test. Les instances d'entraînement correspondants aux 60 requêtes de la collection sont divisées en 4 sous-ensembles, chacun comportant les instances relatives à 15 requêtes. Le processus d'expérimentation se fait en quatre itérations, à chaque fois 3 sous-ensembles d'instances sont considérées pour construire le classifieur et le quatrième est utilisé pour la validation du modèle. Nous faisons de sorte que les 4 sous-ensembles soient testés. Les résultats de classification obtenus dans les quatre phases de test sont fusionnés en un ensemble de 60 requêtes étendues par apprentissage qui est soumis à une évaluation de la recherche.

Le pourcentage de classification correcte (PCC) est une mesure de performance largement utilisée dans la littérature pour l'évaluation des classifieurs. Dans certains cas, cette mesure s'avère peu ou pas du tout significative. Cette inefficacité s'exprime, surtout, dans le cas où il existe un déséquilibre important dans la distribution des classes comme c'est le cas dans nos instances d'entraînement. En effet, un classifieur qui affecte systématiquement les instances à la classe la plus représentée dans la base, aura un PCC correspondant à la proportion de cette classe qui dépasse les 90%. Par conséquent, ce classifieur pourra être considéré comme étant un excellent classifieur si la majorité des instances appartiennent à la classe en question. Ceci est faux, car, réellement, ce classifieur ne dispose d'aucun pouvoir prédictif. Nous avons eu recours

**Tableau 2.** *Performance des classifieurs J48*

| Itération | PCC    | PVP     | PVN    |
|-----------|--------|---------|--------|
| 1         | 96,00% | 11,5%   | 98,4%  |
| 2         | 97,08% | 7,4,52% | 99,7%  |
| 3         | 97,09% | 7,7%    | 99,7%  |
| 4         | 95,32% | 8,0%    | 98,9%  |
| Moyenne   | 96,37% | 8,65%   | 99,17% |

**Tableau 3.** *Performance des classifieurs Random Forest*

| Itération | PCC    | PVP      | PVN    |
|-----------|--------|----------|--------|
| 1         | 97,03% | 7,07%    | 99,55% |
| 2         | 97,08% | 4,62,52% | 99,75% |
| 3         | 97,17% | 1,92%    | 99,94% |
| 4         | 95,88% | 5,35%    | 99,56% |
| Moyenne   | 96,79% | 4,74%    | 99,7%  |

à des mesures plus précises qui représentent les proportions des instances correctement prédites parmi les instances positives (PVP) et négatives (PVN). Un classifieur sera d'autant plus performant qu'il sera capable de concilier un PVP et un PVN élevés.

#### 5.3.1. *Classifieurs obtenus par la méthode de l'arbre de décision*

A chaque itération de la validation croisée, la méthode J48 de Weka qui implémente la méthode de l'arbre de décision C4.5 (Quinlan, 1987) est utilisée pour générer un classifieur. Les valeurs des paramètres retenus sont ceux par défaut. Les performances des quatre classifieurs obtenus sont résumées dans la Table 2

#### 5.3.2. *Classifieurs obtenus par la méthode Random Forest*

De même que pour la méthode de l'arbre de décision, à chaque itération de la validation croisée un classifieur est construit par la méthode *Random Forest* de Weka. Les performances des quatre classifieurs obtenus sont résumées dans la Table 3

### 5.4. *Résultats expérimentaux et discussion*

Nous avons utilisé les mesures de précision P@5, P@10, P@20 et P@30 documents qui sont respectivement, la précision moyenne aux 5, 10, 20 et 30 premiers documents retournés et la MAP (Mean Average Précision), sur l'ensemble des 60 requêtes. Pour chaque requête, les 1 000 premiers documents sont renvoyés par le SRI expérimental et les précisions moyennes sont calculées pour mesurer la pertinence

**Tableau 4.** *Résultats expérimentaux pour la collection SDA-95*

|                 | MAP    | P@5     | P@10  | P@20   | P@30   |
|-----------------|--------|---------|-------|--------|--------|
| <b>Baseline</b> | 0,355  | 0,380   | 0,280 | 0,216  | 0,176  |
| BE              | 0,282  | 0,286   | 0,221 | 0,153  | 0,128  |
| $\Delta$ BE     | -20,5% | -24,7%  | -21%  | -29,1% | -27,2% |
| PRF             | 0,358  | 0,392   | 0,296 | 0,212  | 0,180  |
| $\Delta$ PRF    | 0,8%   | 3,2%    | 5,7%  | -1,9%  | 2,3%   |
| AG              | 0,432  | 0,453   | 0,355 | 0,253  | 0,198  |
| $\Delta$ AG     | 21,7%  | 19,2%   | 26,8% | 17%    | 12,5%  |
| DT              | 0,358  | 0,376   | 0,278 | 0,215  | 0,175  |
| $\Delta$ DT     | 0,8%   | -0,011% | -0,7% | -0,5%  | -0,6%  |
| RF              | 0,358  | 0,380   | 0,282 | 0,216  | 0,176  |
| $\Delta$ RF     | 0,8%   | 0,00%   | 0,7%  | 0%     | 0%     |

système. Nous avons aussi calculé la variation de la pertinence système, notée  $\Delta$ , est calculée comme suit :

$$\Delta = \frac{(Pertinence\ avec\ expansion) - Baseline}{Baseline}$$

La Table 4 synthétise les résultats obtenus pour la collection SDA95. Les valeurs dans la ligne *BE* indiquent les résultats obtenus avec les requêtes étendue par la méthode aveugle. La ligne *PRF* reporte les résultats obtenus avec les requêtes étendues en utilisant la méthode de pseudo relevance feed back. Les valeurs dans la ligne *AG* sont obtenues pour les requêtes étendues en utilisant l'algorithme d'exploration basé sur les algorithmes génétiques. Les valeurs de la ligne *DT* correspondent aux expérimentations sur les requêtes étendues par apprentissage en appliquant le classifieur *J48*. Les valeurs de la ligne *RF* présentent les résultats obtenus pour les requêtes étendues par apprentissage en appliquant le classifieur *RandomForest*. Nous indiquons pour chacune de ces lignes les variations par rapport à la baseline.

Ces résultats montrent que les MAP obtenus avec les requêtes optimisées par les AG sont nettement meilleure en terme de MAP et de précision. En effet, ces résultats représentent une borne supérieur à atteindre idéalement avec l'expansion par apprentissage puisque l'expansion par AG explore l'espace des règles d'association tout en maximisant la MAP. L'expansion de requêtes par apprentissage apporte une amélioration de 0,8% des résultats de la recherche en terme de MAP par rapport à la *baseline*. Nous constatons d'autre part que notre approche est aussi performante avec PRF. L'expansion "aveugle" se basant sur les règles d'association s'est avérée inefficace pour la collection testée ce qui montre l'intérêt de l'approche de sélection des règles d'association que nous proposons dans ce travail. Ceci peut s'expliquer par la nature des instances d'apprentissages qui présentent un déséquilibre considérable entre le nombre d'instances négatives et positives. En effet, la classe positive est minoritaire (3% du

nombre total d'instances), ce qui ne permet pas au classifieur de mettre en évidence des plages de valeurs des attributs qui soit discriminantes pour cette classe comme le montrent les faibles valeurs des PVP dans les tables 2 et 3. Des techniques de sur-échantillonnage et sous-échantillonnage ont été utilisées pour remédier à ce problème de déséquilibre d'instances mais aucune de ces technique n'a permis d'améliorer les taux des PVP ni des résultats de la recherche.

## 6. Conclusion et travaux en cours

Nous avons présenté dans cet article, une approche d'expansion de requêtes par apprentissage qui se base sur les règles d'association entre termes. Le problème d'expansion est modélisé en tant qu'un problème de classification supervisée. Un processus exploratoire basé sur les algorithmes génétiques explore l'espace des règles d'association à la recherche des meilleurs termes d'expansion et génère en parallèle des instances d'entraînement qui sont utilisés par la suite pour construire un classifieur. La résolution du problème d'apprentissage se fait par arbre de décision et par la méthode *RandomForest*. Les expérimentations menées sur la collection de textes français SDA-95 montrent que les requêtes étendues par apprentissage pour les deux méthodes permettent d'améliorer les résultats de la recherche par rapport aux requêtes originelles et restent au même niveau que les résultats de PRF. Nous travaillons à présent sur une nouvelle modélisation du problème qui permettra de classer les règles d'association relatives à une requête selon leur capacité de fournir une expansion adéquate de la requête.

## 7. Remerciements

Ce travail est partiellement financé par le projet de collaboration Franco-Tunisien DGRST-CNRS :14/R 1401.

## 8. Bibliographie

- Araujo L., Zaragoza H., Pérez-Agüera J. R., Pérez-Iglesias J., « Structure of morphologically expanded queries : A genetic algorithm approach », *Data Knowl. Eng.*, vol. 69, n<sup>o</sup> 3, p. 279-289, 2010.
- Boughanem M., Tamine L., « A Study on Using Genetic Niching for Query Optimisation in Document Retrieval », *Advances in Information Retrieval, 24th BCS-IRSG European Colloquium on IR Research Glasgow, UK, March 25-27, 2002 Proceedings*, p. 135-149, 2002.
- Bouziri A., Latiri C., Gaussier É., Belhareth Y., « Learning Query Expansion from Association Rules Between Terms », *KDIR 2015 - Proceedings of the International Conference on Knowledge Discovery and Information Retrieval, part of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2015), Volume 1, Lisbon, Portugal, November 12-14, 2015*, p. 525-530, 2015.

- Buckley C., Salton G., Allan J., Singhal A., « Automatic Query Expansion Using SMART : TREC-3 », *Proceedings of the 3<sup>rd</sup> Text REtrieval Conference*, 1994.
- Carpineto C., Romano G., « A survey of automatic query expansion in information retrieval », *ACM Computing Surveys (CSUR)*, vol. 44, n° 1, p. 1, 2012.
- Chifu A., Mothe J., « Expansion sélective de requêtes par apprentissage », in M. Moens, C. Viard-Gaudin, H. Zargayouna, O. R. Terrades (eds), *CORIA 2014 - Conférence en Recherche d'Informations et Applications- 11th French Information Retrieval Conference. CI-FED 2014 Colloque International Francophone sur l'Ecrit et le Document, Nancy, France, March 19-23, 2014.*, ARIA-GRCE, p. 257-272, 2014.
- Grefenstette G., « Use of semantic context to produce term association lists for text retrieval », *Proceedings of the 15<sup>th</sup> Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'92*, ACM Press, Copenhagen, Denmark, p. 89-97, June, 1992.
- Haddad H., Chevallet J. P., Bruandet M. F., « Relations between Terms Discovered by Association Rules », *Proceedings of the Workshop on Machine Learning and Textual Information Access in conjunction with the 4<sup>th</sup> European Conference on Principles and Practices of Knowledge Discovery in Databases, PKDD 2000*, Lyon, France, September, 2000.
- Jones K. S., Walker S., Robertson S. E., « A probabilistic model of information retrieval : development and comparative experiments », *Information Processing and Management*, vol. 36, n° 6, p. 779-840, 2000.
- Kumaran G., Allan J., « Adapting information retrieval systems to user queries », *Information Processing and Management*, vol. 44, n° 6, p. 1838-1862, 2008.
- Latiri C., Haddad H., Hamrouni T., « Towards An Effective Automatic Query Expansion Process Using An Association Rule Mining Approach », *Journal of Intelligent Information Systems*, vol. 39, n° 1, p. 209-247, 2012.
- Quinlan J., « Simplifying decision trees », *International Journal of Man Machine Studies*, vol. , n° 27, p. 221-234, 1987.
- Rijsbergen C. V., *Information Retrieval*, Butterworths, London, 1979.
- Rungsawang A., Tangpong A., Laohawee P., Khampachua T., « Novel Query Expansion Technique Using Apriori Algorithm », *Proceedings of the 8<sup>th</sup> Text REtrieval Conference, TREC 8*, Gaithersburg, Maryland, p. 453-456, November, 1999.
- Ruthven I., Lalmas M., « A survey on the use of relevance feedback for information access systems », *Knowledge Engineering Review*, vol. 18, n° 2, p. 95-145, 2003.
- Salton G., McGill M. J., *Introduction to Modern Information Retrieval*, McGraw-Hill, 1983.
- Tangpong A., Rungsawang A., « Applying Association Rules Discovery in Query Expansion Process », *Proceedings of the 4<sup>th</sup> World Multi-Conference on Systemics, Cybernetics and Informatics, SCI 2000*, Orlando, Florida, USA, July, 2000.
- Zaki M. J., « Mining Non-Redundant Association Rules », *Data Mining and Knowledge Discovery*, vol. 9(3), p. 223-248, 2004.



# CORIA – CIFED 2016

## Rencontres Jeunes Chercheurs

Présidentes :

*Sabine Barrat,*  
*Lorraine Gueuriot*





---

# Tied Spatial Transformer Networks for Character Recognition

**Bogdan-Ionuț CIRSTEA, Laurence LIKFORMAN-SULEM, Institut Mines-Télécom / Télécom ParisTech, Université Paris-Saclay, Paris, France**

---

*ABSTRACT. This paper reports a new approach applied to convolutional neural networks (CNNs), which uses spatial transformer networks (STNs). It consists in training an architecture which combines a localization CNN and a classification CNN, for which most of the weights are tied, which from here on we will name Tied Spatial Transformer Networks (TSTNs). The localization CNN is used for predicting the best affine transform for the input image, which is then processed according to the predicted parameters and passed through the classification CNN. We have conducted initial experiments on the cluttered MNIST dataset, comparing the TSTN and Spatial Transformer Networks (STN) with untied weights, as well as the classification CNN only. In all these cases, we obtain better results using the TSTN architecture.*

*RÉSUMÉ. Cet article présente une nouvelle approche appliquée aux réseaux de neurones convolutionnels (RNC), qui utilise les réseaux de transformations spatiales (RTS). L'approche consiste à construire une architecture combinant un RNC pour la localisation et un RNC pour la classification. Bien que les deux réseaux soient dédiés à des tâches différentes, la majorité de leurs poids sont partagées. Par la suite nous appelons ce type de réseaux réseaux de transformations spatiales liées ou RTSL. Le RNC de localisation est utilisé pour prédire la meilleure transformée affine que l'on peut appliquer sur l'image d'entrée, qui est ensuite traitée selon les paramètres prédits et passée au RNC de classification, qui fournit la prédiction. Nous avons mené des expériences initiales sur l'ensemble de données cluttered-MNIST comprenant des bruits additionnels. Nous comparons le RTSL et un RTS avec poids non liés, ainsi qu'avec le RNC de classification seul. Dans tous les cas, de meilleurs résultats sont obtenus en utilisant l'architecture RTSL proposée.*

**KEYWORDS:** Convolutional Neural Network, Deep Learning, Character Recognition,

**MOTS-CLÉS :** Réseau de neurones convolutionnel, Apprentissage Profond, Reconnaissance de caractères,

---

## 1. Introduction

Recently, deep architectures based on convolutional neural networks (CNNs) have obtained state-of-the-art results for several recognition tasks in computer vision, such as handwriting recognition (Graves and Schmidhuber, 2009) (Bluche *et al.*, 2015) and object recognition (Krizhevsky *et al.*, 2012) (Simonyan and Zisserman, 2014) (Russakovsky *et al.*, 2015). In contrast to classical approaches, deep learning systems learn features in an automatic fashion, avoiding the time-consuming task of hand-crafting them.

In this article we use a variation on STNs (Spatial Transformer Networks), deep learning architectures that couple two CNNs using a Spatial Transformer module, which was first introduced by (Jaderberg *et al.*, 2015). The first one, the so-called *localization* CNN, is used to perform a spatial transformation on the input image. This spatially transformed image is then used by the second CNN, the *classification* CNN, to perform the class prediction.

The advantage of using such coupled architectures (transformation/localization coupled to perform classification) is that the weights of the localization network are learned during the training phase, along with those of the classification network, so that images (of characters in our case) can be transformed and classified in an unified way. Since localization and classification are different tasks, the weights of the localization and classification networks generally differ. We propose here to tie the weights of both networks in order to obtain a regularization effect which can improve performance. We apply our approach to the recognition of noisy characters, using the MNIST-cluttered database, derived from the MNIST database but including additional noises.

## 2. Common elements

In this section, we present the common elements used between the classification CNN, the untied STN, and the TSTN. For simplicity and to make the comparison of the results easier, we have tried to keep as many elements as possible (convolutional architectures, activation function, optimization procedure, regularization) identical or as similar as possible between the three neural network architectures.

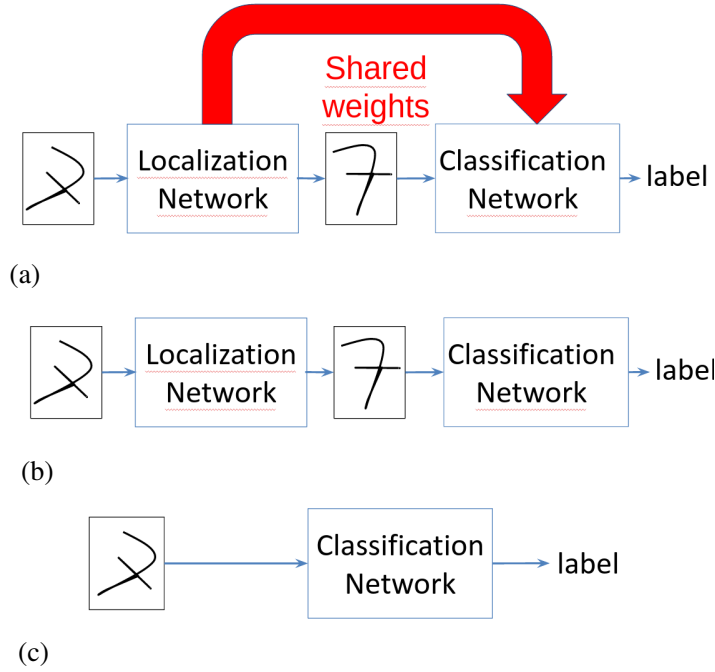


Figure 1 – TSTN, untied STN and CNN architectures

## 2.1. Convolutional architectures

We perform experiments using two different architectures for the classification and localization CNNs, the second one being more complex (and more expressive) than the first. These architectures follow many of the guidelines from (Simonyan and Zisserman, 2014), the system which has obtained the best single-model performance in the ILSVRC 2014 object classification competition (Russakovsky *et al.*, 2015). Each of the convolutional layers contains filters of size 3x3. 'Same' mode convolution (which allows the size of the output of each convolutional map to be equal to the input) is used in all the layers. The subsampling layers are always 2x2 max-pooling with stride 2, which results in subsampling both image height and width by 2. We double the number of convolutional maps between each consecutive convolutional layer, from 32 in the first convolutional layer to 64 then 128 in the last one, drawing once more inspiration from the guidelines outlined in (Simonyan and Zisserman, 2014).

The fully connected parts are identical between the two CNN architectures described in Tables 1 and 2, but different between the localization and classification CNNs. The convolutional parts are also very similar; the difference is that for Architecture 2 we use 2 operations of convolution followed by nonlinearity instead of a single one, followed by the pooling layer, for each of the 3 convolutional layers, making Architecture 2 more powerful and expressive. In both architectures, we remove the last pooling layer of the localization CNN. The motivation for this architectural

choice is to keep more of the spatial information that the fully connected layer of the localization CNN might need in order to output the estimated parameters of the affine transformation. Notice also that since the max-pooling layers don't contain any trainable parameters, this means that the classification and localization CNNs still have the same (tied) weights. For both architectures, the last layer of the localization CNN is smaller than that of the classification CNN, imitating the design of the architectures used in (Sønderby *et al.*, 2015) and (Sønderby, 2015b).

STNs can be useful for character recognition because they are able to disentangle the characters' pose and deformation from their shape. In noisy images such as those of the cluttered MNIST dataset, STNs can also crop out and scale-normalize the appropriate image region for the classification task. The Spatial Transformer module is differentiable, so it can be easily inserted into existing convolutional architectures without having to change the optimization algorithm. It can also be interpreted as a differentiable attention mechanism, whose output is conditioned on individual data samples (input images). The appropriate output for each data sample (i.e. its class) is learned during the training / optimization phase; when integrated into CNN architectures used for recognition, the Spatial Transformer requires no extra supervision signal.

Generally, the transformation performed on the input image using the Spatial Transformer can include scaling, cropping, rotations, as well as non-rigid deformations (Jaderberg *et al.*, 2015). In all of our architectures, we only use the affine transform Spatial Transformer, introduced in (Jaderberg *et al.*, 2015) and also used in (Sønderby *et al.*, 2015). It allows cropping, translation, rotation, scale, and skew to be applied to the input image and only requires 6 parameters to be estimated by the localization network. The implementation we use, introduced and described in (Sønderby *et al.*, 2015), also allows the use of a subsampling factor  $d$ , by which both the height and the width of the image are scaled, after applying the affine transform. To make the models simpler and help simplify comparisons between different models (including between CNNs and STNs), we use  $d = 1$  in all of our experiments involving STNs. By performing affine transforms, architectures integrating the Spatial Transformer module can both select relevant portions of an image (a form of attention) and transform those regions into a pose which can simplify the recognition task for the following layers of the architecture. For all the technical details about Spatial Transformers, we refer the reader to the relevant publication (Jaderberg *et al.*, 2015).

## 2.2. Activation functions and parameter initialization

As activation function, we use in all of our experiments the Rectified linear unit (ReLU) (Glorot *et al.*, 2011), one of the most successful activation functions in deep learning.

For simplicity, we initialize all the weight matrices using a method similar to the one proposed in (He *et al.*, 2015) and which is also used in (Sønderby, 2015b). Weight

matrices are initialized using values drawn from uniform distributions with mean 0 and standard deviation  $\sqrt{\frac{2}{n_i}}$ , where  $n_i$  is the number of inputs to the neuron, while the biases are initialized to 0. The procedure, presented in detail in (He *et al.*, 2015), helps avoid instability in gradients during the training phase, as well as instability of inputs to each layer during inference. The affine transform layer parameters are initialized to the identity transform, like in (Sønderby, 2015b). We refer you to (He *et al.*, 2015) for the full mathematical treatment.

### 2.3. Loss function and optimization

The loss function we minimize is the negative log-likelihood of the conditional distribution of the correct label given the input. Each one of the systems we experiment with represents a differentiable function. To optimize each of them, we can use stochastic gradient descent (SGD), applied to neural networks using the backpropagation algorithm on mini-batches of multiple examples (computation over mini-batches is much more efficient than over single samples when using GPUs).

We use as optimization algorithm a variant of Adam (Kingma and Ba, 2014), which uses the gradients computed through backpropagation. Adam is a state of the art algorithm developed for improving the performance of stochastic gradient descent (SGD) on stochastic non-stationary loss functions and sparse gradients. It introduces a few additional hyper-parameters, all described in detail in (Kingma and Ba, 2014); we have kept all these hyper-parameters fixed, using the values recommended in (Kingma and Ba, 2014). The only parameter we vary is the learning rate, which we multiply by a fixed amount after a fixed number of epochs. We have used the same values as in (Sønderby, 2015b) and multiplied the learning rate by 0.7 after each 20 epochs.

### 2.4. Regularization

Dropout (Srivastava *et al.*, 2014) is a regularization method which allows for much bigger networks to be trained without overfitting, by randomly dropping units from the neural network during training with probability  $p$  sampled from a Bernoulli distribution. During validation and testing, the units are no longer dropped, but their activations are scaled deterministically by the same probability  $p$ . We use  $p = 0.5$  dropout in the fully connected layer of the classification CNN only, because it is the layer with the largest number of parameters, where regularization is potentially most useful. We don't use dropout in the fully connected layer of the localization network. The intuition behind this choice is that dropout, as a noisy form of regularization, tends to remove some of the spatial information present in the activation maps. This spatial information might be more valuable for the localization task (as compared to the classification one), so we prefer to preserve more of it (at the risk of overfitting) by not using dropout in the fully connected layer of the localization network.

| <b>Localization CNN</b>                      | <b>Classification CNN</b>                    |
|----------------------------------------------|----------------------------------------------|
| <b>Convolutional Layer 1</b>                 | <b>Convolutional Layer 1</b>                 |
| conv: 3 x 3 same, stride 1, 32 feature maps  | conv: 3 x 3 same, stride 1, 32 feature maps  |
| ReLU                                         | ReLU                                         |
| 2 x 2 max pooling, stride 2                  | 2 x 2 max pooling, stride 2                  |
| <b>Convolutional Layer 2</b>                 | <b>Convolutional Layer 2</b>                 |
| conv: 3 x 3 same, stride 1, 64 feature maps  | conv: 3 x 3 same, stride 1, 64 feature maps  |
| ReLU                                         | ReLU                                         |
| 2 x 2 max pooling, stride 2                  | 2 x 2 max pooling, stride 2                  |
| <b>Convolutional Layer 3</b>                 | <b>Convolutional Layer 3</b>                 |
| conv: 3 x 3 same, stride 1, 128 feature maps | conv: 3 x 3 same, stride 1, 128 feature maps |
| ReLU                                         | ReLU                                         |
|                                              | 2 x 2 max pooling, stride 2                  |
| flatten                                      | flatten                                      |
| <b>Fully Connected Layer</b>                 | <b>Fully Connected Layer</b>                 |
| 200 neurons linear layer                     | 625 neurons linear layer                     |
| ReLU                                         | ReLU                                         |
|                                              | 0.5 dropout                                  |
| <b>Affine Transform Layer</b>                | <b>Softmax Layer</b>                         |
| 6 parameters                                 | 10-class softmax                             |

Table 1 – Architecture 1: localization CNN (left) and classification CNN (right).

| <b>Localization CNN</b>                      | <b>Classification CNN</b>                    |
|----------------------------------------------|----------------------------------------------|
| <b>Convolutional Layer 1</b>                 | <b>Convolutional Layer 1</b>                 |
| conv: 3 x 3 same, stride 1, 32 feature maps  | conv: 3 x 3 same, stride 1, 32 feature maps  |
| ReLU                                         | ReLU                                         |
| conv: 3 x 3 same, stride 1, 32 feature maps  | conv: 3 x 3 same, stride 1, 32 feature maps  |
| ReLU                                         | ReLU                                         |
| 2 x 2 max pooling, stride 2                  | 2 x 2 max pooling, stride 2                  |
| <b>Convolutional Layer 2</b>                 | <b>Convolutional Layer 2</b>                 |
| conv: 3 x 3 same, stride 1, 64 feature maps  | conv: 3 x 3 same, stride 1, 64 feature maps  |
| ReLU                                         | ReLU                                         |
| conv: 3 x 3 same, stride 1, 64 feature maps  | conv: 3 x 3 same, stride 1, 64 feature maps  |
| ReLU                                         | ReLU                                         |
| 2 x 2 max pooling, stride 2                  | 2 x 2 max pooling, stride 2                  |
| <b>Convolutional Layer 3</b>                 | <b>Convolutional Layer 3</b>                 |
| conv: 3 x 3 same, stride 1, 128 feature maps | conv: 3 x 3 same, stride 1, 128 feature maps |
| ReLU                                         | ReLU                                         |
| conv: 3 x 3 same, stride 1, 128 feature maps | conv: 3 x 3 same, stride 1, 128 feature maps |
| ReLU                                         | ReLU                                         |
|                                              | 2 x 2 max pooling, stride 2                  |
| flatten                                      | flatten                                      |
| <b>Fully Connected Layer</b>                 | <b>Fully Connected Layer</b>                 |
| 200 neurons linear layer                     | 625 neurons linear layer                     |
| ReLU                                         | ReLU                                         |
|                                              | 0.5 dropout                                  |
| <b>Affine Transform Layer</b>                | <b>Softmax Layer</b>                         |
| 6 parameters                                 | 10-class softmax                             |

Table 2 – Architecture 2: localization CNN (left) and classification CNN

### 3. Experiments

As initial experiments, we apply TSTNs to the recognition of noisy handwritten digits. We conduct recognition experiments on the cluttered MNIST database (Sønderby, 2015a) which is derived from the MNIST database (Yann LeCun, 2015) but includes additional noise and transformations, which makes it harder to classify: the digits are distorted using random translation, scale, rotation, and clutter. We show some images samples from cluttered MNIST in Figure 2.

For our implementation, we use Theano (Bergstra *et al.*, 2010) (Bastien *et al.*, 2012) and Lasagne (Dieleman *et al.*, 2015) and rely heavily on the following implementation (Sønderby, 2015c) and example (Sønderby, 2015b) of Spatial Transformer Networks.

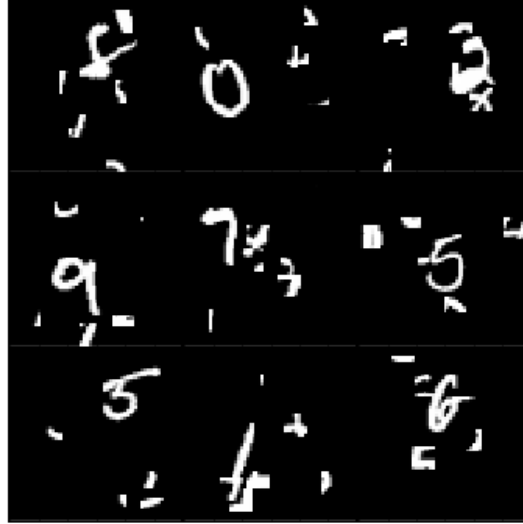


Figure 2 – Cluttered MNIST dataset image samples

#### 3.1. CNN and STN comparison

We compare in Table 3 the proposed TSTN network (tied weights) against the untied STN and the classification CNN alone, using two different architectures (architecture 2 being more complex) on the cluttered MNIST dataset. We list here the tested networks:

- 1) Classification CNN only, Architecture 1
- 2) Untied STN, Architecture 1
- 3) TSTN, Architecture 1
- 4) Classification CNN only, Architecture 2

5) Untied STN, Architecture 2

6) TSTN, Architecture 2

We have stopped the training of each one of the networks manually, after checking that the training accuracy is (almost) perfect. In a single case, the untied STN architecture 2, we haven't been able to get the training procedure to converge. To highlight that the number of epochs of training is comparable and that we have allowed each network a reasonable amount of training to achieve good performance, we also show in Table 3 the number of training epochs and best training accuracy (achieved up to the maximum training epoch). For each architecture and training procedure, we choose the parameters for which the best validation accuracy is obtained for testing. The results are shown in the last column of Table 3.

As can be seen in the results table, the TSTN obtains significantly better performance than both the classification CNN and the untied STN, for both architectures 1 and 2.

| Model                                   | Maximum training epochs | Best training error (%) | Validation error (%) | Test error (%) |
|-----------------------------------------|-------------------------|-------------------------|----------------------|----------------|
| Architecture 1, classification CNN only | 145                     | 0                       | 4.39                 | 4.44           |
| Architecture 1, untied STN              | 179                     | 0                       | 4.38                 | 4.33           |
| Architecture 1, TSTN                    | 196                     | 0.002                   | 3.02                 | 3.14           |
| Architecture 2, classification CNN only | 95                      | 0                       | 1.82                 | 2.18           |
| Architecture 2, TSTN                    | 231                     | 0                       | 1.3                  | 1.74           |

Table 3 – Results on the cluttered MNIST database for different architectures and training schemes.

#### 4. Discussion

In this section we provide a short possible interpretation of the preliminary results of our experiments with TSTNs.

First, the parameters learned by the localization CNN can be interpreted as encoding the knowledge of how to transform training samples (the transform being conditioned on each individual sample) so as to simplify the task of recognition for the following layers of the entire network. By tying the parameters of the classification and localization CNNs, the TSTN is constrained to become good at both tasks (of transforming the input image and classifying the transformed image). From this perspective, tying the parameters of the classification and localization CNNs can be interpreted as a form of regularization. We have performed additional preliminary



experiments which would seem to support this hypothesis; those experiments will be presented in a future publication.

An advantage of tying parameters in the TSTNs is that we no longer have to do a separate hyper-parameter search for the localization CNN. A disadvantage of STNs (both untied and tied), however, is that the computational time needed to use them is approximately twice the computation cost of the classification CNN alone. From this perspective, using training procedures where we combine training the TSTN and classification CNN alternately and only use the classification CNN at test time is interesting. This can be interpreted as a curriculum form of learning and is one of the experiments we are currently pursuing to validate our hypothesis of the tying the parameters of the classification and localization CNNs as a form of regularization.

## 5. Conclusion

In this article we have proposed a new way of using Spatial Transformer Network architectures, by tying the parameters between the localization and classification networks, which we call Tied Spatial Transformer Networks (TSTNs). A first extension of our work should be performing more experiments and analyzing the representations learned by the TSTN.

For improved results, we could keep the tied weights between the localization and classification CNNs in the TSTN, while also add a recurrent layer on top of the localization network, like in Recurrent Spatial Transformer Networks (RSTNs) (Sønderby *et al.*, 2015). Since RSTNs obtain better results than STNs and TSTNs do the same, we could expect improved results by combining these two ideas (using TRSTNs).

TSTNs could also be applied to handwriting recognition in 'noisy environments' e.g. historical documents or textline recognition of textlines in which the result of the segmentation process produces noise (e.g. fragments of text from neighboring lines). They could also be used for more difficult tasks, an obvious choice being object recognition.

## Acknowledgements

This research was supported by a DGA-MRIS scholarship. We wish to thank NVIDIA Corporation for the donation of the GPU which was used to perform this research.

## 6. References

Bastien F., Lamblin P., Pascanu R., Bergstra J., Goodfellow I. J., Bergeron A., Bouchard N., Bengio Y., "Theano: new features and speed improvements", , Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012.

- Bergstra J., Breuleux O., Bastien F., Lamblin P., Pascanu R., Desjardins G., Turian J., Warde-Farley D., Bengio Y., “Theano: a CPU and GPU Math Expression Compiler”, *Proceedings of the Python for Scientific Computing Conference (SciPy)*, June, 2010. Oral Presentation.
- Bluche T., Ney H., Kermorvant C., “The LIMSI handwriting recognition system for the HTRtS 2014 contest”, *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, IEEE, p. 86-90, 8, 2015.
- Dieleman S., Schlüter J., Raffel C., Olson E., Sønderby S. K., Nouri D., Maturana D., Thoma M., Battenberg E., Kelly J., Fauw J. D., Heilman M., diogo149, McFee B., Weideman H., takacsg84, peterderivaz, Jon, instagibbs, Rasul D. K., CongLiu, Britefury, Degraive J., “Lasagne: First release.”, August, 2015.
- Glorot X., Bordes A., Bengio Y., “Deep Sparse Rectifier Neural Networks”, *International Conference on Artificial Intelligence and Statistics*, p. 315-323, 2011.
- Graves A., Schmidhuber J., “Offline Handwriting Recognition with Multidimensional Recurrent Neural Networks”, *Advances in Neural Information Processing Systems*, p. 545-552, 2009.
- He K., Zhang X., Ren S., Sun J., “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification”, 2015.
- Jaderberg M., Simonyan K., Zisserman A., Kavukcuoglu K., “Spatial Transformer Networks”, 6, 2015.
- Kingma D., Ba J., “Adam: A Method for Stochastic Optimization”, 2014.
- Krizhevsky A., Sutskever I., Hinton G. E., “ImageNet Classification with Deep Convolutional Neural Networks”, *Advances in Neural Information Processing Systems*, p. 1097-1105, 2012.
- Russakovsky O., Deng J., Su H., Krause J., Satheesh S., Ma S., Huang Z., Karpathy A., Khosla A., Bernstein M., Berg A. C., Fei-Fei L., “ImageNet Large Scale Visual Recognition Challenge”, *International Journal of Computer Vision*, vol. 115, n<sup>o</sup> 3, p. 211-252, 2015.
- Simonyan K., Zisserman A., “Very Deep Convolutional Networks for Large-Scale Image Recognition”, 2014.
- Sønderby S. K., Sønderby C. K., Maaløe L., Winther O., “Recurrent Spatial Transformer Networks”, 9, 2015.
- Srivastava N., Hinton G. E., Krizhevsky A., Sutskever I., Salakhutdinov R., “Dropout: a simple way to prevent neural networks from overfitting.”, *Journal of Machine Learning Research*, vol. 15, n<sup>o</sup> 1, p. 1929-1958, 2014.
- Sønderby S. K., “Cluttered MNIST dataset”, , [https://s3.amazonaws.com/lasagne/recipes/datasets/mnist\\_cluttered\\_60x60\\_6distortions.npz](https://s3.amazonaws.com/lasagne/recipes/datasets/mnist_cluttered_60x60_6distortions.npz), 11, 2015a. [Online; accessed 30-November-2015].
- Sønderby S. K., “Spatial Transformer Network code example (‘recipe’)”, , [https://github.com/Lasagne/Recipes/blob/master/examples/spatial\\_transformer\\_network.ipynb](https://github.com/Lasagne/Recipes/blob/master/examples/spatial_transformer_network.ipynb), 11, 2015b. [Online; accessed 30-November-2015].
- Sønderby S. K., “Spatial Transformer Network code repository”, , [https://github.com/skaae/transformer\\_network](https://github.com/skaae/transformer_network), 11, 2015c. [Online; accessed 30-November-2015].
- Yann LeCun Corinna Cortes C. J. B., “The MNIST database of handwritten digits”, , <http://yann.lecun.com/exdb/mnist/>, 11, 2015. [Online; accessed 30-November-2015].

---

# Building a knowledge base using Microblogs: the case of festivals and location-based events

**Hoang Thi Bich Ngoc, Josiane Mothe**

*Université de Toulouse, UT2J, IRIT, CNRS UMR5505, Toulouse, France  
{thi-bich-ngoc.hoang, josiane.mothe}@irit.fr*

---

*RÉSUMÉ. Les médias sociaux comme twitter sont très utilisés lors d'un événement (conférence, catastrophe, événement culturel...) pour collaborativement commenter ou donner des avis sur son déroulement. Les utilisateurs du réseau social sont alors avertis via les personnes qu'ils suivent ou en recherchant les tweets portant sur l'évènement. Cependant compte tenu de la taille d'un tweet, l'information obtenue par un seul post est souvent très partielle. L'utilisation d'un ensemble de tweets sur un événement peut permettre d'avoir une vue plus complète en combinant toutes les informations postées. Dans cet article, nous proposons un modèle de représentation d'une collection de microblogs basé sur une ontologie de domaine. Nous indiquons également comment populer cette ontologie en se basant à la fois sur la collection de tweets mais également sur des collections externes. Nous appliquons notre modèle au cas des tweets sur les festivals (collection issue du challenge CLEF 2016) et montrons comment il pourra être utilisé pour faire des recommandations.*

*ABSTRACT. Social media like Twitter are used during an event (catastrophe, cultural events ...) to collaboratively comment or advise on that event. Social network users are then notified through the people they follow or by seeking tweets related to the event. However, given the size of a tweet, the information obtained by a single post is often very partial. Using a set of tweets about an event makes it possible to have a more complete view by combining all the information posted. In this paper, we propose a model to represent a collection of micro-blogs based on a domain ontology. We also show how to populate this ontology based on both the collection of tweets and external collections. We apply our model to the case of tweets on festivals (a collection of the CLEF 2016 challenge) and show how it can be used to make recommendations.*

*MOTS-CLÉS : Base de connaissance de domaine, Microblog, Extraction d'information à partir de tweets.*

*KEYWORDS: Domain knowledge base, Microblog, Information extraction from tweets.*

---

## 1. Introduction

The power of social networks can be illustrated by the number of worldwide users which is expected to reach some 2.5 billion by 2018 according to Statista<sup>1</sup>. According to the same source Twitter is one of the leading worldwide social networks based on active users. Twitter enables users to send short 140-character messages and to follow posts from other users. Live-tweeting events such as conferences or cultural events is very popular and is basically a community that engages online while sharing topical conversations and thoughts on current experiences (Nagarajan *et al.*, 2010). During an event, some Twitter users will discuss, comment, or advise on this event while their followers will be notified. Alternatively, it is possible for a Twitter user to search for tweets related to some content using the Search API.

However, given the 140-character size of a tweet, the information obtained by a single tweet is often very partial. It is more likely that a user rather needs to read a set of tweets to get a clear picture of an event.

For example, the two following tweets all related to Cannes 2015 provide different and complementary pieces of information :

Vincent Lindon & Gaspar No  , guests of honour at #VentanaSur  
Festival de Cannes Film Week from 30/11/15 to 6/12/15!  
[pic.twitter.com/slPVKflt24](http://pic.twitter.com/slPVKflt24)

Irina Shayk, somptueuse, lors du tapis rouge du 19 mai 2015     
Cannes, [pinterest.com/pin/4530340437...](http://pinterest.com/pin/4530340437...)

The first tweet is about the film *Carol* directed by *Todd Haynes* to be presented at the *Cannes 2015* festival. While the second tweet provides the date of a related event in Buenos Aires (*VentanaSur*) along with two actors who were there ; it is an add for the Buenos Aires festival.

From these two tweets, it is obvious that some users will lack of context to understand them individually and that some information from various tweets help understanding a given tweet. If all the pieces of information from the set of tweets were used to build a knowledge base, it would then be possible to understand better each tweet individually by enriching it using additional knowledge. For example, understanding a tweet would be easier if each entity mentioned in a tweet was explicitly associated with its entity type such as film, festival, player, .... It is possible to populate different parts of the knowledge base by using information extracted from tweets. Even if each tweet taken individually provides partial information, the sum of them could give a better picture of the information. Moreover, some parts of the knowledge base can rely on existing resources such as geographical hierarchies to model the location part for example.

---

1. <http://www.statista.com/topics/1164/social-networks/>

In this paper, we propose a model to represent a collection of micro-blogs that allows to understand better information from a set of tweets on events. There are various ways to represent knowledge, but the semantic web has made ontology a very popular way to encode domain knowledge. The reason is that it enables the implementation of algorithms that include reasoning, inferring new knowledge from existing data (Margara *et al.*, 2014). In the case of the incomplete data from the tweets, an ontology can help inferring new data. This is specifically interesting when considering incomplete stream data.

Our model is based on a domain ontology which is populated using the micro-blogs as well as other Internet-based information. We focus on a specific domain which is the festivals domain that has been chosen by the CLEF 2016 challenge. This paper is centered on the domain representation and on the ontology population ; but we also mention some ways this knowledge base could be used in many applications including for localization-based recommendation.

The rest of the paper is organized as follows : Section 2 presents the related works. Section 3 details the model we suggest to represent the festival domain. Section 4 explains how the knowledge base is populated. Finally, section 6 concludes this paper and presents future works.

## 2. Related work

Due to the rising popularity of social media, many studies have been conducted to study ways to effectively extract information from this plentiful resource. Prior works relevant to our knowledge base are grouped into three categories : ontology-based information extraction, event detection, and location estimation in microblog.

In recent years, a number of papers have addressed the ontology-based information extraction. (Narayan *et al.*, 2010) suggest an approach to populate an ontology with the events retrieved from Twitter. Data is parsed and mined for various properties such as name, date, time, location, type and URL that are later used to populate the ontology. The authors use existing ontology of (Hobbs *et al.*, 2004) to identify Time and Alexandria Digital Library Gazetteer (1999) to recognize Location and Name. Using these ways, they omit NE when it is not explicitly mentioned in the tweet content. Our work also recognizes NE in tweets (artist, time, location, festival...) but we combine Stanford Named Entity Recognition (NER) <sup>2</sup> and mining Twitter users' profile. To detect festivals in tweets, we compare string similarity of tweet content to the list of festivals accompanied by some properties such as festival names, twitter accounts, twitter hash-tags and keywords retrieved from Wikipedia and tourism websites. Different from our approach, (Kontopoulos *et al.*, 2013) present the deployment of ontology-based sentiment analysis of tweets. They first identify the subject discussed in tweets and then give each tweet the sentiment score for each distinct aspect relevant to the subject.

---

2. <http://stanfordnlp.github.io/CoreNLP/>

In the area of event detection, (Weng *et al.*, 2011) build signals for individual words and filter out trivial words based on their corresponding auto correlations signal. They extract events by clustering signals together using modularity-based graph partitioning. Similarly, (Zhao *et al.*, 2007) propose the text-based clustering and temporal segmentation combined with information flow-based graph. Using a different approach, (Quack *et al.*, 2008) detect local events by analyzing community photo collections while (Lee *et al.*, 2010) and (Watanabe *et al.*, 2011) analyze the geographical distribution of geotagged micro-blogs to detect events. Our work addresses the event detection in a different way. We use external resources from Wikipedia and official websites to get a list of festivals and accompanied properties retrieve tweets related to events based on the string comparison.

Finally, location estimation from microblog documents has been widely studied in recent years. NE recognition (NER) systems have addressed the problem of retrieving location specified in documents ; however they do not perform very well on informal texts (Huang *et al.*, 2015). The literature proposes some methods to improve this limitation. (Liu *et al.*, 2011) combine a K-Nearest Neighbors classifier with a linear Conditional Random Fields model under a semi-supervised learning framework to tackle the lack of information in microblog. Another location estimation approach based on analyzing geo-location by content analysis either with terms in gazetteer (Fink *et al.*, 2009), with probabilistic model (Cheng *et al.*, 2010), or users' networking (Chandra *et al.*, 2011). In our approach, we solve the problem of location identification in tweets combining two techniques : 1) using Stanford NER ; 2) extracting user hometown. These techniques complement each others in the location detection process. In the cases when location is not detected by Stanford NER, we mine Twitter user's profile to extract his hometown and consider this location as the one a tweet is about.

### **3. Knowledge base model : the geographical-festival ontology**

Events have several dimensions, the main are :

- Location information which indicates *where* the event took, takes or will take place ;
- Temporal information that indicates *when* the event takes place ;
- Entity-related information which indicates what the event *is about* ;

In addition to their content, tweets contain additional dimensions that can be considered as meta-information and which are related to the user who posted the tweet. These dimensions can also be separated into location (from where the tweet was posted), temporal information (when the tweet has been posted) and entity-related information (who posted the tweet).

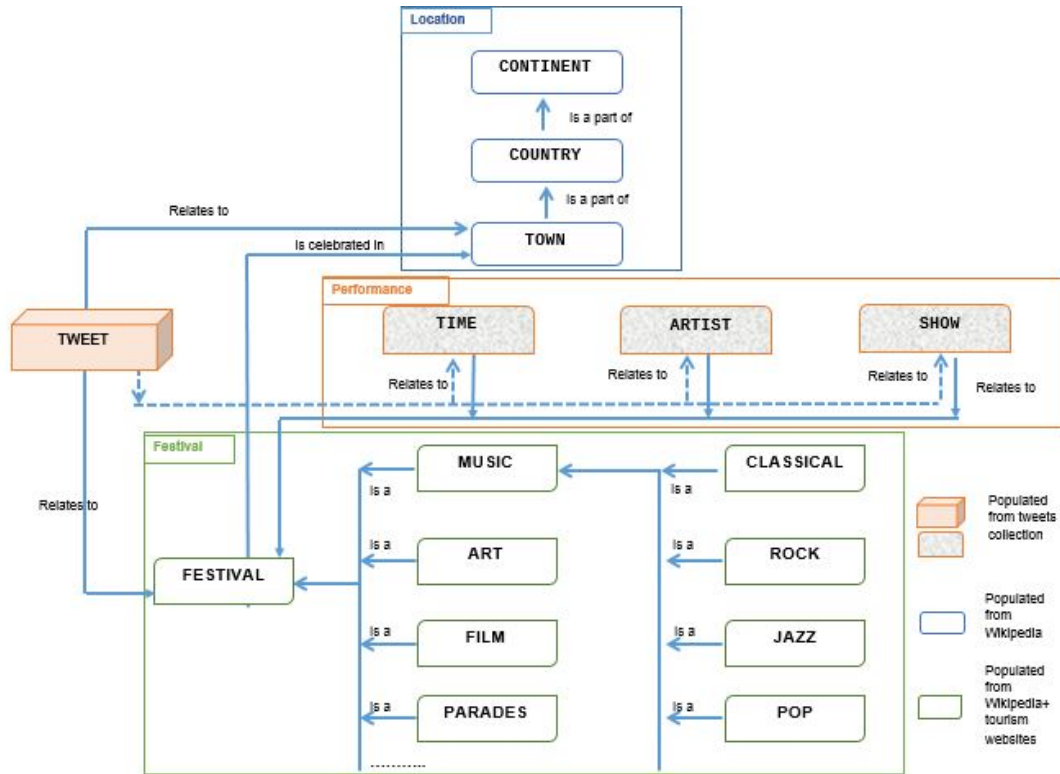


**Figure 1.** Model to represent events - the case of the Festival ontology

In the case of festival-related events, entities can be depicted in a more specific way since the types of entities can be set such as the type of festival for example, as we will see in details in the rest of this section.

Figure 1 provides the model of the knowledge base that represents the events associated to festivals. In this ontology, at this stage, we only represent the events, that is to say the tweet content, not the additional information or meta-information related to tweets such as the users.

Our geographical-festival ontology includes four sub-parts : the first part (top part of the Figure 1 - Location) represents the locations of the events, the second part (Performance) presents performance information related to each event while the third part (Festival) concerns the festivals in general. Finally, Tweet class includes the tweets related to festivals or locations. We make this splitting mainly to ease the description of the ontology.

The location part of the ontology is a hierarchy. Countries over the world are constituted in different ways, for example the United-States is divided in States, then in counties or county-equivalents, then in towns, while France is divided into regions, departments, towns. Towns can in turns be divided in parts. Considering the domain we are interested in, the town level looks appropriated as the deeper level. We then

simplify the hierarchy so that it works for any part of the world. We finally kept a three level hierarchy : Town, Country, Continent, related by Is-part-of relationships. The way this part of the ontology is populated presented in section 4.2.

The second part of the ontology gathers the Performance information with three classes : Time, Artist and Show. There is not a unique resource to consider to populate this part of the ontology ; rather, it can be based both on festival or related web sites and on the tweets them-selves.

The third part of the ontology represents the Festivals. Wikipedia classifies many festivals into a set of categories and provides a hierarchy : some classes, in turn, are composed of a set of other classes. For instance, the *Music* class consists of Classical, Rock, Jazz, Pop... We re-use this set of categories to contribute to the Festival part of our ontology including a number of classes such as Music, Art, Film, Parades and so on which are types of Festivals. It might not be complete but it is appropriate to start with, and it can be completed later on, considering tweets contents.

Finally, Tweet class contains tweets relates to a festival or a specific location. From a tweet, it is possible to extract information to populate entities in the Performance part of the ontology.

#### **4. Populating the domain ontology**

In this section, we first provide the general principles of the knowledge base population (section 4.1), then we detail the various steps of the ontology population (section 4.2).

##### **4.1. Principles**

The domain ontology is populated considering complementary resources. As mentioned in the previous section, we use both a flow of tweets that match the information need *festival* and which can be seen as our main resource for fresh information, and external resources such as Wikipedia or web sites that contains more stable information even if they can be frequently up-dated (specifically considering future festivals).

Figure 2 depicts the overall principle of the ontology population : Wikipedia and tourism websites are used to first populate the ontology. Wikipedia provides some general information about existing locations, festival categories and even most of recognized festivals ; websites that advertise festivals provide more specific information about some festivals and some hub such as the Syndicats d’initiative web sites can also provide some additional links to other festivals. Then the domain collection consisting of tweets on festivals is used to extract additional information. The ontology and the collection are used in a process that combines the information : from the ontology, we know locations and types of festivals that help analyzing the tweets and allows us to extract new information to populate the ontology.



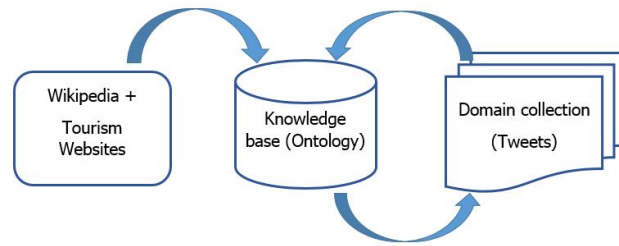


**Figure 2.** The arrows show how a resource is used. Wikipedia and tourism websites are used to populate the ontology ; the ontology is used to help information extraction from the tweet collection and the additional extracted information is used to populate the ontology.

Thus, the ontology population using Wikipedia and official websites resources can be seen as resources for background ontology population while tweet collection is a resource for providing complementary views about the events

In our approach, we choose Protégé<sup>3</sup> to build the ontology that implements the knowledge base and use a part of CLEF 2016 collection which includes 38,686,650 tweets about festivals in the world collected from May to October 2015.

#### 4.2. Detailed process

Populating the festival ontology is achieved using the following steps :

1) The Location part of the ontology is populated using (Ngo *et al.*, 2012) results. They extract the geographic data from Wikipedia<sup>4</sup> which proposes adequate information of all cities by countries in the world. Data is structured in RDF format before being populated to the ontology.

2) We extract data of festivals by categories from Wikipedia and official tourism websites to populate the Festival part of the ontology. Although the information from these resources changes, the update rate from these resources is not necessarily very high to keep the ontology accurate. In our approach, we have not yet automated the festivals collecting process, but this structured information can be extracted using SPARQL on DBpedia<sup>5</sup>.

3) The location entities from the ontology are then used to retrieve all tweets related to a specific location (we consider the tweets that explicitly mention a location and possibly tweets for which location can be inferred from the tweet content). We use Stanford (NER) to identify locations in a tweet content if locations are explicitly

3. <http://protege.stanford.edu/> is an open-source ontology editor and framework

4. [https://en.wikipedia.org/wiki/Lists\\_of\\_cities\\_by\\_country](https://en.wikipedia.org/wiki/Lists_of_cities_by_country)

5. <http://dbpedia.org/snorql/> : BDpedia presents structured data of Wikipedia pages

mentioned. Otherwise, we mine Twitter user profile to extract hometown and consider it as the location this tweet is about.

4) From above identified tweets, we then use the list of festivals (resulted in step 2) with some properties such as Twitter account, hashtag, keywords collected from festivals official websites to detect festival events by string comparison to tweet contents. The relationships between tweets and festivals are established in this step.

5) Finally, we automatically populate the Performance part by time, artist... extracted from tweets (results of step 3) using Stanford NER.

## **5. Some applications**

Our knowledge base model could be applied in a broad range of applications in several domains such as tourism, transportation, marketing and advertisement.

In the field of tourism, a graphical event recommender system could be developed on top of the festival ontology. Tourists would be proposed information time, famous people and related activities of events celebrated surrounding them or any specific place without spending time to search and process information. In addition, users could find latest news, opinions and feedback of attendees on tweets about events rather than browsing many tourism websites.

In the transportation domain, a system suggesting suitable route or transportation mean to avoid crowds, traffic jam or other problems linked to events could be developed using the information stored in the knowledge base for travelers.

Besides, festival events would be perfect places for companies to market their brand. They could communicate with thousands of participants and engage participants through targeted campaigns. Knowing the type of festivals, type of participants as well as the time, artists, shows ; companies could propose and implement effective advertisement campaigns for their products.

## **6. Conclusions and Future work**

In this paper, we introduce an approach of building a knowledge base using Twitter and other external resources for the case of festivals and location-based events.

The model considers festivals organized in a specific location and related information such as time, artist or show in order to be able to provide short highly informative suggestions by analyzing tweets.

For this purpose, we define a geographical – festival ontology. As a background task, a first population of the ontology is based on various resources such as Wikipedia and official tourism websites. In addition, retrieved tweets of a specific location and events are used to populate the ontology and are analyzed to extract concerned data.

In future work, our knowledge base can be developed to be a visualized event recommendation system for users basing on their current location and other aspects such as their profile, interest and festivals that their friends participate.

## 7. Bibliographie

- Chandra S., Khan L., Muhaya F. B., « Estimating twitter user location using social interactions—a content based approach », *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third International Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on*, 2011.
- Cheng Z., Caverlee J., Lee K., « You are where you tweet : a content-based approach to geolocating twitter users », *Proceedings of the 19th ACM international conference on Information and knowledge management*, 2010.
- Fink C., Piatko C. D., Mayfield J., Finin T., Martineau J., « Geolocating Blogs from Their Textual Content. », *AAAI Spring Symposium : Social Semantic Web : Where Web 2.0 Meets Web 3.0*, 2009.
- Hobbs J. R., Pan, Feng, « An ontology of time for the semantic web », *ACM Transactions on Asian Language Information Processing*, vol. 3, n° 1, p. 66-85, 2004.
- Huang Y., Liu Z., Nguyen P., « Location-based event search in social texts », *Computing, Networking and Communications (ICNC), 2015 International Conference on*, 2015.
- Kontopoulos E., Berberidis C., Dergiades T., Bassiliades N., « Ontology-based sentiment analysis of twitter posts », *Expert systems with applications*, 2013.
- Lee R., Sumiya, Kazutoshi, « Measuring geographical regularities of crowd behaviors for Twitter-based geo-social event detection », *Proceedings of the 2nd ACM SIGSPATIAL international workshop on location based social networksp*. 1-10, 2010.
- Liu X., Zhang S., Wei F., Zhou M., « Recognizing named entities in tweets », *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies-Volume 1*, 2011.
- Margara A., Urbani J., van Harmelen F., Bal H., « Streaming the web : Reasoning over dynamic data », *Web Semantics : Science, Services and Agents on the World Wide Web*, vol. 25, p. 24-44, 2014.
- Nagarajan M., Purohit H., Sheth A. P., « A Qualitative Examination of Topical Tweet and Retweet Practices. », *ICWSM*, 2010.
- Narayan S., Prodanovic S., Elahi M. F., Bogart Z., « Population and Enrichment of Event Ontology using Twitter », *Information Management SPIM 2010*, 2010.
- Ngo Q.-H., Doan S., Winiwarer W., « Using Wikipedia for extracting hierarchy and building geo-ontology », *International Journal of Web Information Systems*, 2012.
- Quack T., Leibe B., Van Gool L., « World-scale mining of objects and events from community photo collections », *Proceedings of the 2008 international conference on Content-based image and video retrievalp*. 47-56, 2008.
- Watanabe K., Ochi M., Okabe M., Onai R., « Jasmine : a real-time local-event detection system based on geolocation information propagated to microblogs », *Proceedings of the 20th ACM international conference on Information and knowledge managementp*. 2541-2544, 2011.

Building a Knowledge Base Using Microblogs: the Case of Festivals and  
Location-Based Events

Weng J., Lee, Bu-Sung., « Event Detection in Twitter », *ICWSM*, vol. 11, n<sup>o</sup> 3, p. 401-408, 2011.

Zhao Q., Mitra P., Chen B., « Temporal and information flow based event detection from social text streams », *AAAI*, vol. 7, p. 1501-1506, 2007.

---

# Processing natural language queries to disambiguate named entities and extract users' goals: application to e-Tourism

**Sanjay Kamath, Lorraine Goeuriot, Marie-Christine Fauvet**

*LIG (MRIM), University of Grenoble Alpes, 38000, Grenoble, France*

---

**RÉSUMÉ.** Cet article présente une étude qui s'inscrit dans le cadre d'un projet plus large qui porte sur la conception et la réalisation d'un système visant à fournir à des utilisateurs mobiles des services personnalisés, dépendant de leur contexte, et adaptés à leurs besoins. Par exemple, un utilisateur peut vouloir des informations sur la météo du lendemain, ou bien réserver des billets d'entrée à un musée voisin, ou encore réserver une table dans un restaurant italien et obtenir les indications pour s'y rendre en voiture. Dans cet article, nous étudions plus particulièrement les problèmes liés à l'extraction de requête fournie par l'utilisateur des paramètres nécessaires à son traitement, ainsi que du contexte de l'utilisateur. Le système que nous proposons est basé sur le traitement de requêtes exprimées en langage naturel. Afin d'extraire de la requête les entités nommées ce système s'appuie sur des bases de connaissances et des outils de désambiguïsation.

**ABSTRACT.** This paper presents a study which is part of a broader project. This latter aims at providing mobile users with context-aware personalised services. The E-tourism Project deals with a variety of queries submitted by a tourist, such as booking a hotel room, getting the weather conditions for the next day, or booking tickets in a museum in the neighbourhood and worth to visit. This paper focuses on the query management and processing. The module described analyses and structures the query by splitting it, identifying the named entities, solving ambiguities... To process the query, the system uses various external knowledge bases and Natural Language Processing tools to understand the named entities and proper context of the query using disambiguation techniques.

**MOTS-CLÉS :** e-Tourism, Traitement de Requêtes, Services Personnalisés Dépendants du Contexte

**KEYWORDS:** e-Tourism, Query Processing, Context Aware Personalised Services

---

## 1. Introduction and Motivating Example

With the development of pervasive computing and mobile technologies, mobile applications are getting more and more attention. Distributed computing systems based on context awareness have been proposed in several domains such as health-care, logistics and tourism. Most of the existing service providers in the tourism domain focus only on a few specific goals, such as booking a restaurant, or searching a hotel. These applications, running on mobile devices, are challenged to locate and deliver the right service to the right person, with the appropriate rendering.

The work reported in this paper is part of a broader project which aims at designing and implementing a framework which provides context-aware personalized services for mobile users according to their needs, profile and context. The issues to be addressed in this project are related to system design, software architecture, distributed and heterogeneous resource access and integration, information retrieval and recommender systems. This paper focuses on the user's query management and processing, mainly to highlight the importance of NLP techniques used. Our system considers the user's input in the form of a text query which contains the details and requirements of the user. To illustrate the challenges tackled in this paper, let us consider the following scenario. Alice is an American tourist visiting Paris in France. She picks up her smartphone and issues the query, once connected to our e-Tourism system : *I Want to book a table tonight at the closest restaurant to the Eiffel Tower and know the directions to get there*. The study reported in this paper has the following contributions :

- Recognition and disambiguation of **named entities** using external knowledge sources such as WolframAlpha etc., according to Alice's context the value returned by the GPS embedded in her smartphone says she is located in Paris, Eiffel Tower is then recognized as a named entity in Paris (not in Las Vegas, Nevada, neither in Brisbane, Australia).

- Extraction of user's **goals** : The goals are inferred from the **Named Entities** extracted from the query, In the Example : Alice has two main goals, the first one is *"I want to book a table at the closest restaurant to Eiffel Tower for tonight"*, and the second one *"get directions to reach the restaurant"*.

This paper is organized as follows : we describe the architecture of the whole system in Section 2 ; while we present in Section 3 some related work. Section 4 details our proposed approach. Section 5 details the implementation of this approach and future evaluation of our system. Eventually, in Section 6 we conclude and sketch some further work.

## 2. Architecture of the eTourism System

The study described in this paper is conducted as part of a broader project whose main goal is to provide mobile users with services according to their needs (Nalumpoon *et al.*, 2013).

We call a goal, the task requested by the user (e.g. movie booking, events nearby...). A query might contain one or more goals. The *Context* includes spatial, temporal, physical, and environmental properties such as date, time, location etc. that could be collected by sensors embedded on the devices used to submit the queries. The *Profile* captures users' personal details, preferences and centers of interest.

Figure 1 sketches the overall architecture of this project dedicated to e-tourism (Fauvet *et al.*, 2015). The role of each module and the flow of information are rapidly introduced below.

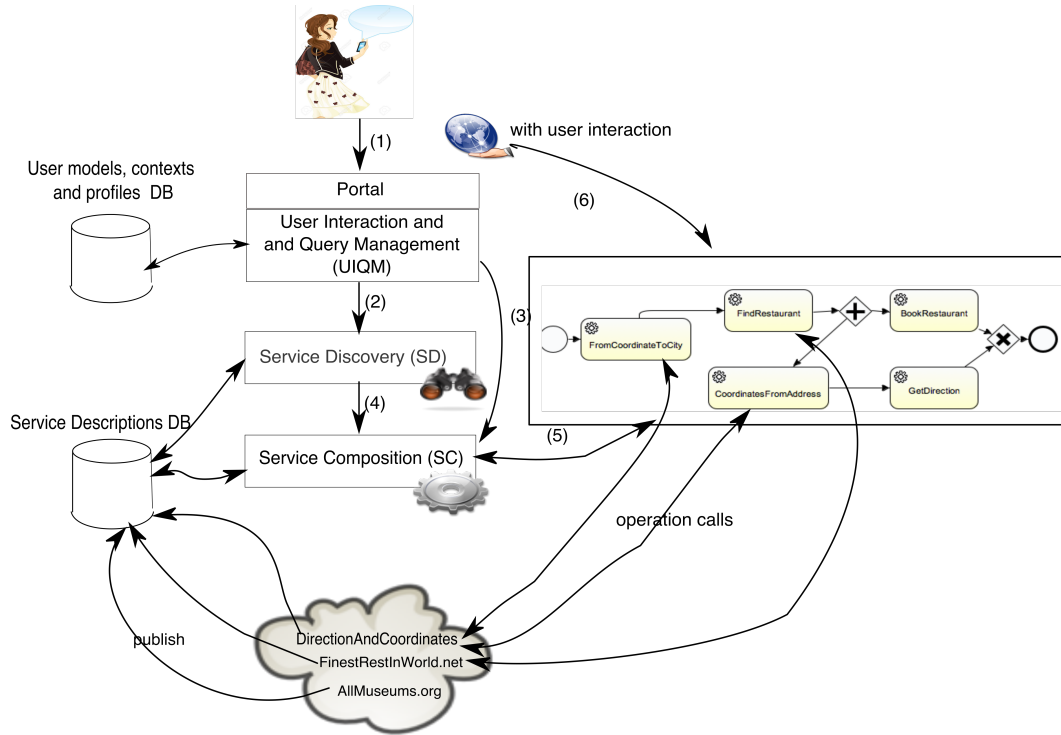


**Figure 1.** A context-aware service discovery and composition system for mobile users.

1) **User interaction and query management module** : aims at managing user interactions by handling the queries submitted by users on their mobile device. A user's query and her identification are received in this module in the data flow (1). This module extracts from the query, the information necessary for the choice and composition of the service components. With data flow (2) it sends single goal queries to the Discovery module and with data flow (3) the user's goals are sent to the Composition and Execution system. This paper focuses only on this module.

2) **Discovery module** : given the user's query received in the data flow (2), her profile and context, this module is responsible for retrieving services among a repository of service descriptions, that once composed can potentially meet the user's goals

expressed in her query. The retrieved services are then sent, in the data flow (4), to the next module. For details see (Caicedo-Castro, 2015 ; Caicedo-Castro *et al.*, 2015).

3) **Composition and execution system** : eventually this module is in charge of automatically composing and executing services returned by the discovery phase. The automated composition is meant to satisfy users' goals. This module results in a BPMN model whose tasks refers to a service operation by the data flow (5) (for details see (Na-Lumpoon, 2015)). During the execution of the resulting process, some interactions with the user might be necessary (see data flow (6)).

### 3. Related Work

(Tomai *et al.*, 2005) discuss a case study on trip planning using two ontologies to represent the user profile and an ontology for tourism data such as shopping malls, cinemas, activity spots nearby etc. Assumption of the user's location is always the city center because of which the user's device is not constantly looking for GPS signals and User profile related information is collected through a questionnaire to the user. The context matching algorithm uses this information to determine the feasibility of the plan in accordance to the time the traveler has.

Ontology population and the evaluation for tourism domain corpus is discussed in (Ruiz-Martinez *et al.*, 2011) who propose a method for extracting semantic content from textual web documents (such as data about hotels and restaurants) which were obtained from an official tourism web page, to automatically instantiate a tourism domain ontology. Ontology Population deals with the extraction and classification of instances of the concepts and relations that have to be defined in the ontology. Semantic contents are obtained after the NL processing of the web documents using GATE<sup>1</sup> framework and are considered as instance candidates which undergo disambiguation for semantic ambiguities with the use of NLP tools. These contents are then related with their ontology entities by ontology population process.

Similar to DBpedia<sup>2</sup>, some efforts were made in building TourPedia (Cresci *et al.*, 2014), which includes point of interests, accommodations, restaurants and attractions using various commercial data sources. However, TourPedia covers only a part of Europe.

Our system aims at extracting named entities from the query, disambiguate the similar results, link the goal(s) of the query with Point of Interest(s) and substitute the required parameters for the goals. To do so, we use widely studied and utilized approaches in Natural Language processing such as Named Entity recognition, Part of Speech tagging and Disambiguation methods.

---

1. <https://gate.ac.uk/>

2. <http://wiki.dbpedia.org>



#### 4. Proposed Approach

As explained in Section 2, we focus here on the Query Management Module of the eTourism system which is depicted in Figure 2.

Firstly the queries are pre-processed using NLP tools to obtain an intermediate representation of the free text query, which facilitates the Named Entity recognition tool to determine the Named Entities from the query. After the Named Entities are obtained, the Disambiguation of them is carried out using external knowledge sources. The parameters required for the Service composition and Service execution modules are inferred from the User profile, Named entities and Goals from the query.

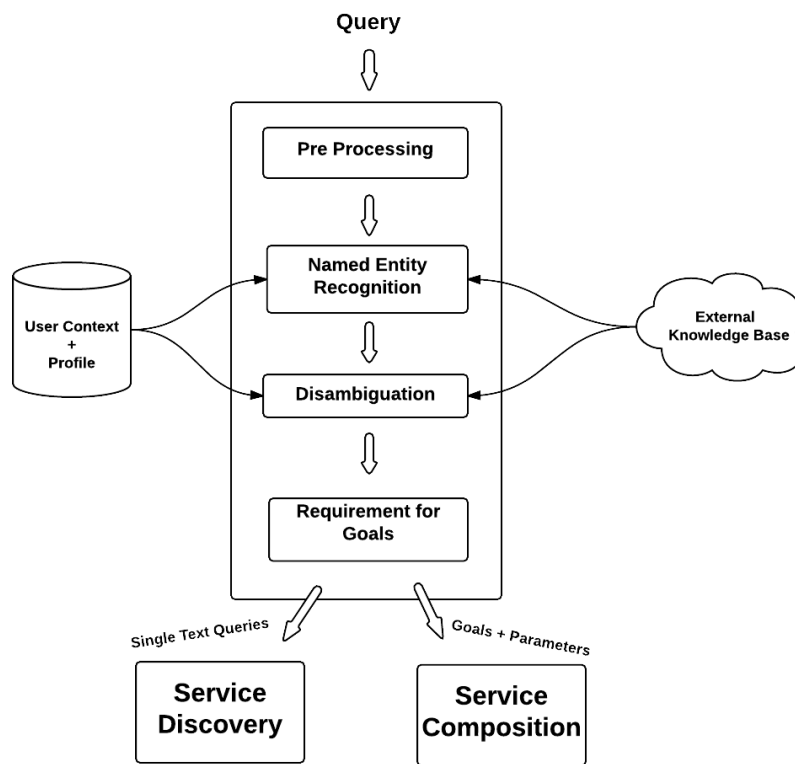


**Figure 2.** Architecture of Query Management Module

##### 4.1. Pre-Processing of the Queries

The queries are short text composed of keywords. They can contain operators such as boolean operators (and, or, not) and conditional operators (if, then, else, otherwise, if not). Queries can contain one or multiple goals, separated by punctuation or operators.

Our approach performs the shallow parsing on the queries for keywords mentioned above, and splits the queries into separate single queries. We consider single queries

as queries containing an unique need (single goal). Single queries are separated using Boolean operator keywords as mentioned above. We perform Part of Speech tagging on the queries to extract the *Verbs* and our assumption is that the *Verbs* represent the Goals.

#### 4.2. Named Entity Recognition

Named Entity Recognition (NER) is a subtask of information extraction which is performed on the input queries to determine the entities which represent locations, places, POIs etc. NER is a wider domain of Natural Language Processing, we discuss the use of NER for our system which uses knowledge sources as input in Section 5.

Example scenario : **Q1** : *How far is the Colosseum from here ?*

The NER tool takes the tokenized form of the query **Q1** as input and detects the Named Entity as **Colosseum**. If there are multiple Entities with the same name, there is ambiguity in the results. Hence we disambiguate such results, as discussed in the next subsection.

In the cases where the **Goals** of the query which are supposedly Verbs, are ambiguous with the Named Entities, we pose a question to the user and get a response to obtain the right choice. One of the example for such scenario is the use of the movie name **The Grand Budapest Hotel** which can either be a goal for Hotel booking query.

#### 4.3. Disambiguation

When two Named Entities such as name of hotels (Park Hyatt, Paris and Park Hyatt, Nice) are the outputs from a query of the user, we decide to consider the closest one to the user (in terms of location) as the correct one. In the case of an ambiguity, we need methods to rank the potential entities. We use an algorithm for the ranking function *R* to do so using the user and the entity locations. The main steps of this algorithm are :

Inputs: SRC = User's location obtained using GPS

S = list containing the named entities from external sources

Threshold = the maximum distance between the user and POIs  
(Threshold value must be agreed by the user.)

R(NE, SRC, Threshold):

for(i=0; i< S.length; i++)

{if (distance (SRC, location(S)) > Threshold) then (remove item(i));}

S.sort(ascending);

Output:

Top result from List S.

We need the top most result after sorting the list in ascending order of the location.

Example scenario : **Q1** : *How far is the Eiffel Tower from here ?*

If the User is in France when he inputs this query, then the **Eiffel Tower** is the one in Paris, France. Whereas if the User is in Las Vegas, Nevada then the **Eiffel Tower** is the one in Las Vegas, Nevada. The context of the location changes the meaning of certain Named Entities. In the example, the result of the algorithm will be Eiffel Tower, Paris because of the user's location which is in France.

#### 4.4. Requirements for goals

Each of the user's goal has specific parameters required for composition and execution of services, as shown in Table 1. For instance, a movie booking has as parameters the movie name, cinema name, number of people, the city where the cinema is located and day and time of the booking.

| Goals          | Parameters                                                                 |
|----------------|----------------------------------------------------------------------------|
| Movie booking  | movie name, cinema name, city name, people number, day and time of booking |
| Hotel booking  | hotel name, city name, people number, checking day, checkout day           |
| Museum booking | museum name, city name, people number, visit day                           |
| Direction      | mode, source, destination                                                  |

**Tableau 1.** *Expected parameters depending on goals.*

In cases where the system has conflicts in filling in the parameters, the interaction module asks the user to input that particular parameter again rather than making wrong assumptions. We encountered the *synonymy problem* in this phase. Words like 'Book' and 'Reserve' both refer to the same verb. To tackle this problem, we use *Wordnet*. The results from the query processing module are the parameters required for the composition and execution of services.

An example query and its intermediate representation :

**If it rains, then order a pizza to my room at 8pm.**

**Q1** : *If it rains* **Q2** : *order a pizza to my room at 8pm.*

*Rain = True ; Goal=PizzaOrder*

*User Profile : Favorite\_Pizza : Orientale ;*

*Hotel\_Address : Room No.23, Park Hotel,Grenoble.*

#### **Parameters :**

*Goal : Weather\_Information,Pizza\_Order*

*SrcLoc : 45.1877997,5.7646932*

*Favorite\_Pizza : Orientale ;*  
*Hotel\_Address : Room No.23, Park Hotel,Grenoble.*  
*Time : 20 :00*

## 5. Implementation and Evaluation

The Figure 2 sketches the overall architecture of the *User Interactions and Query Management module*. The role of each sub module is discussed in details below. The query is obtained from the user through a *Graphical User Interface*<sup>3</sup>. Our system is still in implementation phase.

### 5.1. Dataset

We built a set of 45 simple queries in Natural Language representing various use case scenarios such as Hotel Booking, Restaurant Reservation, Movie Ticket Booking, Driving Directions etc. We plan to build larger corpus of queries for performing evaluations. The queries are divided into different categories such as Single goal queries, Multiple goal queries(separated by keywords such as and,or,along with,also etc.), Conditional queries(separated by keywords such as if,not,if not,without etc.).

### 5.2. System Setup

Apache OpenNLP<sup>4</sup> is a Java machine learning toolkit for Natural Language Processing (NLP). It supports the most common NLP tasks, including rule based and statistical Named Entity Recognition. For processing the queries we use the provided tools such as Tokenizer and POS Tagger<sup>5</sup>. Tokenizer splits the queries into tokens and the POS Tagger appends the tokens with the part of the speech corresponding to that word in the query. This part of the speech value can be used to determine the goals, as goals are always verbs in our approach.

### 5.3. Named Entity Recognition and Disambiguation

Apache OpenNLP toolkit also provides a tool for Named Entity Recognition. The external knowledge sources we use are WolframAlpha<sup>6</sup>, Google places<sup>7</sup>, Foursquare<sup>8</sup>.

3. <http://lig-membres.imag.fr/etourism/en/>

4. <https://opennlp.apache.org/documentation/1.5.3/manual/opennlp.html>

5. <http://opennlp.apache.org>

6. <http://www.wolframalpha.com>

7. <http://developers.google.com/places>

8. <https://foursquare.com/>

For queries containing the use of keywords representing POIs in it, such as *Eiffel Tower*, we use the Foursquare data to obtain results of places similar to the Named entity. For other keywords such as "Movie", "Monument" etc. we use Wolfram Alpha to extract the Named entities. We use Google Places API to obtain the coordinates for these locations which we use in the Disambiguation process. The System parses the output from the sources and ranks them using the ranking function as described in the Disambiguation Algorithm.

#### **5.4. Requirements for goals**

Goal(s) from the query are extracted using the tools discussed above. We use Wordnet data to compare the results with goals. We classify the goals mentioned above using a key for representing each goal. This key is stored along with the set of goals. For example, "Book" is a key for all booking-based queries. Whenever a *Verb* is encountered in the query, Wordnet is accessed to get the verbs which are synonyms to the input verb. Data obtained from Wordnet are matched with keys to determine the key for the set of goals which it matched. We substitute the data present in the query for the parameters we assumed in Table 1. Some of the parameters present in the query match the parameters assumed, and some parameters might be missing.

The system labels it as missing parameter and passes it to the service composition module. This module does not invoke an interaction with the user for the missing parameters, because the service operations might need few parameters and some of the parameters obtained from the user might not be used. Hence invoking the user for appropriate missing parameter is solely dependent on service execution module.

#### **5.5. Results**

The results generated from this module are : 1) Single Text queries or Single Goal queries which are the inputs for the Service Discovery module. 2) Goals + Parameters which are the inputs for the Service Composition module.

#### **5.6. Evaluation**

The Implementation of eTourism system is in progress yet along with the corpus of queries. We plan to evaluate this module in two ways :

- Comparing the expected goals from a query with the predefined goals for each query in the corpus.
- Determining the number of missing goals from the query which contains complete set of information.

## 6. Conclusion and Future Work

We present an approach to disambiguate Named entities related to Tourism based on location information. Other context information can also be used to process the queries. One of the main concerns about the above approach is privacy. Users might not disclose their location. The temporal information which are hidden in named entities can be used to determine the time related information and compute the intervals between the events to make the system more dynamic to hidden timed entities and hidden time intervals in them.

## Acknowledgement

This work has been carried out in the context of the Guimuteic project funded by Fonds Européen de Développement Régional (FEDER) of région Auvergne Rhône-Alpes.

## 7. Bibliographie

- Caicedo-Castro I.-B., « Sniffer : A text description-based service search system », , PhD Dissertation, University of Grenoble, 2015.
- Caicedo-Castro I.-B., Fauvet M.-C., Lbath A., Duarte-Amaya H., « Toward the highest effectiveness in text description-based service retrieval », *Document Numérique*, vol. 2-3, p. 155-177, 2015.
- Cresci S., D'Errico A., Gazzé D., Lo Duca A., Marchetti A., Tesconi M., « Towards a DBpedia of Tourism : the case of Tourpedia », *Proceedings of the 2014 International Conference on Semantic Web-Poster and Demo Track, ISWC2014*, 2014.
- Fauvet M.-C., Kamath S., Caicedo-Castro I., Lbath A., Goeuriot L., « Offering Context-Aware Personalised Services for Mobile Users », *Proc. of the Inter. Conf. on Service Computing Conf. (ICSOC) demo session*, Goa, India, 2015.
- Na-Lumpoon P., « Toward a Framework for Automated Service Composition and Execution : E\_Tourism Applications », , PhD Dissertation, University of Grenoble, 2015.
- Na-Lumpoon P., Lei M., Caicedo-Castro I., Fauvet M., Lbath A., « Context-Aware Service Discovering System for Nomad Users », *Proc. of the 7th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*, 2013.
- Ruiz-Martinez J., Minarro-Giménez J., Castellanos-Nieves D., García-Sánchez F., Valencia-García R., « Ontology population : an application for the E-tourism domain », *International Journal of Innovative Computing, Information and Control (IJICIC)*, vol. 7, n° 11, p. 6115-6134, 2011.
- Tomai E., Spanaki M., Prastacos P., Kavouras M., « Ontology assisted decision making—a case study in trip planning for tourism », *On the Move to Meaningful Internet Systems 2005 : OTM 2005 Workshops*, Springer, p. 1137-1146, 2005.

---

# Extraction automatique de termes-clés : Comparaison de méthodes non supervisées

Sous la direction de Josiane Mothe\*\*

Faneva Ramiandrisoa\*

\* LAMAI (Laboratoire de Mathématiques Appliquées et Informatique)  
Université d'Antananarivo, Madagascar. r.faneva.mahery@gmail.com

\*\* IRIT (Institut de Recherche en Informatique de Toulouse), UMR5505 CNRS,  
Université de Toulouse, Toulouse, France. josiane.mothe@irit.fr

---

*RÉSUMÉ.* Cet article présente un état de l'art et une comparaison des méthodes non supervisées et automatiques d'extraction de mots-clés à partir des contenus textuels de documents. Nous évaluons plusieurs méthodes de la littérature sur deux corpus de documents en comparant des termes-clés extraits et ceux associés initialement aux documents. Nous avons pu constater que la méthode basée sur une mesure TF-IDF était celle qui renvoie les résultats les plus proches des mots-clés des auteurs des documents.

*ABSTRACT.* This article presents a state of the art and a comparison of unsupervised methods for automatic keywords extraction from documents. We evaluate several methods from the literature on two sets of documents by comparing the keywords extracted and those initially associated with documents. We found that the best method (the one that retrieves keywords the closest to the authors' keywords) is based on TF-IDF.

*MOTS-CLÉS :* extraction automatique de termes-clés, méthodes non supervisées, représentation

*KEYWORDS:* keywords extraction, unsupervised method, representation

---

## 1. Introduction

Les outils et moteurs de recherche utilisent une représentation des documents pour permettre la recherche de l'information susceptible de répondre au besoin de l'utilisateur que celui-ci exprime au travers de sa requête. Cette représentation est construite pendant la phase d'indexation. Plusieurs formes d'indexation existent et parfois cohabitent dans les systèmes de classement et d'accès. L'*indexation manuelle analytique* vise à représenter des contenus à travers des termes-clés. Les documentalistes s'appuient sur un thésaurus et sur des règles d'indexation pour choisir les termes à associer à une ressource. Les auteurs des ressources sont également souvent appelés à associer des termes-clés aux documents qu'ils écrivent, en s'appuyant sur des hiérarchies de concepts ou en proposant des termes-clés libres. L'*indexation automatique* utilisée par les moteurs de recherche suit le processus suivant : extraction des mots du texte, suppression des mots-vides et outils de la langue, radicalisation pour limiter la variante des mots et pondération.

Quelle que soit la façon d'associer les termes-clés à un document, <sup>non supervisées de la littérature</sup> manuellement ou automatiquement, ils ont pour rôle de synthétiser au mieux le contenu des documents, soit pour aider à leur sélection à partir d'une requête qui contient ces termes-clés, soit pour aider l'utilisateur à choisir parmi un ensemble de documents ceux qui répondent le mieux à leur besoin. Depuis les années 2000, plusieurs travaux se sont intéressés à l'extraction automatique de termes-clés comme cela peut être fait manuellement. C'est à cette forme de termes-clés que s'intéresse ce présent article. Les méthodes de la littérature sont soit supervisées soit non supervisées.

Les méthodes supervisées, comme celle de Liu (Liu *et al*, 2011) et de Sarkar (Sarkar *et al*, 2010) ou encore celle de Witten (Witten *et al*, 1999), ont la capacité d'apprendre à partir d'exemples. Elles s'appuient sur une collection de documents déjà annotés en termes-clés dans la phase d'apprentissage. Cette contrainte nous a fait choisir les méthodes non supervisées. Dans les méthodes non supervisées, nous pouvons distinguer les méthodes s'appuyant sur une représentation sous forme de graphes comme TextRank (Mihalcea *et al*, 2004), SingleRank (Wan *et al*, 2008), TopicRank (Bougouin *et al*, 2013), Kcore et WKcore (Rousseau *et al*, 2015), de celles qui ne les utilisent pas comme les méthodes à base de regroupement ou de fréquence comme l'algorithme KeyCluster (Liu *et al*, 2009).

L'objectif de cet article est de comparer ces méthodes afin d'étudier leur complémentarité. Nous nous sommes centrés sur les méthodes non supervisées à base de graphe car elles sont plus facilement généralisables et ne nécessitent pas de phase d'apprentissage. Par ailleurs, un graphe présente de manière efficace et simple les mots d'un document et les relations qu'ils entretiennent. Nous utilisons la méthode TF-IDF comme référence. Dans la suite de cet article, la section 2 décrit les méthodes étudiées, la section 3 présente les résultats. La section 4 conclut cet article.

## 2. Etat de l'art

### 2.1. Termes-candidats

Les termes-candidats sont des unités textuelles pouvant devenir des termes-clés (groupes de mots). Dans nos expérimentations, c'est la méthode « les plus longues séquences de noms et d'adjectifs » qui a été utilisée pour l'extraction des termes candidats ; les méthodes citées ci-dessous sont appliquées à ces candidats.

### 2.2. TF-IDF

TF-IDF (ou Term Frequency – Inverse Document Frequency) (Sparck Jones, 1972) mesure le pouvoir discriminant d'un mot ou d'un groupe de mots dans un document donné. Essentiellement, cette technique mesure l'importance d'un certain terme dans un document par rapport aux autres documents de la même collection.

$TF - IDF(terme) = TF(terme) \times \log(\frac{N}{DF(terme)})$  avec : **TF** nombre d'occurrences du terme dans le document analysé ; **DF** nombre de documents



dans lequel le terme est présent et  $N$  nombre total de documents.

Cette mesure est utilisée pour pondérer les termes-candidats : plus la valeur TF-IDF d'un terme-candidat est élevée, plus celui-ci est important dans le document analysé. En prenant compte de tous les documents dans le corpus, cette méthode présente généralement de meilleurs résultats.

Les méthodes qui vont suivre sont des méthodes à base de graphe noté  $G(N, A)$  où  $N$  est l'ensemble des nœuds et  $A$  l'ensemble de ses arcs sortants et entrants. Chaque sommet du graphe représente un terme-candidat et la constitution des arêtes est propre à chaque méthode.

### 2.3. TextRank

TextRank est basée sur le calcul du score d'importance des sommets en utilisant le principe de vote ou de recommandation entre deux sommets (Mihalcea *et al*, 2004) et inspiré de l'algorithme Pagerank (Page *et al*, 1999). TextRank utilise une représentation efficace d'un document, elle peut aussi être utilisée pour faire des résumés automatiques d'un document. Malgré cela, elle possède un inconvénient : au lieu d'ordonner des termes-candidats, elle n'ordonne que des mots. Cette méthode repose sur les étapes suivantes : *construction du graphe, calcul des scores des sommets et extractions des termes clés*.

*Construction du graphe* : les termes-candidats initiaux sont des mots simples ; ils sont utilisés comme sommets du graphe. Deux sommets sont reliés par une arête s'ils co-occurrent dans une fenêtre de  $N$  mots.

*Calcul des scores des sommets* : Au départ, les scores de tous les sommets du graphe sont initialisés aléatoirement. Un algorithme de classement calcule les scores de chaque sommet à chaque itération et s'arrête lorsque le seuil donné est atteint. Le score d'un sommet  $C_i$  est calculé avec la formule  $S(C_i) = (1 - d) + d * \sum_{C_j \in In(C_i)} \frac{1}{|Out(C_j)|} S(C_j)$  avec :  $In(C_i)$  l'ensemble des sommets qui pointent vers  $C_i$  ;  $Out(C_j)$  l'ensemble des sommets que pointent  $C_j$  et  $d$  un facteur d'amortissement<sup>1</sup>. Dans le cas où le graphe  $G$  serait non orienté,  $In(C_i) = Out(C_i)$ .

La formule suivante qui est utilisée dans le cas où les arêtes reliant les sommets sont pondérées :  $WS(C_i) = (1 - d) + d * \sum_{C_j \in In(C_i)} \frac{w_{ji}}{\sum_{C_k \in Out(C_j)} w_{jk}} WS(C_j)$

*Extraction des termes-clés* : C'est à partir des sommets les plus importants (selon leur score par ordre décroissant) que sont choisis les mots-clés. Les séquences des mots (des sommets importants) adjacents dans le document constituent les termes-clés composés de plusieurs mots; les autres mots non adjacents dans les documents qui obtiennent les meilleurs scores sont également retenus comme mots-clés.

### 2.4. SingleRank

---

<sup>1</sup> Peut être défini entre 0 et 1

SingleRank est une modification de la méthode <sup>non supervisées de la littérature</sup> TextRank, la différence se trouve dans la pondération des arêtes du graphe de mots et dans l'extraction des mots-clés à partir des mots-candidats, c'est-à-dire le calcul des scores des termes (Wan et al, 2008). Dans la majorité des cas, cette méthode fournit de meilleurs résultats que TextRank. L'inconvénient de cette méthode est qu'elle favorise les termes-candidats les plus longs (c'est-à-dire les termes formés par de nombreux mots) tout en faisant monter les candidats redondants dans le classement. Cette méthode, comme TextRank, repose sur trois étapes : *construction du graphe, calcul des scores des sommets et extractions des termes clés*.

*Construction du graphe* : les terme-candidats peuvent être composés et chaque mot composant chaque terme-candidat est considéré comme sommet du graphe. Deux sommets sont reliés par une arête s'ils co-occurrent dans une fenêtre de N mots et le poids de cette arête est le nombre de cooccurrence de ces deux sommets.

*Calcul des scores des sommets* : SingleRank utilise un algorithme de classement pour calculer les scores des sommets avec la formule suivante :

$$S(C_i) = (1 - \lambda) + \lambda \times \sum_{C_j \in A_{entrant}(C_i)} \frac{p_{j,i} \times S(C_j)}{\sum_{C_k \in A_{sortant}(C_j)} p_{j,k}}$$

Avec :  $\lambda$  : facteur d'atténuation (peut être considéré comme la probabilité que le nœud  $C_i$  soit atteint par recommandation) ;  $p_{j,i}$  : poids de l'arc allant du nœud  $C_j$  vers le nœud  $C_i$ , correspondant au nombre de cooccurrences entre les deux mots  $i$  et  $j$ .

Les scores des termes-candidats constitués de plusieurs mots sont calculés par :  $TermeScore(p_i) = \sum S(C_j)$  Avec :  $p_i$  est un terme-candidat constitué de plusieurs mots ;  $C_j$  est un des mots composant le terme-candidat  $p_i$

*Extraction des termes-clés* : Ce sont les termes-candidats ayant les scores les plus importants qui sont retenus comme termes-clés. Il n'y a donc pas de génération de termes-clés comme cela est le cas dans TextRank.

## 2.5. TopicRank

TopicRank, fondée sur TextRank, est différente par rapport aux autres méthodes à base de graphe, parce qu'au lieu de faire une recherche des unités textuelles importantes du document, elle cherche ses sujets importants. Par rapport à TextRank et SingleRank, elle présente les avantages suivants : suppression des problèmes de redondance dans les termes-clés extraits, construction d'un graphe plus compacte, renforcement des poids des arêtes dans le graphe, amélioration de la qualité d'ordonnancement et suppression du paramètre de la fenêtre de co-occurrences. Cette méthode repose sur trois étapes : *identification des sujets, ordonnancement des sujets, sélection des mots-clés* (Bougouin et al, 2013).

*Construction du graphe* : ce sont les termes-candidats, composés de plusieurs mots ou non, qui représentent les sommets du graphe et tous les sommets sont reliés entre eux, nous avons un graphe complet.

*Identification des sujets* : un sujet est une information spécifique (le plus souvent) ou générale transportée au minimum par une unité textuelle. Deux termes-candidats  $C_1$  et  $C_2$  sont groupés à partir de la similarité de Jaccard :  $sim(C_1, C_2) = \frac{\|C_1 \cap C_2\|}{\|C_1 \cup C_2\|}$

Avec :  $C_1 \cup C_2$  : nombre de mots commun à  $C_1$  et  $C_2$ ;  $C_1 \cap C_2$  : nombre de mots composant  $C_1$  et  $C_2$ .

Dès que la similarité entre toutes les paires de termes-candidats est connue, l'algorithme de classification ascendante hiérarchique est appliqué. Au début, chaque terme-candidat est considéré comme un groupe et puis les deux groupes présentant la plus forte similarité sont réunis en un seul. Ce regroupement est répété jusqu'à ce que le nombre (prédéfini) de groupes soit atteint. La similarité entre deux groupes est obtenue en calculant la similarité entre les termes-candidats composant chaque groupe. La valeur de similarité entre deux groupes peut être obtenue en choisissant parmi les trois méthodes suivantes : (*simple*) : la plus grande valeur de similarité est retenue ; (*complète*) : la plus petite valeur de similarité est retenue ; (*moyenne*) : la moyenne de toutes les similarités est retenue

Une fois les sujets connus, un nouveau graphe est défini comme suit :  $G = (N, A)$  :  $N$  = ensemble des sujets du document et  $A$  = ensemble des liens entre les nœuds.

*Ordonnancement des sujets* : la pondération des arêtes est très importante durant cette étape. C'est la force du lien sémantique (Wan and Xiao, 2008) entre les nœuds du graphe (ou plutôt les sujets) qui est considérée comme poids d'une arête.

Pour représenter cette force sémantique, la distance entre les termes-candidats des sujets est utilisée.  $poids(S_i, S_j) = \sum_{C_i \in S_i} \sum_{C_j \in S_j} dist(C_i, C_j)$  avec  $dist(C_i, C_j) = \sum_{p_i \in pos(C_i)} \sum_{p_j \in pos(C_j)} \frac{1}{|p_i - p_j|}$  Avec :  $poids(S_i, S_j)$  est le poids de l'arête entre les sujets  $S_i$  et  $S_j$ ;  $dist(C_i, C_j)$  est la force sémantique entre les termes-candidats  $C_i$  et  $C_j$ ;  $pos(C_i)$  est la position du terme candidat  $C_i$  dans le document analysé et  $pos(C_j)$  est la position du terme candidat  $C_j$  dans le document analysé.

L'ordonnancement est basé sur le principe de vote ou de recommandation, c'est à dire qu'un nœud (sujet) est très important s'il est fortement connecté avec plusieurs autres nœuds importants.

$$importance(S_i) = (1 - \lambda) + \lambda \times \sum_{S_j \in V_i} \frac{poids(s_i, s_j) \times importance(s_j)}{\sum_{S_k \in V_j} poids(s_j, s_k)} \text{ Avec :}$$

$V_i$  est l'ensemble des sujets reliés au sujet  $S_i$  et  $\lambda$  est le facteur d'atténuation.

*Sélection des termes-clés* : chaque sujet important ne va fournir qu'un seul terme-clé. Pour le choix d'un terme-clé représentant le mieux un sujet, trois méthodes ont été proposées :

- *Première position* : le terme-candidat d'un sujet apparaissant le premier dans le document est sélectionné,
- *Fréquence* : le terme-candidat d'un sujet le plus fréquent dans le document analysé est sélectionné,

– *Centroïde* : le terme-candidat d'un sujet <sup>non supervisées de la littérature</sup> le plus similaire aux autres mot-candidat du même sujet est sélectionné.

## 2.6. Kcore

Kcore (Rousseau *et al*, 2015) est aussi une méthode d'extraction de termes-clés à base de graphe. La construction de son graphe de mots est semblable à celle de TextRank ou SingleRank. Contrairement aux méthodes à base de graphe que nous avons présentées ci-dessus, elle n'utilise pas le principe de vote ou de recommandation pour calculer les scores d'importance des sommets mais utilise l'algorithme de Batagelj et Zaveršnik (Batagelj *et al*, 2011). Le problème est qu'elle dépend de la fenêtre de co-occurrence de mots.

*Construction du graphe* : ce sont les termes-candidats, constitués de plusieurs mots ou non, qui représentent les sommets du graphe et deux sommets sont reliés si les termes-candidats représentant ces sommets co-occurent dans une fenêtre de N mots. Ces liens peuvent être pondérés par le nombre de co-occurrences des mots qu'ils relient dans le document, on parle de WKcore, sinon (dans le cas d'un graphe non pondéré) ils seront tous pondérés par 1, on parle de Kcore.

Le degré d'un sommet  $C \in G$  dans  $G$  est noté  $deg_G(C)$ . Autrement dit, dans le cas où  $G$  est un graphe non orienté,  $deg_G(C)$  est la somme des poids des arêtes adjacentes à  $C$  (poids unitaire dans le cas d'un graphe non pondéré). Dans le cas où  $G$  est orienté, la notion de degré se divise en deux : degré sortant et degré entrant qui correspondent respectivement au nombre de liens sortants et de liens entrants.

Un Kcore est un sous graphe  $H_k = (v', \varepsilon')$  d'un graphe  $G = (N, A)$  où  $v' \subseteq N$  et  $\varepsilon' \subseteq A$  tel que  $\forall v \in N, deg_{H_k}(v) \geq k$  et  $H_k$  est le sous-graphe maximal, c'est à dire qu'il ne peut pas être augmenté sans perdre cette propriété (Rousseau *et al*, 2015).

Une fois le graphe de mots construit, une décomposition Kcore avec l'algorithme de Batagelj et Zaveršnik (Batagelj *et al*, 2011) du graphe est réalisée. L'algorithme attribut un nombre  $n$  à chaque sommet du graphe Kcore auquel il appartient (si  $v \in Kcore$  alors  $n = k$ ). Puis l'ordonnancement des sommets se fait par ordre décroissant des nombres  $n$  et les premiers sont retenus comme termes-clés.

## 3. Expérimentations et résultats

### 3.1. Cadre expérimental

Nous utilisons deux collections afin d'étudier et de comparer les méthodes présentées dans la section 2. Nous avons construit ces collections en interrogeant le Web of Sciences. La collection que nous appelons IPM correspond aux 689 articles publiés entre 1975-2015 dans le journal *Information Processing and Managment* ; il y a en moyenne 5,7 mots-clés par documents associés par les auteurs. La collection IRJ est constituée de 344 articles publiés entre 2000-2015 dans le journal *Information Retrieval Journal*; il a en moyenne 4,7 mots-clés.

Pour analyser les résultats des différentes méthodes, nous comparons les termes-clés extraits automatiquement et ceux associés manuellement par les auteurs. Nous mesurons l'efficacité des méthodes d'extraction par le Rappel<sup>2</sup> (R) lorsque l'on considère les 15 premiers termes. Cette mesure est définie par :

$$R = \frac{\text{Nombre}_{\text{match}}}{\text{Nombre}_{\text{ref}}}$$
 : avec :  $\text{Nombre}_{\text{match}}$  nombre de termes-clés générés identiques à ceux formulés par les auteurs,  $\text{Nombre}_{\text{result}}$  nombre total de termes-clés générés automatiquement et  $\text{Nombre}_{\text{ref}}$  nombre total de termes-clés formulés par les auteurs.

Une comparaison entre les termes-clés fournis par les auteurs et les termes-clés extraits à partir des méthodes implémentées est réalisée. Dans les expérimentations, nous considérons les mots soit tels qu'ils apparaissent dans les documents, soit après racinisation<sup>3</sup> par l'algorithme de Martin Porter (Porter, 1980). Nous avons implémenté toutes ces méthodes et les avons comparées en s'inspirant du code de Bougouin (disponible sur GitHub<sup>4</sup>).

### 3.2. Influence des autres paramètres des méthodes

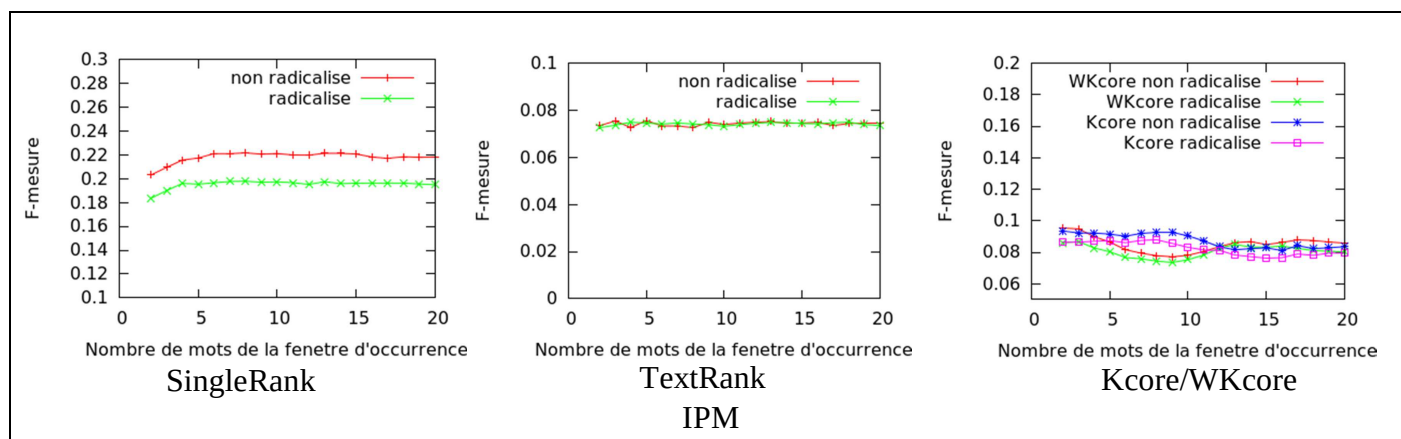


**Figure 1.** Résultats de l'extraction avec SingleRank, TextRank, Kcore et WKcore en fonction de la fenêtre de cooccurrences utilisée, avec et sans racinisation.

Le paramètre commun à TextRank, SingleRank, Kcore et WKcore est N, le nombre de mots dans la fenêtre de cooccurrences que nous faisons varier.

La figure 1 présente les résultats de SingleRank, TextRank, Kcore et WKcore sur le corpus IPM lorsque nous faisons varier la fenêtre de cooccurrence. Les meilleurs résultats sont obtenus par SingleRank. Pour SingleRank et TextRank, les résultats sont assez stables quelle que soit la valeur de N. La radicalisation diminue les résultats pour SingleRank alors qu'elle n'a que peu d'effet sur TextRank. En revanche, le paramètre N a plus d'influence sur les méthodes Kcore et WKcore. Kcore obtient de meilleurs résultats que WKcore pour N entre 6 et 11 alors que

<sup>2</sup> Rapport des résultats pertinents parmi les résultats retrouvés.

<sup>3</sup> Transformation d'un mot en sa racine c'est-à-dire suppression de ses suffixes et préfixes.

<sup>4</sup> [https://github.com/adrien-bougouin/KeyBench/tree/ijcnlp\\_2013](https://github.com/adrien-bougouin/KeyBench/tree/ijcnlp_2013)

WKcore devient meilleur lorsque N est plus important. Nous obtenons les mêmes résultats avec la collection IRJ.

Le paramètre de la méthode TopicRank est le seuil de similarité, la meilleure stratégie de groupement des termes-candidats et la meilleure stratégie de sélection du terme-candidat le plus représentatif d'un sujet. Nous avons fait varier le seuil de similarité  $\zeta$  d'un pas de 0,10 pour toutes les stratégies de groupement (cf. figure 2).

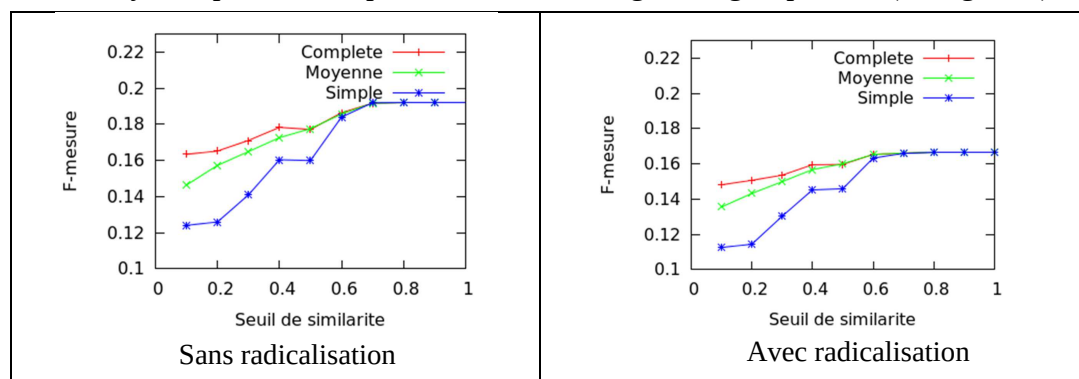


**Figure 2.** Résultats de l'extraction de mots-clés avec TopicRank, en fonction de la stratégie de regroupement et de la valeur du seuil de similarité

Les trois stratégies de regroupement ont chacune un comportement qui leur est propre jusqu'à un premier point de convergence dès que  $\zeta > 0,50$ , ce point de convergence indique que plus d'un mot sur deux sont identiques dans les deux termes-candidats comparés. Le second point de convergence  $\zeta = 0,70$  correspond à strictement plus de deux mots sur trois sont identiques dans les deux termes-candidats comparés. Une fois passé ce deuxième point, les résultats sont stables ; cela correspond à la rareté des termes-clés composés de plus de 4 mots. Nous obtenons la même valeur de convergence pour la collection IRJ. Les meilleurs résultats sont obtenus en utilisant la stratégie complète, avec ou sans radicalisation.

Avec ou sans radicalisation, c'est le choix des candidats apparaissant en premier dans le document offre de meilleurs mots-clés que le choix des candidats centroïdes ou des candidats les plus fréquents ce qui confirme les résultats de Bouguoin (Bouguoin *et al*, 2013). La stratégie centroïde fournit de très faibles résultats par rapport aux deux autres stratégies.

### 3.3. Comparaison des différentes méthodes

Les valeurs des mesures dans le tableau suivant sont les moyennes des mesures obtenues sur l'ensemble les documents. Toutes les méthodes sont configurées de façon optimale c'est-à-dire dans l'optique d'obtenir les meilleurs termes-clés. Nous allons voir ainsi quelle est la méthode qui convient le mieux à notre cas et si la radicalisation des mots a une influence positive sur ces méthodes.

| Méthodes | TF-IDF | TopicRank | TextRank | SingleRank | Kcore | WKcore |
|----------|--------|-----------|----------|------------|-------|--------|
|----------|--------|-----------|----------|------------|-------|--------|

|                  |                |                |                |                |                |                |
|------------------|----------------|----------------|----------------|----------------|----------------|----------------|
| <b>Rappel@5</b>  | 0.13<br>(0.11) | 0.13<br>(0.11) | 0.03<br>(0.03) | 0.11<br>(0.10) | 0.03<br>(0.03) | 0.04<br>(0.04) |
| <b>Rappel@10</b> | 0.19<br>(0.17) | 0.16<br>(0.14) | 0.05<br>(0.05) | 0.18<br>(0.16) | 0.06<br>(0.06) | 0.06<br>(0.06) |
| <b>Rappel@15</b> | 0.22<br>(0.20) | 0.19<br>(0.17) | 0.08<br>(0.07) | 0.22<br>(0.20) | 0.09<br>(0.09) | 0.10<br>(0.09) |

**Tableau 1.** Résultats de l'extraction des 5-10-15 mots-clés de plus fort poids selon les différentes méthodes, sans radicalisation et (avec radicalisation)

TF-IDF fournit de meilleurs résultats que les autres méthodes, avec ou sans radicalisation. Il est aussi important à noter que c'est SingleRank qui fournit les meilleurs résultats parmi les méthodes basées sur un graphe de mots.

#### 4. Conclusion

Nous avons constaté que globalement, TF-IDF fournit les mots-clés les plus proches de ceux formulés par les auteurs des articles ; ceci est possiblement dû au fait que cette méthode utilise tous les documents composant le corpus. Mais SingleRank, qui n'utilise que le document analysé, est l'une des meilleures méthodes d'extraction automatiques de mots-clés : ses performances ne sont pas éloignées de celles de TF-IDF. SingleRank serait d'autant plus performante qu'elle est utilisée sur un document complet au lieu d'un document composé seulement du titre et de son résumé (comme dans notre cas). L'utilisation de la radicalisation sur les documents avant l'extraction des mots-clés ne fournit pas de meilleurs résultats donc son intégration ne s'avère pas être utile dans notre analyse. Nous souhaitons maintenant confirmer sur de plus grand corpus de documents. Nous souhaitons également analyser la variation des valeurs des paramètres en fonction des collections et voir si les paramètres optimisés sur cette collection sont les mêmes sur d'autres collections.

La comparaison des performances en précision de ces méthodes d'extraction automatiques de mots-clés sont des comparaisons simples. Des tests statistiques seraient à utiliser dans le but de savoir si la différence de performance entre deux méthodes est significative ou non, surtout entre les méthodes TF-IDF et SingleRank.

Nous nous sommes intéressés à comparer les résultats de d'extraction automatique avec les mots choisis par les auteurs; ces méthodes pourraient également être utilisées pour suggérer des mots-clés aux auteurs.

#### 5. Bibliographie

Batagelj V., Zaveršnik M., *Fast algorithms for determining (generalized) core groups in social networks*, Advances in Data Analysis and Classification, vol. 5, n° 2, 2011, p. 129-145.

Bougouin A., *Etat de l'art des méthodes d'extraction automatique de termes-clés*, In Rencontre des Etudiants Chercheurs en Informatique pour le Traitement Automatique des Langues (RECITAL), 2013.

Bougouin A., Boudin F., and Daille B., *Topicrank: Graph-based topic ranking for keyphrase extraction*, In International Joint Conference on Natural Language Processing (IJCNLP), 2013. p. 543-551.

Brin S., Page L., *The anatomy of a large-scale hypertextual web search engine*, Computer networks and ISDN systems, vol. 30, n° 1, p. 107-117.

Liu Z., Chen X., Zheng Y., Sun M., *Automatic keyphrase extraction by bridging vocabulary gap*, In Proceedings of the Fifteenth Conference on Computational Natural Language Learning, Association for Computational Linguistics, 2011 p. 135-144.

Liu Z., Li P., Zheng Y., Sun M., *Clustering to find exemplar terms for keyphrase extraction*, In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Volume 1-Volume 1, 2009, p. 257-266.

Mihalcea R., Tarau P., *Textrank: Bringing order into texts*, Association for Computational Linguistics, 2004.

Page L., Brin S., Motwani R., Winograd T., *The PageRank citation ranking: bringing order to the Web*, 1999.

Porter M. F., *An algorithm for suffix stripping*, Program, vol. 14, n° 3, 1980, p. 130-137.

Rousseau F., Vazirgiannis M., *Main core retention on graph-of-words for single-document keyword extraction*, In Advances in Information Retrieval, 2015, p. 382-393.

Sarkar K., Nasipuri M., Ghose S., *A new approach to keyphrase extraction using neural networks*, arXiv preprint arXiv:1004.3274, 2010.

Sparck Jones K., *A statistical interpretation of term specificity and its application in retrieval*, Journal of documentation, vol. 28, n° 1, 1972, p. 11-21.

Wan X., Xiao J., *Single document keyphrase extraction using neighborhood knowledge*, In AAAI, vol. 8, 2008, p. 855-860.

Witten I. H., Paynter G. W., Frank E., Gutwin C., Nevill-Manning C. G., (1999, August). *KEA: Practical automatic keyphrase extraction*, In Proceedings of the fourth ACM conference on Digital libraries, ACM, p. 254-255.



---

# ANASTASIA : recommandation de séquences d'activités spatio-temporelles

**Diana Nurbakova\*** — **Léa Laporte\*** — **Sylvie Calabretto\*** — **Jérôme Gensel\*\***

\* *Laboratoire d'InfoRmatique en Image et Système d'information (LIRIS)*  
*UMR 5205 CNRS/INSA de Lyon*

*20 avenue Albert Einstein, 69621 Villeurbanne cedex*

\*\* *Laboratoire d'Informatique de Grenoble (LIG)*

*UMR 5217 CNRS/Université Grenoble Alpes*

*BP 53, 38041 Grenoble cedex 9*

---

*RÉSUMÉ. Avec l'augmentation du nombre et de la variété des activités accessibles par les utilisateurs, la recommandation personnalisée de séquences d'activités devient un enjeu important. Or, la plupart des systèmes de recommandation ne tiennent pas compte des contraintes temporelles liées aux activités, ce qui rend la recommandation difficile à suivre par un utilisateur. Dans cet article, nous décrivons une nouvelle approche pour la recommandation de séquences d'activités limitées dans le temps et concurrentes. Elle s'appuie sur l'historique des activités des utilisateurs pour extraire des motifs comportementaux et intègre différents éléments contextuels (popularité, informations démographiques et spatio-temporelles). Nous présentons un protocole d'évaluation et un jeu de données qui permettra l'évaluation de notre approche.*

*ABSTRACT. As amount of activities available for users and their variety have grown, personalised recommendation of activities sequences has become an important challenge. However, most of recommender systems do not consider temporal constraints of activities, making the recommendation hard for user to follow. In this article, we describe a novel approach for recommendation of competing activities limited in time. It makes use of historical records of users' activities in order to mine users' behavioral patterns, and combines different contextual elements (popularity, demographic and spatio-temporal information). We present an evaluation framework and a dataset that will allow us to evaluate our approach.*

*MOTS-CLÉS : système de recommandation, séquence d'activités, score personnalisé, contexte d'utilisateur.*

*KEYWORDS: recommender system, activities sequence, personalised score, user context.*

---

## 1. Introduction

Au cours des dernières années, les services et les réseaux sociaux basés sur la localisation (*LBSNs*) ont attiré des millions d'utilisateurs. Une des fonctionnalités qu'ils proposent est la recommandation de points d'intérêt (*POIs*). Un *POI* est une position physique correspondant à une localisation géographique d'une personne ou d'un événement et présentant un intérêt spécifique pour un utilisateur<sup>1</sup>. Les systèmes de recommandation de *POIs* proposent à l'utilisateur un ensemble de *POIs*, ordonnés selon le degré d'intérêt de l'utilisateur. Récemment, des travaux (Sang *et al.*, 2015 ; Yu *et al.*, 2015) se sont intéressés au problème de la recommandation de séquences de *POIs*. Un inconvénient de ces approches est qu'elles ne tiennent pas compte des contraintes temporelles associées aux *POIs* (*e.g.* horaires d'ouverture-fermeture).

Dans cet article, nous nous concentrons sur la recommandation de séquences d'activités spatio-temporelles, que nous nommons R-STAS et qui s'applique à plusieurs scénarios : événements distribués (conférences, festivals ou expositions) ; tourisme (recommandation d'animations pendant une croisière) ; crowdsourcing mobile. Nous définissons la notion d'*activité* comme un événement géo-localisé à durée limitée et associé à une ou plusieurs catégories.

Parmi les travaux sur les systèmes de recommandation, il y a très peu d'approches qui exploitent conjointement les centres d'intérêt des utilisateurs, l'ensemble des facteurs spatio-temporels et les habitudes de l'utilisateur afin de construire un itinéraire personnalisé des activités. La motivation du travail actuel peut être formulée selon les aspects suivants : (1) *Personnalisation et Contexte* : accorder les activités aux intérêts de l'utilisateur et son contexte ; (2) *Espace et Temps* : prendre en compte le caractère spatial et la disponibilité limitée des activités ; (3) *Séquence et Succession* : gérer de façon optimale les séquences d'activités concurrentes et interdépendantes.

Nous proposons une approche qui prend en compte les contraintes et dépendances temporelles entre activités au sein d'une séquence. En s'appuyant sur les historiques d'activités de l'utilisateur et sur ses centres d'intérêts, nous récupérons les habitudes, ou motifs de comportement, de l'utilisateur et calculons le score de pertinence des activités pour chaque utilisateur. Ensuite, nous utilisons les résultats obtenus et les horaires de disponibilité des activités pour construire une séquence d'activités. Afin d'évaluer l'approche proposée et en l'absence de collection adaptée, nous proposons un protocole d'évaluation et la construction d'un jeu de données.

L'article est organisé de la manière suivante. Nous discutons des travaux liés à la section 2. La section 3 décrit l'approche proposée. Dans la section 4 nous proposons une discussion sur le protocole d'évaluation. Enfin, la section 5 conclut l'article.

---

1. [http://www.w3.org/2010/POI/wiki/Main\\_Page](http://www.w3.org/2010/POI/wiki/Main_Page)

## 2. Travaux liés

Les travaux de l'état-de-l'art touchent deux domaines en recommandation : la recommandation de POIs et la recommandation de séquences de POIs. La recommandation de POIs considère des éléments indépendants restitués sous la forme d'une liste ordonnée selon la pertinence aux centres d'intérêt de l'utilisateur, tandis que la recommandation de séquences de POIs gère l'interdépendance géographique des éléments.

La plupart des algorithmes de recommandation de POIs supposent généralement l'indépendance entre les POIs et se basent sur le principe de l'ordonnement des POIs selon l'intérêt de l'utilisateur. Les centres d'intérêt sont indiqués explicitement dans le profil de l'utilisateur ou récupérés à partir des POIs visités, via leur position géographique, la fréquence de visite (*check-in*) ou leur popularité. Ils sont utilisés pour calculer un score de pertinence des POIs. Les POIs sont ensuite ordonnés selon le score calculé et les tops  $k$  résultats sont proposés à l'utilisateur (Zhang et Chow, 2015a ; Yang et Fang, 2014 ; Li *et al.*, 2015 ; Yang et Fang, 2014 ; Zhang et Chow, 2015b). Ce type d'algorithmes de recommandation exploite des combinaisons de différents facteurs (influence géographique, évolution des goûts de l'utilisateur, etc.), mais ne tient compte ni de l'aspect consécutif des visites, ni de la disponibilité temporelle limitée des POIs.

La recommandation de séquences de POIs permet la prise en compte de l'aspect consécutif des visites. Nous pouvons distinguer deux types d'approches. Dans la première approche, majoritairement présente dans la littérature, la recommandation de séquences de POIs est divisée en deux sous-problèmes : l'estimation des centres d'intérêt de l'utilisateur pour calculer un score de pertinence personnalisé pour chaque POI et la résolution d'un problème d'optimisation pour la construction d'un itinéraire optimal (par exemple, (Yu *et al.*, 2015 ; Zhang *et al.*, 2015 ; Schaller *et al.*, 2014b)). Le calcul de l'itinéraire est généralement effectué à l'aide d'algorithmes d'optimisation sous contraintes issus du domaine de la recherche opérationnelle (Vansteenwegen *et al.*, 2011 ; Gavalas *et al.*, 2014). Dans la deuxième approche, les algorithmes déterminent une séquence de POIs en calculant la probabilité de transition d'un POI à un autre (Sang *et al.*, 2015). Cependant, ces approches ne tiennent pas compte de la disponibilité temporelle limitée des POIs.

Dans nos travaux, nous nous focaliserons sur la recommandation de séquences d'activités spatio-temporelles prenant en compte les contraintes et dépendances temporelles entre activités et les habitudes des utilisateurs.

## 3. Approche Proposée

Nous proposons une approche de recommandation de séquences d'activités, nommée ANASTASIA (« A Novel Approach for Spatio-Temporal Activities Sequences and Itineraries recommendAtion »). Nous adaptons des algorithmes de l'état de l'art pour tenir compte des centres d'intérêts de l'utilisateur et des contraintes et dépendances séquentielles, temporelles et spatiales. Notre approche est composée de 3 mo-

dules : un module d'extraction des motifs de séquences, un module de calcul des scores des activités et un module de construction d'itinéraires. Les données sont traitées pour extraire des motifs caractéristiques dans les historiques des utilisateurs et pour calculer des scores de pertinence pour chaque activité. Ces motifs et scores sont ensuite intégrés dans le calcul de l'itinéraire.

### 3.1. Module d'extraction des motifs de séquences

Le module d'extraction des motifs de séquences cherche à apprendre le comportement typique de l'utilisateur en récupérant des séquences d'activités consécutives. Deux activités sont considérées consécutives si l'intervalle temporel entre la fin de la première activité et le début de la deuxième est inférieur à un seuil fixé.

Nous proposons d'adapter les travaux de (Zhang *et al.*, 2014) sur l'extraction de motifs de séquences. (Zhang *et al.*, 2014) utilisent un graphe de transition localisation-localisation ( $L^2TG$ ) pour modéliser les séquences de POIs visités par l'utilisateur. Nous étendons ce principe en considérant deux graphes de transition : un graphe de transition activité-activité, noté  $A^2TG$ , et un graphe de transition catégorie-catégorie, noté  $C^2TG$ . Le premier graphe permet de modéliser les transitions entre activités uniques (e.g. concert avec date et lieu fixé). Le deuxième graphe permet une généralisation en modélisant les transitions entre catégories d'activités (e.g. spectacles).

Nous nous appuyons sur une liste d'activités antérieures de l'utilisateur et sur les horaires de ces activités pour construire le graphe  $A^2TG$ . Les nœuds du graphe, notés  $V = \{a_1, \dots, a_N\}$ , correspondent aux activités. Contrairement à (Zhang *et al.*, 2014) qui utilisent le nombre d'arcs sortants du nœud, nous proposons d'utiliser le nombre d'arcs entrants, noté  $InCount(a_i)$ . Cette modification est motivée par l'hypothèse que la satisfaction de l'utilisateur dépend des activités précédentes recommandées et réalisées. Par exemple, il n'est pas pertinent de recommander un restaurant à un utilisateur qui vient de manger dans une pizzeria. Les arcs du graphe reflètent les transitions d'une activité à une autre. Ils sont pondérés par le nombre de transitions d'une activité à une autre, que nous notons  $TransCount(a_i \rightarrow a_j)$ . La probabilité de transition d'une activité à une autre, notée  $P_T(a_i \rightarrow a_j)$ , est définie comme suit :

$$P_T(a_i \rightarrow a_j) = \begin{cases} \frac{TransCount(a_i \rightarrow a_j)}{InCount(a_j)}, & \text{si } InCount(a_j) \neq 0 \\ 0, & \text{si } InCount(a_j) = 0 \text{ et } a_i \neq a_j \\ 1, & \text{si } InCount(a_j) = 0 \text{ et } a_i = a_j \end{cases}$$

Notons que les activités que nous considérons sont uniques. Dans ce cas, la probabilité de transition  $P_T(a_i \rightarrow a_j)$  devient binaire :  $P_T(a_i \rightarrow a_j) = 1$  si la transition d'une activité à une autre existe, et  $P_T(a_i \rightarrow a_j) = 0$  sinon. Ainsi, nous proposons de passer au niveau hiérarchique supérieur, c'est-à-dire au niveau des catégories d'activités. Nous supposons que l'attribution des activités aux catégories est connue.

Pour construire le graphe  $C^2TG$  nous passons par une étape intermédiaire de substitution des activités par leurs catégories, puis nous appliquons le même principe que pour le graphe  $A^2TG$ . Les nœuds du graphe  $C^2TG$  représentent les catégories

correspondant aux activités effectuées par l'utilisateur et sont associés au nombre d'arcs entrants dans le nœud  $c_i$ , noté  $InCount(c_i)$ .  $InCount(c_i)$  est fonction des activités composant la catégorie  $c_i$ , tel que :  $InCount(c_i) = \sum_{a_j \in c_i} InCount(a_j)$ .

Les arcs correspondent aux transitions d'une catégorie à une autre et sont pondérés par le nombre de transitions,  $TransCount(c_i \rightarrow c_j)$ , défini comme la somme des valeurs de  $TransCount$  des activités de cette paire de catégories :  $TransCount(c_i \rightarrow c_j) = \sum_{a_k \in c_i, a_g \in c_j} TransCount(a_k \rightarrow a_g)$ . La probabi-

lité de transition d'une catégorie à une autre est calculée de la façon suivante :

$$P_T(c_i \rightarrow c_j) = \begin{cases} \frac{TransCount(c_i \rightarrow c_j)}{InCount(c_j)}, & \text{si } InCount(c_j) \neq 0 \\ 0, & \text{si } InCount(c_j) = 0 \text{ et } c_i \neq c_j \\ 1, & \text{si } InCount(c_j) = 0 \text{ et } c_i = c_j \end{cases}$$

En s'appuyant sur la liste d'activités effectuées par l'utilisateur, nous proposons d'explorer l'importance d'une activité au sein d'une séquence, en calculant son poids, noté  $Weight(a_i)$ . L'objectif est de comprendre si une activité a été effectuée uniquement pour patienter entre deux activités ou si elle a présenté un intérêt pour l'utilisateur. Ce poids est basé sur la probabilité de choisir l'activité au hasard dans la collection des activités concurrentes. Elle est calculée de la manière suivante. Parmi les motifs de séquences d'utilisateur, nous choisissons les sous-séquences composées de 3 éléments, c'est-à-dire pour l'activité  $a_i$  nous cherchons une sous-séquence :  $a_{i-1} \rightarrow a_i \rightarrow a_{i+1}$ . Étant donnée la liste d'activités avec leurs fenêtres temporelles et leur durée, nous proposons de chercher toutes les activités,  $a_j \in Concurrents(a_i)$ , concurrentes à l'activité  $a_i$ . La probabilité de choix de l'activité  $a_i$  au hasard est calculée comme un ratio :  $PH(a_i) = \frac{1}{|Concurrents(a_i)|}$ , où  $|\cdot|$  est la cardinalité de l'ensemble. Le poids de l'activité est donnée par :  $Weight(a_i) = 1 - PH(a_i)$ .

### 3.2. Module de calcul des scores des activités

Le module de calcul des scores des activités vise à mesurer la pertinence d'une activité donnée par rapport aux centres d'intérêts d'un utilisateur. Nous proposons de calculer le score des activités en explorant différentes dimensions (spatiale, personnelle, sociale, financière) du contexte de l'utilisateur.

Chaque activité est associée à la localisation. Nous supposons que dans le cadre de la recommandation d'activités spatio-temporelles, la proximité géographique a moins d'influence que dans le cas de la recommandation de POIs et que les contraintes temporelles sont plus fortes et englobent une partie des contraintes géographiques. Nous intégrons le score spatial, noté  $Score_{spatial}(a_i)$ , dans le score global de pertinence d'une activité. Ce score explore l'influence de la distance entre les activités afin de récupérer la préférence de l'utilisateur à se déplacer ou à rester au même endroit.

En nous appuyant sur les profils des utilisateurs, nous proposons de construire des vecteurs d'informations personnelles pour chaque utilisateur qui contiennent les attributs suivants : classes d'âges (<13, 14-17, 18-20, 21-30, 31-50, >50), genre (féminin,

masculin), nombre de participations aux éditions d'un événement ou à des événements similaires (0 fois, 1 fois, 2-5 fois, >5 fois). Les valeurs des attributs sont binaires. Ainsi, nous proposons de récupérer les fréquences de visites des activités par rapport aux attributs décrits et de calculer le score personnel  $Score_{pers}(a_i)$  de l'activité.

Nous proposons d'explorer l'influence sociale en nous basant sur la popularité des activités et les centres d'intérêt de l'utilisateur qui changent en fonction du groupe qui l'accompagne (seul, en couple, avec des amis, avec la famille, autre). Nous notons cette influence le score social,  $Score_{social}(a_i)$ , et l'intégrons dans le score global.

Certaines activités exigent des frais, d'autres sont gratuites. Les utilisateurs peuvent avoir des contraintes financières, identifiant le budget maximal à ne pas dépasser. Nous proposons d'explorer la préférence de l'utilisateur à exercer les activités qui exigent ou n'exigent pas des frais en se basant sur la fréquence de ces activités. Ainsi nous obtenons le score financier,  $Score_{finance}(a_i)$ .

Le score de pertinence de l'activité,  $Score(a_i)$ , sera ainsi fonction des différentes dimensions du contexte de l'utilisateur, c'est-à-dire :  $Score(a_i) = \alpha \cdot Score_{spatial}(a_i) + \beta \cdot Score_{pers}(a_i) + \gamma \cdot Score_{social}(a_i) + \delta \cdot Score_{finance}(a_i)$ . À notre connaissance, il n'existe pas, d'approche qui exploite toutes les dimensions envisagées. (Sang *et al.*, 2015) utilisent les facteurs personnels de l'utilisateur, les relations temporelles et géographiques entre les POIs, tandis que (Zhang et Chow, 2015b) proposent d'utiliser l'influence spatio-temporelle et la popularité des POIs.

### 3.3. Module de construction d'itinéraire

Le module de construction d'itinéraire cherche un parcours optimal, c'est-à-dire une séquence d'activités qui maximise la satisfaction de l'utilisateur par rapport aux activités pratiquées. Le problème de construction d'itinéraire peut être formulé comme un problème d'optimisation et plus précisément comme le problème nommé *Team Orienteering Problem with Time Windows* (TOPTW) (Vansteenwegen *et al.*, 2009). (Vansteenwegen *et al.*, 2009) formule ce problème de la façon suivante : soient un ensemble de POIs  $\{a_i\}, i = 1, \dots, n$ , des scores  $S_i$  associés à chaque point  $a_i$ , le temps  $T_i$  de visite de chaque POI  $a_i$ , des horaires  $[O_i, C_i]$  de la disponibilité de chaque POI  $a_i$ , le point de départ (point 1) et le point de destination (point  $n$ ), le temps  $t_{ij}$  de déplacement entre les points  $a_i$  et  $a_j$  et le budget temporel  $T_{max}$ , de trouver  $m$  parcours limités par  $T_{max}$  qui maximisent le score total collecté. Le parcours est constitué de visites de sous-ensemble de POIs, chacun ne pouvant être visité qu'une seule fois et pendant ses horaires de disponibilité. Il est permis d'arriver à POI avant son ouverture et y attendre. Dans notre cas, nous considérons des activités au lieu de POIs.

Pour résoudre ce problème, nous proposons d'adapter l'algorithme ILS (*Iterative Local Search*) introduit par (Vansteenwegen *et al.*, 2009) en le modifiant pour prendre en compte le caractère séquentiel des activités. ILS est un algorithme itératif heuristique qui cherche à maximiser le score en ajoutant une activité (nœud) au parcours existant à chaque itération. Pour qu'un nœud puisse être ajouté, il faut vérifier

que son insertion ne rend pas le parcours infaisable, c'est-à-dire, toutes les activités planifiées après ce nœud doivent satisfaire les contraintes de leurs fenêtres temporelles. Deux mesures sont ainsi introduites pour chaque nœud inclus au parcours : (1) le temps d'attente au nœud  $a_i$  dans le cas d'arrivée  $arrive_i$  avant l'ouverture :  $Wait_i = \max[0, O_i - arrive_i]$ ; et (2) le délai maximal de début de l'activité au nœud  $a_i$ , noté  $s_i$ , c'est-à-dire le délai maximum pendant lequel le début de l'activité au nœud  $a_i$  peut être retardé sans rendre infaisable l'activité suivante. Étant donnée la durée de l'activité  $T_i$  au point  $a_i$ , le moment de fermeture,  $C_i$ , et l'activité suivante  $a_{i+1}$ , ce délai est :  $MaxShift_i = \min[C_i - s_i - T_i, Wait_{i+1} + MaxShift_{i+1}]$ .

Pour chaque nœud  $a_k$  non inclus dans le parcours, l'algorithme détermine sa meilleure position d'insertion entre les nœuds  $a_i$  et  $a_j$ , c'est-à-dire celle qui a le plus bas coût temporel d'insertion, déterminé par :  $Shift_k = t_{ik} + Wait_k + T_k + t_{kj} - t_{ij}$ , avec  $Shift_k \leq Wait_j + MaxShift_j$ , condition de faisabilité. Au lieu de maximiser directement le score, l'algorithme maximise le ratio du score par le coût temporel :  $Ratio_k = \frac{S_k^2}{Shift_k}$ .

Après avoir inséré le nœud  $a_k$  les paramètres des nœuds doivent être mis à jour selon les formules présentées dans le Tableau 1. Trois cas sont considérés : les paramètres du nœud  $a_k$ , des nœuds suivants et des nœuds précédents dans le parcours.

|                          | Nœud $a_k$                         | Nœuds suivants                                  | Nœuds précédents                                                  |
|--------------------------|------------------------------------|-------------------------------------------------|-------------------------------------------------------------------|
| Temps d'arrivée          | $arrive_k = s_i + T_i + t_{ik}$    | $arrive_{j*} = arrive_j + Shift_{j-1}$          |                                                                   |
| Temps de début de visite | $s_k = arrive_k + Wait_k$          | $s_{j*} = s_j + Shift_j$                        |                                                                   |
| Temps d'attente          | $Wait_k = \max[0, O_k - arrive_k]$ | $Wait_{j*} = \max[0, Wait_j - Shift_{j-1}]$     |                                                                   |
| Délais maximal           |                                    | $MaxShift_{j*} = \max[0, MaxShift_j - Shift_j]$ | $MaxShift_i = \min[C_i - s_i - T_i, Wait_{i+1} + MaxShift_{i+1}]$ |
| Coût temporel            |                                    | $Shift_j = \max[0, Shift_{j-1} - Wait_j]$       |                                                                   |

**Tableau 1.** Les formules de mise à jour des paramètres des nœuds après l'insertion d'un nouveau nœud  $a_k$  dans le parcours.

Pour prendre en compte l'influence du caractère consécutif des activités, nous proposons de pondérer le score d'activité  $a_k$  par la probabilité de transition entre l'activité courante  $a_k$  et l'activité précédente  $a_{k-1}$  :  $Ratio_k = \frac{(S_k * P_T(a_{k-1} \rightarrow a_k))^2}{Shift_k}$ , où  $S_k$  est le score de l'activité  $a_k$  calculé par le module de calcul des scores des activités,  $P_T(a_{k-1} \rightarrow a_k)$  est une probabilité de transition entre l'activité  $a_{k-1}$  et l'activité  $a_k$ ; et  $Shift_k$  représente le coût temporel d'insertion de l'activité  $a_k$  dans le parcours.

#### 4. Discussion autour de l'évaluation

Bien qu'il existe des protocoles et jeux de test pour l'évaluation de la recommandation, il n'y a pas, à notre connaissance, de jeux de données et de protocole adaptés pour l'évaluation de séquences de POIs.

Différents types de jeux de données et de collections peuvent être utilisés pour l'évaluation des algorithmes de recommandation de POIs : issus de campagnes d'évaluation internationales comme TREC, issus des traces d'utilisation de LSBNs, issus d'applications développées spécialement pour cette tâche ou issus de la recherche opérationnelle. Ces jeux de données présentent des limites qui empêchent de les utiliser pour l'évaluation de séquences d'activités spatio-temporelles.

Les collections issues de campagnes d'évaluation, comme par exemple la tâche de suggestion contextuelle de TREC (TREC CS) (Dean-Hall *et al.*, 2014), fournissent un ensemble de profils d'utilisateurs, de contextes et de POIs, mais ne tiennent pas compte ni des contraintes temporelles entre POIs, ni des aspects séquentiels. Les jeux de données issus de LSBNs comme Yelp<sup>2</sup>, Foursquare<sup>3</sup>, Twitter<sup>4</sup>, Flickr<sup>5</sup> fournissent un ensemble de POIs et de *check-ins* effectués par les utilisateurs. Néanmoins, ces jeux de données sont souvent agrégés ce qui rend impossible la récupération des préférences personnelles. Les jeux de données issus d'applications développées spécialement pour cette tâche (comme (Schaller *et al.*, 2014a)) sont pertinents pour le problème considéré. Néanmoins, ils ne sont pas disponibles. Enfin, les jeux de données issus de la recherche opérationnelle sont disponibles en ligne<sup>6</sup>. La limitation principale de ces jeux de données est qu'ils ne fournissent aucune information sur les POIs ou les utilisateurs : tous les scores sont pré-calculés.

Dans ce contexte, nous proposons de construire un nouveau jeu de données, composé d'une collection d'activités spatio-temporelles et de profils d'utilisateurs. La collection annotée devra contenir également une hiérarchie des catégories d'activités utilisée pour attribuer une catégorie à chaque activité. La collection d'activités doit former un événement distribué et être composée d'activités concurrentes caractérisées par des horaires de disponibilité et une localisation. Nous considérons les données de 994 animations organisées lors d'une croisière de 7 nuits, disponibles en ligne<sup>7</sup>. Nous souhaitons annoter les données des activités par des utilisateurs afin de créer des profils et d'extraire les centres d'intérêt des utilisateurs, en lançant un projet de crowdsourcing sur Amazon Mechanical Turk<sup>8</sup>. La tâche consistera dans un premier temps, à compléter un questionnaire portant sur les caractéristiques personnelles, l'expérience

2. [www.yelp.com](http://www.yelp.com)

3. <https://foursquare.com>

4. <https://twitter.com/>

5. <https://www.flickr.com/>

6. <http://www.mech.kuleuven.be/en/cib/op>

7. <http://disneycruiselineblog.com/2015/07/personal-navigators-7-night-eastern-caribbean-cruise-on-disney-fantasy-itinerary-a-june-20-2015/>

8. <https://www.mturk.com>



et les goûts d'utilisateur, et dans un second temps, à annoter des activités disponibles. Le jeu de données composé sera librement distribué.

Les métriques utilisées pour évaluer les systèmes de recommandation (Shani et Gunawardana, 2011) ne sont pas adaptées au cas de la recommandation de séquences et ne tiennent pas compte de l'aspect séquentiel et de la disponibilité limitée des éléments recommandés, ce qui rend l'évaluation difficile. Nous proposons le protocole d'évaluation suivant. Dans un premier temps, nous évaluerons séparément le calcul des scores d'intérêt via des mesures connues en recommandation (comme celles utilisées par TREC CS (Précision au rang 5 (P@5), Mean Reciprocal Rank (MRR) (Dean-Hall *et al.*, 2014) ou Erreur Absolue Moyenne (MAE) (Shani et Gunawardana, 2011)) et la construction de l'itinéraire par rapport à une solution optimale. Dans un second temps, nous utiliserons le crowdsourcing pour évaluer des résultats obtenus.

## 5. Conclusion et perspectives

Dans cet article, nous avons proposé une nouvelle approche de recommandation de séquences d'activités spatio-temporelles. Elle construit des séquences d'activités personnalisées, en tenant compte de motifs de comportement typique d'utilisateur et de plusieurs facteurs qui sont utilisés pour calculer le score personnalisé des activités.

Nous pouvons distinguer les perspectives à court terme et à long terme. À court terme, nous allons implémenter l'approche proposée, construire le jeu de données décrit et tester l'approche sur ce jeu de données. À long terme, nous allons proposer une approche probabiliste qui tiendra compte du temps de déplacement stochastique et de toute la chaîne des déplacements antérieurs de l'utilisateur.

## Remerciements

Ce projet est soutenu par la Région Rhône Alpes.

## 6. Bibliographie

- Dean-Hall A., Clarke C. L. A., Kamps J., Thomas P., Voorhes E., « Overview of the TREC 2014 Contextual Suggestion Track », *NIST Special Publication 500-308 : The Twenty-Third Text REtrieval Conference Proceedings (TREC 2014)*, 2014.
- Gavalas D., Konstantopoulos C., Mastakas K., Pantziou G., « A survey on algorithmic approaches for solving tourist trip design problems », *Journal of Heuristics*, vol. 20, n° 3, p. 291-328, 2014.
- Li X., Cong G., Li X.-L., Pham T.-A. N., Krishnaswamy S., « Rank-GeoFM : A Ranking Based Geographical Factorization Method for Point of Interest Recommendation », *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '15*, ACM, New York, NY, USA, p. 433-442, 2015.

- Sang J., Mei T., Xu C., « Activity Sensor : Check-In Usage Mining for Local Recommendation », *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 6, n° 3, p. 41 :1-41 :24, April, 2015.
- Schaller R., Harvey M., Elswailer D., « Detecting Event Visits in Urban Areas via Smartphone GPS Data », in M. de Rijke, T. Kenter, A. P. de Vries, C. Zhai, F. de Jong, K. Radinsky, K. Hofmann (eds), *Advances in Information Retrieval*, vol. 8416 of *Lecture Notes in Computer Science*, Springer International Publishing, p. 681-686, 2014a.
- Schaller R., Harvey M., Elswailer D., « Relating User Interaction to Experience During Festivals », *Proceedings of the 5th Information Interaction in Context Symposium, IliX '14*, ACM, New York, NY, USA, p. 38-47, 2014b.
- Shani G., Gunawardana A., « Evaluating Recommendation Systems », in F. Ricci, L. Rokach, B. Shapira, P. B. Kantor (eds), *Recommender Systems Handbook*, Springer US, p. 257-297, 2011.
- Vansteenwegen P., Souffriau W., Oudheusden D. V., « The orienteering problem : A survey », *European Journal of Operational Research*, vol. 209, n° 1, p. 1-10, 2011.
- Vansteenwegen P., Souffriau W., Vanden Berghe G., Van Oudheusden D., « Iterated Local Search for the Team Orienteering Problem with Time Windows », *Computers and Operations Research*, vol. 36, n° 12, p. 3281-3290, December, 2009.
- Yang P., Fang H., « Exploration of Opinion-aware Approach to Contextual Suggestion », in E. M. Voorhees, A. Ellis (eds), *NIST Special Publication 500-308 : The Twenty-Third Text REtrieval Conference Proceedings (TREC 2014)*, 2014.
- Yu Z., Xu H., Yang Z., Guo B., « Personalized Travel Package With Multi-Point-of-Interest Recommendation Based on Crowdsourced User Footprints », *Human-Machine Systems, IEEE Transactions on*, vol. PP, n° 99, p. 1-8, 2015.
- Zhang C., Liang H., Wang K., Sun J., « Personalized Trip Recommendation with POI Availability and Uncertain Traveling Time », *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, CIKM '15*, ACM, New York, NY, USA, p. 911-920, 2015.
- Zhang J.-D., Chow C.-Y., « GeoSoCa : Exploiting Geographical, Social and Categorical Correlations for Point-of-Interest Recommendations », *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '15*, ACM, New York, NY, USA, p. 443-452, 2015a.
- Zhang J.-D., Chow C.-Y., « Spatiotemporal Sequential Influence Modeling for Location Recommendations : A Gravity-based Approach », *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 7, n° 1, p. 11 :1-11 :25, October, 2015b.
- Zhang J.-D., Chow C.-Y., Li Y., « LORE : Exploiting Sequential Influence for Location Recommendations », *Proceedings of the 22Nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, SIGSPATIAL '14*, ACM, New York, NY, USA, p. 103-112, 2014.

---

# Étude préliminaire à la recherche de photographies muséales en mobilité

**Maxime Portaz, Philippe Mulhem, Jean-Pierre Chevallet**

*Université Grenoble Alpes / CNRS, LIG UMR 5217, Grenoble, F-38041, France  
{Maxime.Portaz, Philippe.Mulhem, Jean-Pierre.Chevallet}@imag.fr*

---

*RÉSUMÉ. Cet article étudie la problématique de l'indexation et de la recherche d'image dans le cadre de visites de musée. Nous nous intéressons en particulier au cas d'utilisation d'outils mobiles "hors ligne" (c'est-à-dire sans connexion à un serveur distant), du point de vue qualité intrinsèque et du point de vue application mobile. Nous décrivons trois approches de référence, et nous étudions leur comportement qualitatif sur une collection de photographies de peintures, prises par des outils mobiles dans le Musée de Grenoble.*

*ABSTRACT. This paper studies the problem of images indexing and retrieval related to museum visits. We especially focus on "offline" use of mobile devices (i.e., without connection to a remote server), from the point view of intrinsic quality and the point of view of mobile potential use. We describe three approaches, and we study their qualitative behavior on a test collection of photographs of paintings taken by mobile devices in the Grenoble museum.*

*MOTS-CLÉS : Recherche d'images, Reconnaissance d'objets, Recherche d'information.*

*KEYWORDS: Image Retrieval, Object recognition, Information Retrieval.*

---

## 1. Introduction

Cet article présente une première étape dans la réalisation de recherche d'images par l'exemple, dans un cadre mobile durant des visites de musées. L'idée de base est qu'un visiteur puisse, par un appareil mobile fourni par le musée, obtenir des informations sur les œuvres qu'il photographie. A partir de cette idée générale, l'approche que nous suivons dans cet article est double : nous présentons trois des approches les plus classiques dans le domaine de l'indexation et la recherche de photographies en les étudiant dans une application mobile autonome (sans connexion avec un serveur extérieur), avant de réaliser un ensemble de tests qualitatifs sur des versions de base de ces approches.

L'indexation et la recherche d'images passent par deux étapes. La première, durant l'indexation, est l'extraction de caractéristiques visuelles et la représentation des images à partir des caractéristiques extraites. La deuxième, durant la recherche, est la représentation de l'image requête et par une correspondance entre la requête et les images du corpus afin de renvoyer les images triées suivant leur valeur de similarité avec la requête. Les trois approches que nous considérons ici sont les suivantes : i) des approches à base de représentations continues des images, qui consistent à garder toutes les caractéristiques extraites ; ii) des approches à base sacs de mots visuels, qui synthétisent les caractéristiques extraites des images par un vecteur ; et enfin iii) les approches plus récentes liées à l'utilisation de réseaux de neurones profonds (*Deep Learning*) pour la représentation des images. Ces trois approches sont détaillées dans la section 2 pour la partie indexation et dans la section 3 pour leur utilisation lors de la recherche d'images.

La section 4 portera ensuite sur la description des spécificités matérielles et logicielles des plate-formes mobiles, et sur l'adéquation sur les 3 approches d'indexation et de recherche d'image dans ce cadre. Afin d'avoir une idée plus précise en terme de qualité de ces approches dans le cadre visé, c'est-à-dire les objets visuels dans les musées, nous présentons des résultats préliminaires obtenus avec des systèmes de bases des 3 approches présentées sur des données muséales du musée de Grenoble (Mulhem *et al.*, 2013). Cette étude nous permet de déterminer comment ses approches s'adaptent par défaut à un exemple de données spécifique. Nous concluons ensuite en section 6.

## 2. Représentation d'image

La recherche d'image passe d'abord par la construction d'une représentation des images pour pouvoir les manipuler et permettre leur comparaison lors de la recherche. De nombreuses méthodes existent dans l'état de l'art pour construire une représentation des images. On peut cependant tirer les grandes lignes suivantes des approches actuelles : a) une première étape repose sur la localisation de régions d'intérêts autour de points remarquables, b) une seconde étape repose sur l'extraction des descripteurs

des régions déterminées précédemment, c) la troisième étape détermine la représentation d'une image complète à partir des caractéristiques extraites.

On retrouve, sous une forme ou une autre, ces étapes dans les trois grandes catégories d'approches pour l'indexation et la recherche d'images que nous considérons : les approches basées sur des sacs des mots visuels (Csurka *et al.*, 2004), celles basées sur des représentations continues (Calonder *et al.*, 2010), et enfin celles à base de réseaux de neurones convolutifs (Simonyan et Zisserman, 2014).

## 2.1. SIFT - Sac de mots

La première méthode, appelée "sac de mots" construit une représentation de l'image en se basant sur un vocabulaire visuel calculé a priori. Ce vocabulaire identifie des "concepts visuels", ou *mots visuels*, qui sont des représentants remarquables d'ensembles de descripteurs visuels. Comme montré sur la figure 1, il est créé grâce à un partitionnement non supervisé d'un grand ensemble de descripteurs extraits d'images, avec l'algorithme des K-Moyennes. Chaque image est alors représentée par le nombre d'occurrence de chaque mot visuel possible, d'une manière très similaire à la *term frequency* en recherche d'information textuelle.

Les descripteurs SIFT, introduits en 2004 par Lowe (Lowe, 2004), sont des descripteurs invariant à l'échelle. Ils ont démontré leurs performances par rapport aux autres descripteurs, que ce soit en reconnaissance d'objets (Mikolajczyk et Schmid, 2005) ou en correspondance d'images (Se *et al.*, 2002). Il sont également utilisés avec des sacs de mots (Csurka *et al.*, 2004).

Pour la représentation d'images à base de SIFT, la détection de régions d'intérêt (avec échelle et orientation) se fait par différence de gaussiennes (Lowe, 1999). A partir de ces régions, les descripteurs SIFT sont calculés. Chaque descripteur (un vecteur à 128 dimensions réelles) est basé sur des différences de niveaux de gris sur huit directions, accordant davantage d'importance au centre de la région d'intérêt. Pour l'étape de représentation d'une image, chaque descripteur SIFT extrait est assigné (par calcul de plus proche voisin) à un mot du vocabulaire visuel. C'est lors de cette assignation que chaque descripteur de l'image est abstrait en un concept visuels, et que l'image entière est représentée par un vecteur du nombre d'occurrences de chaque mot visuel.

Dans cette approche à base de SIFT, le calcul des régions d'intérêt est assez gourmand en temps de calcul : il faut appliquer des flous gaussiens sur l'image d'origine pour trouver les régions les plus intéressantes (i.e., celles qui ont les plus grosses variations de luminosité). Dans cette approche, l'étape d'abstraction possède l'avantage de représenter une image de manière très compacte, au détriment du niveau de détail. En effet, de nombreux descripteurs peuvent être assignés au même concept visuel : si tout se passe bien ils sont très similaires visuellement, mais cela n'est pas toujours vrai. Des améliorations ont été apportées à ce modèle initial, comme les VLAD (Jégou *et al.*, 2010), qui consistent à dépasser une assignation simples à base de plus proche

voisin. Le calcul est le même qu’avec les sacs de mots, à la différence que l’on cumule l’écart entre le descripteur local et le centroïde.

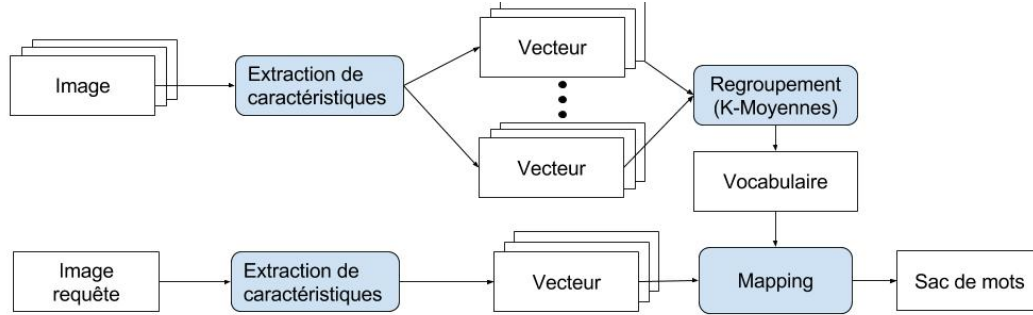


**Figure 1.** Construction de vocabulaire et indexation d'image par sac de mots visuels.

## 2.2. ORB - Représentation continue

Les méthodes à base de représentation continue des images évitent l'utilisation de dictionnaire visuel comme dans les approches à sacs de mots, mais conservent les descripteurs de l'image afin de les utiliser tels quels lors de la recherche. Pour être utilisables, ces approches nécessitent des descripteurs très rapides à calculer, et qui prennent peu d'espace mémoire. Les descripteurs ORB (Rublee *et al.*, 2011) - Oriented FAST Rotated BRIEF - valident ces deux critères.

Comme dans l'approche par sacs de mots, il faut tout d'abord déterminer des régions d'intérêt des images. Elles sont détectées par un détecteur FAST - *Features from Accelerated Segment Test* - (Rosten et Drummond, 2006). Comme l'approche FAST initiale ne permet pas de déterminer l'information d'orientation qui caractérise une région, l'extension proposée dans (Rublee *et al.*, 2011) décrit les *oFAST* - *Oriented FAST* -, détermine l'orientation de la région grâce à la méthode “*intensity centroid*” (Rosin, 1999).

Le descripteur BRIEF - *Binary Robust Independent Elementary Features* - (Calonder *et al.*, 2010) est ensuite utilisé pour décrire chaque région d'intérêt. Un descripteur BRIEF applique, en utilisant le centre de la région et des points en périphérie, un test binaire qui mesure le signe des variations d'intensité entre deux points, suivant l'équation (1), où  $p(x)$  est l'intensité de l'image  $p$  au point  $x$ .

$$f(p, x, y) = \begin{cases} 1 & p(x) < p(y) \\ 0 & p(x) \geq p(y) \end{cases} \quad [1]$$

La liste des résultats de ces tests binaires est stocké dans un vecteur de  $n$  bits (créé à partir des points comparés), suivant l'équation (2). Pour le descripteur ORB,  $n = 256$ .

$$u_n(p) = \sum_{1 \leq i \leq n} 2^{i-1} f(p, x_i, y_i) \quad [2]$$

(Rublee *et al.*, 2011) ajoutent une partie d'apprentissage pour déterminer les meilleurs paires de points à sélectionner, en : a) minimisant la corrélation entre les paires pour que chaque paire apporte de l'information, et b) maximisant la variance des paires, pour qu'elles soient discriminantes.

Calculer les descripteurs ORB est très rapide. La taille du descripteur ORB permet de conserver en mémoire les descripteurs des images du corpus afin d'éviter la perte d'information liée à la création du dictionnaire visuel.

### **2.3. Réseaux de neurones convolutifs**

Depuis quelques années, les réseaux de neurones convolutifs profonds (CNN) se sont imposés en vision par ordinateur, et ont radicalement changé les domaines de recherche d'image et d'objets. Les réseaux de neurones profonds sont constitués d'un grand nombre de neurones, organisés en couches successives, qui comportent des couches qui extraient des descripteurs et des couches qui représentent les images, toutes ces couches étant apprises conjointement. Comme montré sur la figure 2, un CNN est composé de couches de convolution et de sous-échantillonnage, dans le but d'extraire des représentations de plus en plus abstraites de l'image. Les dernières couches sont des couches entièrement connectées. L'apprentissage du réseau se fait grâce à la rétro-propagation (LeCun *et al.*, 1989), qui permet d'adapter le poids de chaque neurone pour correspondre le mieux possible à la sortie. Ce qui implique que plus on possède de neurones à entraîner, plus le nombre de données nécessaires à l'apprentissage est importante.

La collection ImageNet, composée de plusieurs millions d'images étiquetées, permet de réaliser l'apprentissage d'un très grand CNN. Cette collection a été utilisée par K. Simonyan et A. Zisserman (Simonyan et Zisserman, 2014). Pour donner une idée de la difficulté d'apprentissage dans un CNN, ils ont eu besoin de 3 semaines de calcul avec GPUs de dernière génération pour calculer les 144 millions de paramètres présents dans un CNN à 19 couches, permettant d'obtenir les meilleurs résultats sur ILSVRC-2012 avec 7,3% d'erreur.

L'avantage des réseaux de neurones profonds se situe dans le fait qu'ils apprennent tous les paramètres nécessaires à la détection d'objet automatique. L'image est passée à travers plusieurs couches de neurones, il n'y a donc pas d'extraction a priori de caractéristiques visuelles par descripteurs figés. Les caractéristiques extraites sont celles qui sont nécessaires au réseau pour calculer la sortie.

## **3. Recherche d'images**

Nous regroupons ici d'un côté les approches à base de sac de mots et de réseaux convolutifs, où chaque image est un vecteur d'entiers ou de réels, et de l'autre l'approche continue qui représente chaque image par un ensemble de vecteurs binaires.

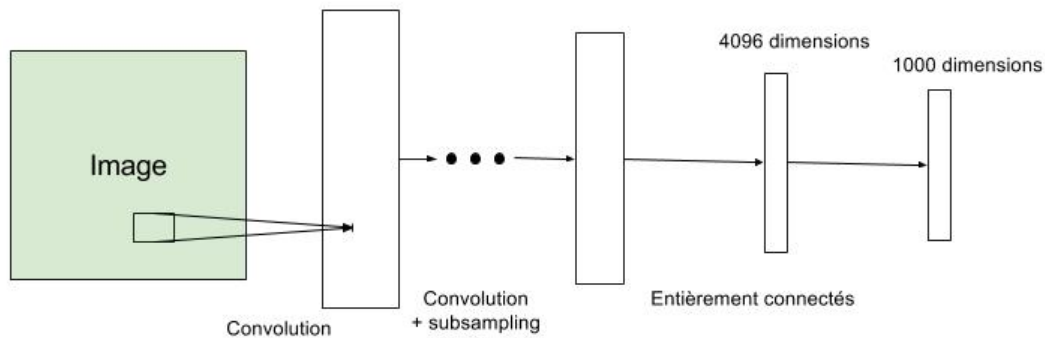


**Figure 2.** Organisation d'un réseau convolutif profond.

### 3.1. Un vecteur par image

Avec la méthode de sac de mots ou l'extraction données grâce au réseau de neurone, une image est représentée par un vecteur de réels ou d'entiers. Étant donnée une image comme requête, nous transformons l'image en sa représentation vectorielle, et nous calculons la distance entre cette image et les images de la base.

Dans le cas des sacs de mots visuels, une implantation utilisant des fichiers inverse, pour accélérer la recherche (Sivic et Zisserman, 2003 ; Jegou *et al.*, 2008), est possible. Pour le représentation extraite du réseaux de neurones, la mesure de similarité classique  $s(u, v)$  entre les vecteurs  $u$  et  $v$  est le cosinus.

### 3.2. Un ensemble de vecteurs par image

Avec la représentation continue de l'image, le calcul de similarité se fait par comparaison directe entre ceux de l'image requête et ceux des images du corpus. Comme décrit en figure 3, pour chaque image nous avons un ensemble de vecteurs, et nous devons comparer chaque vecteur entre eux pour trouver le plus proche. Les implantations utilisant les représentations Local Sensitive Hashing (LSH) (Gionis *et al.*, 1999) permettent d'obtenir très rapidement les plus proches voisins sur des descripteurs binaires. La mesure de similarité est le nombre de descripteurs correspondant entre deux images. Une vérification géométrique peut également être fait pour vérifier que la disposition des descripteur d'une image à l'autre. Ces deux méthodes permettent de trouver l'image la plus proche dans le corpus.



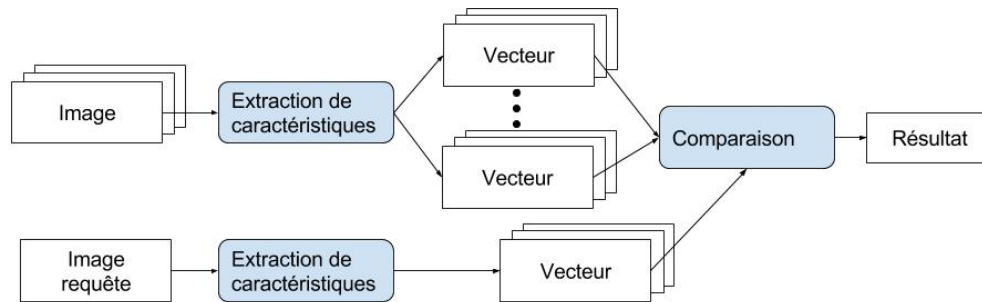


**Figure 3.** *Traitement d'une requête avec représentation d'image par un ensemble de vecteurs.*

#### 4. Comparaison des approches pour le contexte en mobilité

Comme nous souhaitons à terme porter le système sur mobile, nous nous concentrons sur les points importants dans le cadre d'un développement sur plate-forme mobile : temps de calcul et occupation mémoire.

Comme les calculs nécessaires pour mettre en place la création du dictionnaire visuel dans le cas des sacs de mots, ou l'entraînement du réseau de neurones pour les CNN peuvent être effectués sur serveur, ils sont donc hors de notre étude.

Pour les représentations à base de sacs de mots, une fois l'extraction de descripteurs et le mapping effectués, la recherche d'image est particulièrement efficace, notamment si on utilise une représentation à base de fichier inverse.

Dans le cas de l'approche continue, le pré-traitement consiste uniquement à extraire l'ensemble des descripteurs pour les images de la base. Dans le cas des descripteur ORB, les calculs sont relativement rapide. L'étape de correspondance est, par contre, plus gourmande en temps de calcul, car il faut comparer l'ensemble des descripteurs des images. Notons également que l'ensemble des descripteurs des images du corpus doit être sauvegardé en mémoire, d'où une utilisation importante de celle-ci. Dans le cas d'une application mobile, l'utilisation de LSH est indispensable, et règle ses problèmes.

Pour les réseaux de neurones, une fois l'apprentissage du réseau effectué sur des serveurs, son utilisation est une opération rapide, notamment avec un processeur graphique, même mobile. Une grande quantité de mémoire est cependant nécessaire pour stocker les paramètres du réseau. Le calcul de correspondance est alors très rapide si on utilise des représentations adaptées. Par contre, l'utilisation de fichiers inverses n'existe pas dans l'état de l'art actuel.

La tableau 1 regroupe les points faibles et points forts de chacune des méthodes étudiées suivant les deux axes d'analyse considérés. On constate qu'a priori les approches par sacs de mots sont plus adaptées à une situation en mobilité, de par leur faible besoin de mémoire et de par la rapidité de la recherche dans la base de donnée.

| Système  | Pré-traitement | Extraction de données | Recherche | Mémoire |
|----------|----------------|-----------------------|-----------|---------|
| BOW-SIFT | -              | -                     | +         | +       |
| ORB      | +              | +                     | -         | -       |
| CNN      | -              | +/-                   | +         | -       |

**Tableau 1.** Points forts (+) et faibles (-) de chaque approche considérée.

## 5. Comparaison Expérimentale

Les expérimentations préliminaires que nous présentons ici ne se préoccupent pas des aspects de traitements liés à la mobilité, mais se focalisent sur l'utilisation d'une implantation-type de chacune des 3 catégories d'approches décrites précédemment. Ces approches sont testées sur une collection de photographies d'œuvres artistiques prises par des outils mobiles. Ce corpus, défini dans (Mulhem *et al.*, 2013), est composé de 3425 images correspondant à 512 œuvres, et de 177 images requêtes. Chaque photographie a été prise par un outil mobile (iPod Touch). Le but étant de simuler la visite d'un musée, dans lequel l'utilisateur peut prendre en photo une œuvre pour obtenir des informations sur celle-ci. Les configurations que nous avons testées sont les suivantes :

- approche sac de mots - SIFT, notée *BOW-SIFT*. Le vocabulaire visuel de 2000 dimensions est créé à partir du corpus. La similarité entre image requête et image du corpus est l'inverse de la distance euclidienne entre les vecteurs. L'implantation utilise les outils par défauts proposés par OpenCV<sup>1</sup>.

- approche par représentation continue, notée *ORB*. Le calcul de similarité n'utilise pas le calcul de correspondance rapide basé sur LSH, mais la cardinalité de l'intersection utilisée entre les descripteurs d'une image et ceux de la requête fournit les mêmes résultats qu'une utilisation de LSH. L'implantation utilise les outils par défauts proposés par OpenCV.

- approche par réseau de neurone, notée *CNN*. Nous utilisons le réseaux de neurones généré par (Simonyan et Zisserman, 2014), qui a prouvé sa qualité. Comme ce réseau a appris à reconnaître les 1000 classes du challenge ILSVRC, la dernière couche de neurones (la sortie du réseau) est un vecteur de 1000 réels. Afin avoir une représentation des images moins spécifique à ILSVRC, nous utilisons l'avant dernière couche de ce réseau comme sortie, qui est un vecteur de dimension 4096. Le calcul de similarité utilisé entre image est le cosinus.

Nos résultats sont également comparés, pour Reco@1, à ceux de (Mulhem *et al.*, 2013), basés sur une approche sac de mots avec des SIFT couleur, avec des calculs de correspondance entre requête et documents par modèles de langue. Nous avons utilisé les deux mesures d'évaluation : le taux de reconnaissance à un document, Reco@1, et le taux de reconnaissance à 5 documents, Reco@5. La mesure Reco@5 se place dans

1. <http://opencv.org>

le cadre où l'on désirerait appliquer des algorithmes de vote sur la liste des résultats obtenus, comme si nous disposions d'une vidéo ou d'un ensemble de photos pour chaque requête.

| system                        | Reco@1  | Reco@5  |
|-------------------------------|---------|---------|
| (Mulhem <i>et al.</i> , 2013) | 70,06%  | /       |
| BOW-SIFT                      | 68.89%* | 93.26%* |
| ORB                           | 54.23%  | 89.26%  |
| CNN                           | 49.72%  | 56.50%  |

**Tableau 2.** Résultats obtenus sur la collection (Mulhem *et al.*, 2013) (\* : résultats préliminaires, sur une partie de la collection).

Nous constatons dans la table 2 des taux de reconnaissances pour tous nos tests nettement inférieurs à la référence. En particulier, à notre surprise, les réseaux de neurones profonds ne se comportent pas aussi bien que les deux autres approches. Ceci est sans doute dû à la spécificité des images de la collection (photographies de peintures prises par des appareils mobiles par des amateurs). Rappelons que nous avons utilisé dans ces expérimentations des approches avec leur configurations "de base", sans aucune adaptation. A la vue des résultats obtenus, il résulte que pour toutes ces approches nous devons tout d'abord nous préoccuper des paramètres qui les feront se comporter mieux sur des corpus muséaux avant d'aller plus loin.

## 6. Conclusion

Nous avons proposé dans cet article une description synthétique des approches classiques en indexation et recherche d'images, à base respectivement de sac de mots, de représentations continues et de réseaux de neurones convolutifs. Nous avons étudié l'adéquation de ces approches dans le cas spécifique pour la recherche d'image mobile "hors ligne". Nous en avons conclu que ces 3 approches sont utilisables en mobilité, moyennant des restrictions.

Les premières expérimentations sur des photographies de peintures de musée que nous rapportons nous poussent à étudier plus en détail les spécificités des données de musée pour obtenir tout d'abord de bons résultats, avant de se focaliser à l'avenir dans un second temps sur les aspects mobiles et l'étude de l'impact de certaines restriction sur la qualité des données qui nous intéressent.

## Remerciements

Ce travail a été effectué dans le cadre du projet Guimuteic financé par le Fonds Européen de Développement Régional (FEDER) et de la région Auvergne Rhône-Alpes.

## 7. Bibliographie

- Calonder M., Lepetit V., Strecha C., Fua P., « Brief : Binary robust independent elementary features », *Computer Vision–ECCV 2010*, p. 778-792, 2010.
- Csurka G., Dance C. R., Fan L., Willamowski J., Bray C., « Visual categorization with bags of keypoints », *In Workshop on Statistical Learning in Computer Vision, ECCV*, p. 1-22, 2004.
- Gionis A., Indyk P., Motwani R. *et al.*, « Similarity search in high dimensions via hashing », *VLDB*, vol. 99, p. 518-529, 1999.
- Jégou H., Douze M., Schmid C., « Hamming embedding and weak geometric consistency for large scale image search », *Computer Vision–ECCV 2008*, Springer, p. 304-317, 2008.
- Jégou H., Douze M., Schmid C., Pérez P., « Aggregating local descriptors into a compact image representation », *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, IEEE, p. 3304-3311, 2010.
- LeCun Y., Boser B., Denker J. S., Henderson D., Howard R. E., Hubbard W., Jackel L. D., « Backpropagation applied to handwritten zip code recognition », *Neural computation*, vol. 1, n° 4, p. 541-551, 1989.
- Lowe D. G., « Object recognition from local scale-invariant features », *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, vol. 2, Ieee, p. 1150-1157, 1999.
- Lowe D. G., « Distinctive image features from scale-invariant keypoints », *International journal of computer vision*, vol. 60, n° 2, p. 91-110, 2004.
- Mikolajczyk K., Schmid C., « A performance evaluation of local descriptors », *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, n° 10, p. 1615-1630, 2005.
- Mulhem P., Chevallet J., Cubaud N., « Recherche d'images en mobilité : le système IOTA-EyeSnap », *CORIA 2013 - Conférence en Recherche d'Informations et Applications - 10th French Information Retrieval Conference, Neuchâtel, Suisse, April 3-5, 2013.*, p. 63-72, 2013.
- Rosin P. L., « Measuring corner properties », *Computer Vision and Image Understanding*, vol. 73, n° 2, p. 291-307, 1999.
- Rosten E., Drummond T., « Machine learning for high-speed corner detection », *Computer Vision–ECCV 2006*, Springer, p. 430-443, 2006.
- Rublee E., Rabaud V., Konolige K., Bradski G., « ORB : an efficient alternative to SIFT or SURF », *Computer Vision (ICCV), 2011 IEEE International Conference on*, IEEE, p. 2564-2571, 2011.
- Se S., Lowe D., Little J., « Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks », *The international Journal of robotics Research*, vol. 21, n° 8, p. 735-758, 2002.
- Simonyan K., Zisserman A., « Very deep convolutional networks for large-scale image recognition », *arXiv preprint arXiv :1409.1556*, 2014.
- Sivic J., Zisserman A., « Video Google : A text retrieval approach to object matching in videos », *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, IEEE, p. 1470-1477, 2003.

---

# Using BLSTM for Interpretation of 2D Languages – Case of Handwritten Mathematical Expressions

**Ting ZHANG\*** — **Harold MOUCHERE\*** — **Christian VIARD-GAUDIN\***

*\* LUNAM/IRCCyN/IVC - UMR CNRS 6597  
Université de Nantes  
Nantes, France*

Using BLSTM for Interpretation of 2D Languages – Case of  
Handwritten Mathematical Expressions



CORIA – CIFED 2016

Structures de documents  
CIFED

Président : *XX*

Using BLSTM for Interpretation of 2D Languages – Case of  
Handwritten Mathematical Expressions



Papier absent

Papier absent

# Papier absent

Papier absent

# Papier absent



Papier absent

Papier absent



*Viet Phuong Le, Nibal Nayef, Muriel Visani, Jean-Marc Ogier, Cao De Tran*

Papier absent

Papier absent

*Viet Phuong Le, Nibal Nayef, Muriel Visani, Jean-Marc Ogier, Cao De Tran*

Papier absent



Papier absent

Papier absent

*Maroua Hammami, Pierre Héroux, Sébastien Adam, Vincent Poulain  
d'Andecy*

# Papier absent

Papier absent



*Maroua Hammami, Pierre Héroux, Sébastien Adam, Vincent Poulain  
d'Andecy*

# Papier absent



Papier absent

Papier absent

# Papier absent

Papier absent

# Papier absent







# CORIA – CIFED 2016

## Domaines spécifiques CORIA

Président : *Marc El-Bèze*



---

# Quels facteurs de pertinence pour la recherche de produits e-commerce ?

Laure Soulier\* — Lamjed Ben Jabeur\*\* — Paul Mousset\*\*  
— Lynda Tamine\*\*

\* Sorbonne Universités, UPMC Univ Paris 06, UMR 7606, LIP6, F-75005, Paris, France

laure.soulier@lip6.fr

\*\* IRIT, Université de Toulouse, CNRS, INPT, UPS, UT1, UT2J, France, 118 Route Narbonne, Toulouse, France

{jabeur,mousset,tamine}@irit.fr

---

**RÉSUMÉ.** Un moteur de recherche e-commerce vise à fournir un accès rapide et efficace à des produits qui correspondent aux besoins et aux préférences de l'utilisateur parmi une liste de produits similaires ou étroitement liés. Nous avons participé à la campagne d'évaluation « Living Lab for Information Retrieval » qui proposait une tâche de recherche de produits évaluée par des utilisateurs réels lors de scénarios de recherche réelle sur un site de e-commerce. L'évaluation expérimentale a montré des résultats prometteurs de notre modèle. Dans ce papier, nous proposons une analyse des fichiers logs issus de notre modèle afin d'identifier des facteurs d'efficacité liés à la requête et aux produits. L'objectif de cette étude est d'ouvrir des pistes de recherche pour la formalisation de modèles de recherche de produits.

**ABSTRACT.** E-commerce product retrieval aims to provide a quick and efficient access to products that fit user's needs and preferences among a tail of similar or closely related products. We participated to the "Living Lab for Information Retrieval" evaluation campaign devoted to a product search task in which real users evaluated participants' retrieval models in real search scenarios on e-commerce websites. The experimental evaluation has shown encouraging results for our proposed model. In this paper, we conduct an analysis of users' feedback with respect to the clicks obtained by our model. The goal of the paper is therefore to identify the effectiveness factors underlying the user's queries and the retrieved products in order to open perspectives in the formalization of product search models.

**MOTS-CLÉS :** Recherche d'information, recherche de produits, facteurs d'efficacité

**KEYWORDS:** Information retrieval, product search, e-commerce, effectiveness factors

---

## 1. Introduction

Ces dernières années, les sites de commerce en ligne, appelés également sites de e-commerce, ont connu une croissance régulière en termes de popularité, mais également du point de vue de leurs bénéfices. Par exemple, le site leader mondial de e-commerce Amazon revendique plus de 240 millions de produits disponibles à la vente sur la version US<sup>1</sup>. Amazon a vendu plus de 2 milliards de produits dans le monde pour l'année 2014<sup>2</sup>. Ainsi, l'expansion de ce marché confronte les utilisateurs à une multitude de produits disponibles en ligne avec des prix toujours plus concurrentiels, qui renforcent la difficulté des utilisateurs à effectuer un choix. En effet, la taille importante du catalogue de produits proposés, aussi bien en termes de types et de caractéristiques, complique l'expérience d'achat des clients sur les sites de e-commerce. Pour aider les utilisateurs, la plupart des sites de e-commerce proposent des outils de recherche de produits directement intégrés dans leurs sites Web, et non plus comme des fonctionnalités optionnelles. Par conséquent, la recherche de produits est devenue un domaine de recherche à part entière, conduisant à proposer des outils de recherche adaptés afin d'aider les clients à trouver les produits qui les intéressent. Un exemple d'outil de recherche de produits est proposé par Google Shopping<sup>3</sup> dont l'utilité a été démontrée avec 100 milliards de requêtes de recherche soumises par mois<sup>4</sup>.

Dans la littérature, la recherche de produits a été abordée comme une tâche de recherche d'information (RI) ayant pour objectif de relier les données de e-commerce avec le besoin en information d'un potentiel client. Alors que la RI classique repose sur des hypothèses d'appariement termes à termes entre une requête utilisateur et une collection de documents, la recherche de produits se retrouve confrontée à la difficulté de l'estimation d'une pertinence multi-dimensionnelle sur la base de produits caractérisés par un ensemble de méta-données (e.g., le titre, la description, le genre ou le prix). Dans ce contexte, des travaux ont proposé d'intégrer plusieurs caractéristiques produits de deux types. Une première catégorie de travaux, orientée « produits », se concentre principalement sur les méta-données des produits, à savoir leur catégorie et leur description, en proposant des approches de diversification de résultats de recherche (Chen *et al.*, 2011) ou de classification de produits à partir d'ontologies sémantiques (Vandic *et al.*, 2012). La deuxième catégorie de travaux repose sur un processus centré utilisateur en s'appuyant sur les préférences des utilisateurs et leurs intentions (Duan *et al.*, 2013).

Du point de vue de l'évaluation de ces modèles, une première initiative dans le cadre de la campagne « Living Labs for IR » (LL4IR) (Schuth *et al.*, 2015) à

1. <http://www.amazon.com/>

2. <http://www.ecommercebytes.com/cab/abn/y14/m07/i15/s04>

3. <http://www.google.fr/shopping>

4. <http://www.godatafeed.com/resources/google-shopping-campaigns>

CLEF 2015 (Mothe *et al.*, 2015) a vu le jour pour formaliser un cadre d'expérimentation centré utilisateur permettant à des vrais clients d'une plateforme de e-commerce<sup>5</sup> de juger des produits identifiés comme pertinents par les participants de la campagne d'évaluation. Dans ce contexte, nous avons proposé un modèle probabiliste (Ben Jabeur *et al.*, 2015) qui allie à la fois l'approche orientée « produits », en se basant sur leurs descripteurs, ainsi que l'approche centrée utilisateur en prenant en compte la notion d'engagement des utilisateurs vis-à-vis d'un produit. Les résultats de la campagne d'évaluation ont souligné l'efficacité de notre approche avec des mesures d'efficacité supérieures à l'ensemble des modèles proposés par les participants ainsi que le système de référence (*baseline*).

Étant donné les résultats encourageants obtenus, nous nous intéressons dans ce papier à l'identification des facteurs d'efficacité de notre modèle pour la recherche de produits à partir des fichiers *logs* de notre modèle. Notre objectif est donc d'analyser les résultats obtenus lors de notre participation à la campagne LL4IR afin d'étudier quels facteurs, en termes de méta-données de produits, impactent sur l'efficacité de la recherche de produits.

La structure du papier est organisée comme suit. La Section 2 introduit une synthèse des travaux de recherche de produits. La section 3 présente le cadre d'expérimentation LL4IR ainsi que notre participation à cette campagne d'évaluation. La section 4 détaille notre analyse des facteurs d'efficacité pour la recherche de produits. Enfin, la section 5 conclue le papier et présente les perspectives de recherche.

## 2. Recherche de produits : synthèse des travaux

Alors que certains travaux se sont intéressés à la recherche de produits sous l'angle de la modélisation utilisateur pour la prise de décision (Chen, 2010; Castagnos *et al.*, 2010), d'autres travaux, plus proches de notre contribution, ont abordé la recherche de produits comme une tâche de RI qui vise exploiter les données de e-commerce pour répondre aux besoins en information exprimés par la clientèle pendant le processus d'achat en ligne.

Detlor et al. (Detlor *et al.*, 2003) ont fait une analyse comparative de deux tâches d'exploration et de recherche de produits sur les sites de e-commerce. A la différence de la tâche d'exploration dans laquelle l'utilisateur n'a pas un objectif précis pour ses achats et dont les spécifications des produits attendus sont encore vagues, la tâche recherche des produits est motivée par le fait que l'utilisateur cherche une information précise dans les spécifications d'un produit afin de prendre une décision. Outre les informations élémentaires sur le produit (par exemple le prix et la description) et les informations sur le commerçant lui-

---

5. <http://www.regiojatek.hu/>

même, les spécifications d'un produit sont considérées comme des informations pertinentes dans la tâche de recherche de produits.

Peu de travaux axés sur les modèles d'ordonnancement se sont intéressés à la tâche de recherche de produits. Ces travaux ont principalement proposé d'intégrer plusieurs fonctionnalités que l'on peut classer selon deux catégories.

D'une part, certains travaux se concentrent principalement sur le produit, à savoir sa catégorie et sa description. Chen et al. (Chen *et al.*, 2011) ont proposé de diversifier les résultats de recherche de produits en prenant en compte les catégories de produits et les valeurs d'attributs comme un critère de diversité. Vandic et al. (Vandic *et al.*, 2012) étudient le vocabulaire utilisé pour décrire le même produit et proposent une classification hiérarchique des produits en fonction de ces termes. En complément de cette méthode et en se basant sur une ontologie sémantique, ils proposent de reconnaître les produits similaires et de les classer selon une taxonomie de produit universelle.

D'autre part, la deuxième catégorie de travaux s'est intéressée à la problématique de recherche des produits axée sur la différence du vocabulaire utilisé sur la fiche produits et du vocabulaire utilisé par les utilisateurs pour formuler leurs requêtes de recherche. Cela inclut des requêtes de type "*PC gamer pas cher*" visant un besoin spécifique par rapport à une caractéristique technique de produit. Considérant la recherche de produits comme une tâche de recherche d'entités, Duan et al. (Duan *et al.*, 2013) proposent d'exploiter la structure attribut-valeur de la fiche de description du produit ainsi que les commentaires des utilisateurs afin de répondre aux requêtes qui relèvent des exigences techniques et qui utilisent des vocabulaires différents. Les résultats sont ainsi ordonnés en utilisant un modèle probabiliste qui estime la pertinence au niveau des préférences d'attributs.

L'objectif de ce papier est de fournir une meilleure compréhension de la tâche de recherche de produits. Contrairement à Detlor et al. (Detlor *et al.*, 2003) qui favorisent une analyse axée sur les comportements de recherche de produits, nous proposons dans ce papier une analyse orientée vers le domaine de la RI en nous intéressant aux facteurs de pertinence sous-jacents aux requêtes soumises et aux produits sélectionnés lors d'une recherche de produits sur un site de e-commerce. Pour cela, nous nous appuyons sur un modèle proposé lors de la campagne d'évaluation « Living Lab », qui allie à la fois l'approche centrée « produits » et l'approche centrée « utilisateurs », proposées dans les travaux précédents (Chen *et al.*, 2011 ; Duan *et al.*, 2013 ; Vandic *et al.*, 2012).

### 3. Cadre Experimental de la campagne « Living Lab »

La campagne d'évaluation "*Living Labs for Information Retrieval*" (LL4IR) (Schuth *et al.*, 2015 ; Cappellato *et al.*, 2015) propose une approche novatrice pour l'évaluation de modèles de RI. En effet, le cadre d'évaluation est carac-

térisé par l'implication d'utilisateurs dans des scénarios d'utilisation réels pour lesquels l'utilisateur soumet sa propre requête sur le site et interagit en temps réel avec les résultats retournés au moyen des *clics*. La période d'évaluation réalisée selon des scénarios de recherche réelle est appelée « période de test ». L'objectif de cette campagne est de tenir compte des *clics* générés par des utilisateurs réels afin de comparer les différents systèmes proposés par les participants et le système de production. Plus particulièrement, pour la campagne 2015, la tâche de recherche principale est une tâche de recherche de produits sur le site de commerce en ligne du leader hongrois de vente de jouets pour enfants *REGIO JÁTÉK*<sup>6</sup>. Dans ce papier, nous nous concentrons sur la période de test ayant eu lieu entre le 15 et 31 juillet 2015. Nous décrivons dans ce qui suit le cadre expérimental mis en œuvre lors de la campagne LL4IR 2015 en introduisant les données expérimentales, le protocole d'évaluation ainsi que notre participation avec une présentation du modèle proposé et des résultats obtenus.

### 3.1. Données expérimentales

La campagne d'évaluation LL4IR 2015 fournit une plate-forme de référence implémentée sous forme d'un service en ligne qui met à disposition des participants un ensemble de données expérimentales :

- « *participant/query* » qui regroupe 50 requêtes orientées produit. Elles sont extraites à partir des requêtes utilisateurs soumises dans le passé afin de garantir qu'elles seront vraisemblablement resoumises durant les périodes de test de la campagne d'évaluation.

- « *participant/doclist* » qui regroupe une collection de produits (appelés aussi documents). Cette collection inclut des produits disponibles, mais également ceux notés comme indisponibles qui seront potentiellement disponibles dans le futur. Le nombre moyen de produits retournés pour chaque requête est d'environ 62 produits. Ce nombre peut dépasser 500 produits pour certaines requêtes. Chaque produit est représenté par un ensemble de méta-données structurelles et sémantiques, comme les personnages associés au produit (par exemple, Barbie, Spiderman, Hello Kitty), sa marque (par exemple, Beados, LEGO, Simba) ou encore l'âge et le genre recommandé. Ces méta-données, sont présentées dans le Tableau 1.

- « *participant/feedback* » qui rassemble les jugements de pertinence mis à jour sur la plate-forme toutes les 5 minutes sur l'ensemble d'une période de test donnée. Chaque jugement de pertinence est binaire en fonction de si le produit présenté à l'utilisateur a été cliqué.

- « *participant/historicalfeedback* » qui regroupe l'historique des *clics* utilisateurs disponible pour toutes les requêtes. Cet historique est généré sur une

---

6. <http://www.regiojatek.hu/>

| Méta-données             | Description                                                    |
|--------------------------|----------------------------------------------------------------|
| Nom                      | Nom du produit                                                 |
| Description              | Description longue du produit                                  |
| Description courte       | Courte description du produit                                  |
| Age maximum              | Âge maximum recommandé                                         |
| Age minimum              | Âge minimum recommandé                                         |
| Date d'arrivée           | Nombre de jours de stockage                                    |
| Prix                     | Prix sans réduction                                            |
| Bonus                    | Indique si le produit bénéficie (1) ou non (0) d'une réduction |
| Réduction appliquée      | Montant de la réduction du produit                             |
| Pourcentage de réduction | Taux de réduction du produit                                   |
| Marque                   | Marque                                                         |
| Catégorie                | Catégorie du produit                                           |
| Nombre de personnages    | Liste des personnages connus associés au produit               |
| Genre                    | Genre (0 : Neutre ; 1 : Garçon ; 2 : Fille)                    |
| Nombre de photos         | Liste des photos                                               |

**Tableau 1.** *Description des produits*

période de plusieurs mois et a permis aux organisateurs de la campagne d'évaluation de constituer la collection de produits selon le critère qu'un produit peu cliqué est exclu de la collection.

Nous soulignons que ces différentes données expérimentales, hormis les jugements de pertinence des utilisateurs, sont statiques pour une période de test. Ces données sont mises à jour peu avant chaque période de test.

### 3.2. *Participants LL4IR*

Quatre organismes sont impliqués dans la tâche de recherche de produits de la campagne LL4IR. Nous présentons les différentes participations et nous détaillerons la description de notre modèle.

– «GESIS» (Schaer et Tavakolpoursaleh, 2015) ont proposé un modèle en deux étapes : (1) identifier les produits pertinents et (2) les ré-ordonnancer à partir de l'historique des jugements de pertinence. Cette approche de recherche a ensuite été améliorée par des techniques de filtrage sur des critères de disponibilité de produits ou de taux de réduction des prix affichés.

– «UiS» (Ghirmatsion et Balog, 2015) se sont appuyés sur des techniques de modèles de langue pour estimer la similarité produit-requête avec une intégration d'une probabilité a priori basé sur l'historique des *clics*. Trois méthodes ont été testées au travers de trois participants différents et sont présentées dans les résultats sous la forme «UiS-\*».



– «Baseline» représente le système de référence proposé par les organisateurs de la campagne d'évaluation. Ce système est basé sur l'historique des *clics* (Schuth *et al.*, 2015).

– «Notre modèle» (Ben Jabeur *et al.*, 2015), notre participation, qui consiste en un modèle probabiliste utilisant les champs descriptifs textuels (e.g., titre, marque, catégorie du produit). Compte tenu de la représentation des produits sous forme des champs (attributs et valeurs) et inspirés par les travaux de Craswell *et al.* (Craswell *et al.*, 2005) et Dakka *et al.* (Dakka *et al.*, 2012), nous proposons de représenter un produit  $p$  par deux ensembles des champs dont (1) l'ensemble de champs descriptifs textuels  $\mathcal{D}_p$  qui décrivent le produit  $p$  et (2) l'ensemble des champs de structuration  $c_p$  et permettent d'organiser les produits selon des catégories. La pertinence  $P(p|q)$  d'un produit  $p$  par rapport à la requête  $q$  est traduite par la probabilité  $P(c_p, \mathcal{D}_p|q)$  (Equation 1). Selon la transformation des probabilités de Bayes (Equation 2) et compte tenu de l'indépendance assumée entre la catégorie d'un produit et sa description (Equation 3), la pertinence d'un produit est estimée par l'équation suivante :

$$P(p|q) = P(c_p, \mathcal{D}_p|q) \quad [1]$$

$$= P(c_p|q) \cdot P(\mathcal{D}_p|c_p, q) \quad [2]$$

$$\propto P(c_p|q) \cdot P(\mathcal{D}_p|q) \quad [3]$$

Avec  $P(c_p|q)$  représentant respectivement la pertinence de la catégorie  $c_p$  du produit  $p$  en se basant sur une distribution de la catégorie  $c_p$  sur l'ensemble des produits.  $P(\mathcal{D}_p|q)$  exprime la pertinence thématique des champs descriptifs  $\mathcal{D}_p$  du produit  $p$  par rapport à la requête  $q$  au travers du modèle *BM25F*. Ce modèle calcule la similarité entre le produit  $p$  et la requête  $q$  tout en attribuant l'importance variée à chacun des champs. Le cadre d'évaluation imposé par le LL4IR ne permettant aux participants de soumettre l'ordonnancement d'un seul modèle, nous nous sommes appuyés sur les expérimentations menées dans les précédents travaux afin de fixer les poids ((Zaragoza *et al.*, 2004) pour les poids des descripteurs et (Craswell *et al.*, 2005; Zaragoza *et al.*, 2004) pour les poids des paramètres du *BM25F*). Les produits pertinents sont ensuite réordonnés selon une notion d'engagement des utilisateurs traduite autour d'une notion de la popularité (Lehmann *et al.*, 2012). Cette dernière est estimée par le nombre de marquages « j'aime » et de partages générés sur la plate-forme sociale Facebook<sup>7</sup>.

### 3.3. Protocole d'évaluation

Afin d'évaluer les modèles de recherche des produits via la plate-forme LL4IR, les participants procèdent aux étapes suivantes :

7. <http://www.facebook.com>

Quels facteurs de pertinence pour la recherche de produits e-commerce ?

- 1) Recueillir l'ensemble des requêtes depuis la ressource requêtes "*participant/query*"
- 2) Pour chaque requête, extraire la liste des produits candidats, disponible via la ressource doclist "*participant/doclist*".
- 3) Pour chaque ID de produit dans la liste, récupérer le contenu du produit par la ressource API "*participant/doc*".
- 4) Pour chaque requête, soumettre la liste des produits ordonnée selon un modèle de recherche de produits via la ressource "*participant/run*".

En conformité avec l'approche centrée utilisateur proposée par la campagne LL4IR 2015, les ordonnancements des produits proposés par l'ensemble des participants sont mélangés avec l'ordonnancement par défaut du système de production du site de e-commerce. Lorsqu'un utilisateur soumet une requête qui appartient à l'ensemble des requêtes pré-sélectionnées sur la plate-forme, l'utilisateur reçoit un ensemble de produits dont la moitié provient du système de production du site web et l'autre moitié des ordonnancements fourni par les participants à la campagne d'évaluation.

Dans l'objectif d'évaluer chacune des approches, les organisateurs de la campagne LL4IR 2015 ont proposé cinq métriques. Ces dernières sont estimées sur la base de l'ensemble des requêtes soumises :

- Le nombre de « victoires », noté  $\#Wins$ , qui exprime le nombre de fois où le système du participant a reçu plus de *clics* que le système de production.
- Le nombre de « pertes », noté  $\#Losses$ , qui exprime le nombre de fois où le système du participant a reçu moins de *clics* que le système de production.
- Le nombre d'« égalités », noté  $\#Ties$ , qui exprime le nombre de fois où le système du participant a reçu autant de *clics* que le système de production.
- Le nombre d'« impressions », noté  $\#Impressions$ , qui exprime le nombre de fois où un ordonnancement d'un participant a été mélangé à celui du système de production. Plus particulièrement, le nombre d'impressions est estimé comme suit :

$$\#Impressions = \#Wins + \#Losses + \#Ties \quad [4]$$

- Le « ratio de victoires », noté *Outcome*, est défini comme le quotient de « victoires » sur la sommes de « victoires » et « pertes », comme indiqué dans l'équation 5. Un ratio supérieur à 0.5 souligne la capacité du système à fournir plus de produits pertinents que de produits non pertinents, en se basant sur l'hypothèse que les *clics* sont des indicateurs de pertinence (Joachims, 2002).

$$Outcome = \frac{\#Wins}{\#Wins + \#Losses} \quad [5]$$

### 3.4. Résultats obtenus pour la campagne LL4IR

Le Tableau 2 présente les résultats obtenus par les différents participants. En excluant le premier participant n'ayant pas fourni des ordonnancements sur toutes les requêtes, les résultats soulignent l'efficacité de notre modèle. En effet, nous obtenons la valeur de « ratio de victoires » la plus élevée (0.489) qui dépasse les valeurs des métriques obtenues pour l'ensemble des participants ainsi que la *Baseline* fournie par les organisateurs de la campagne d'évaluation (Schuth *et al.*, 2015). Ces modèles reposent essentiellement sur les historiques des *clics* ainsi que les descripteurs des produits; ces derniers étant utilisés également dans notre modèle. Alors que l'historique des *clics* aurait pu être un facteur significatif de l'engagement des utilisateurs, les augmentations obtenues pour notre modèle (comprises entre +8.49% et +41.37%) mettent en évidence la pertinence de notre estimation de l'engagement utilisateur au sein d'une tâche de recherche de produits. De plus, notre modèle a obtenu des valeurs de « ratio de victoires » très proches de 0.5, suggérant que notre modèle est très proche de celui de système de production déployé sur le site de e-commerce. Les valeurs de « ratio de victoires » obtenues par les participants demeurent cependant inférieures à 0,5; soulignant la difficulté de formaliser des modèles de recherche de produits qui est un domaine de recherche relativement jeune. Cependant, les résultats encourageants obtenus par notre modèle nous poussent à effectuer des analyses complémentaires afin de mieux comprendre les facteurs de pertinence pour la recherche de produits. Nous aborderons cet aspect dans la prochaine section.

|                | Outcome | #Wins | #Losses | #Ties | #Impressions |
|----------------|---------|-------|---------|-------|--------------|
| Peter Dekker * | 0.5429  | 19    | 16      | 76    | 111          |
| Notre modèle   | 0.4890  | 89    | 93      | 533   | 715          |
| UiS-Mira       | 0.4507  | 64    | 78      | 527   | 669          |
| Baseline       | 0.4430  | 66    | 83      | 498   | 647          |
| GESIS          | 0.4134  | 74    | 105     | 513   | 692          |
| UiS-Jern       | 0.3702  | 67    | 114     | 511   | 692          |
| UiS-UiS        | 0.3459  | 55    | 104     | 521   | 680          |

**Tableau 2.** Résultats LL4IR sur la période 15-31 juillet 2015. \* : participant n'ayant pas fourni des ordonnancements sur toutes les requêtes.

## 4. Analyse des facteurs de pertinence pour la recherche de produits

Dans cette section, nous nous intéressons particulièrement à l'analyse des facteurs de pertinence afin d'ouvrir des pistes de recherche pour la formalisation de modèles de recherche de produits. Ne disposant pas des fichiers log du système de production pour cette période d'évaluation, nous nous appuyons sur les résultats obtenus par notre modèle en faisant l'hypothèse qu'il constitue une bonne approximation des préférences des utilisateurs dans la mesure où il a

obtenu des résultats sensiblement équivalents à ceux du système de production. Dans ce contexte, nous nous intéresserons particulièrement aux deux questions de recherche suivantes :

– **RQ1** : Quelles sont les caractéristiques des requêtes utilisateurs lors d’une recherche de produits ? Quel est l’impact de ces caractéristiques sur l’efficacité de la recherche ?

– **RQ2** : Quelles sont les caractéristiques des produits retournés par notre modèle ? Quelles caractéristiques des produits impactent l’efficacité de la recherche ?

#### 4.1. Facteurs de pertinence liés à la requête

Nous proposons de nous intéresser ici à la première question de recherche (RQ1) axée sur l’analyse des caractéristiques des requêtes ainsi que les facteurs sous-jacents qui peuvent impacter la pertinence de chacune des requêtes. Dans un premier temps, nous nous intéressons à caractériser les requêtes générées dans le cas de recherches de produits et formulées par des utilisateurs réels. Le Tableau 3 présente les caractéristiques des 50 requêtes évaluées lors de la campagne LL4IR. Ce sont des requêtes généralement courtes, voire très courtes (1 mot) qui font référence à des jeux et objets ou également des produits dérivés de marques (par exemple « Monster », « Lego » ou « Playmobil »). Peu de requêtes évoquent des personnages tels que « Angry Bird » et « Cars ».

|        | Caractéristique                                          | Nombre de requêtes |
|--------|----------------------------------------------------------|--------------------|
| Long.  | 1 mot                                                    | 45                 |
|        | 2 mots                                                   | 5                  |
| Classe | Object, dispositif et jeux                               | 31                 |
|        | Marque, franchise, ligne de produit et modèle de produit | 10                 |
|        | Thème, catégorie et concept                              | 5                  |
|        | Personnage fictif, film, servie TV et jeux vidéos        | 3                  |
|        | Animal                                                   | 1                  |
|        | Autres                                                   | 1                  |
| Pers.  | Avec personnage                                          | 7                  |
|        | Sans personnage                                          | 53                 |
| Marq.  | Avec marque                                              | 10                 |
|        | Sans marque                                              | 40                 |

**Tableau 3.** *Analyse descriptive des caractéristiques des requêtes*

Nous nous intéressons maintenant à l’analyse des relations entre les caractéristiques et l’efficacité de ces requêtes. Dans un premier temps, nous présentons au travers de la Figure 1 l’efficacité de chaque requête par rapport à la mesure « ratio de victoires » fournie à l’issue de la campagne d’évaluation LL4IR. Nous soulignons que parmi les 50 requêtes, nous avons obtenu un « ratio de victoires » supérieur ou égal à 0.5 pour 26 requêtes. Cependant, nous observons

également 17 requêtes avec une mesure d'efficacité égale à 0. Nous remarquons ainsi qu'il existe trois classes de requêtes identifiées selon le critère d'efficacité permettant de classer les requêtes suivant si elles obtiennent des résultats supérieurs ou inférieurs au système de production ou encore des requêtes en situation d'échec. Afin de construire des classes de requêtes de façon formelle, nous procédons à une classification *k-means* en trois classes (*3-means*), permettant d'identifier les requêtes suivant leur échec, succès ou si elles ont obtenu des résultats acceptables en termes d'efficacité. Le Tableau 4 résume les statistiques d'efficacité sous-jacentes à chaque classe. La classification est bien conforme à notre intuition, fixant des seuils entre les classes à 0,2 et 0,6 pour accéder respectivement aux classes « Acceptable » et « Succès ».

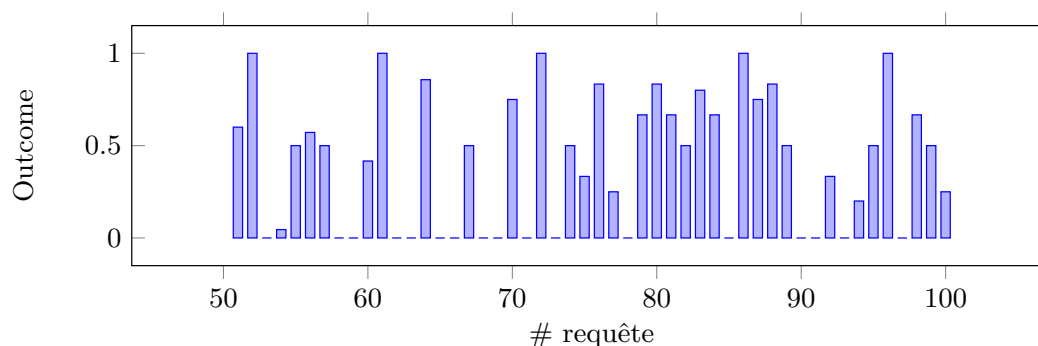


**Figure 1.** Analyse des requêtes selon la mesure d'efficacité « ratio de victoires »

|            | Nombre de requêtes | Min   | Moyenne | Max   | Écart-type |
|------------|--------------------|-------|---------|-------|------------|
| Échec      | 19                 | 0     | 0,0129  | 0,200 | 0,046      |
| Acceptable | 15                 | 0,250 | 0,450   | 0,600 | 0,109      |
| Succès     | 16                 | 0,667 | 0,833   | 1     | 0,133      |

**Tableau 4.** Analyse de l'efficacité selon chaque classe de requêtes

Nous nous sommes ensuite intéressés à définir le modèle statistique qui permet d'expliquer quelles caractéristiques ont un impact significatif sur l'efficacité. Pour cela, nous avons considéré la variable issue de la classification *3-means* comme variable à expliquer et les caractéristiques présentées précédemment dans le Tableau 3 comme variables explicatives. Les résultats montrent qu'aucune des caractéristiques liées à la requête impacte l'efficacité obtenue. Ces résultats peuvent être expliqués par le fait que les requêtes ne sont pas assez nombreuses dans chaque catégorie des caractéristiques (voir Tableau 3), ne permettant pas d'avoir des différences significatives entre les classes.

#### 4.2. Facteurs de pertinence liés au produit

La synthèse des travaux du domaine ainsi que les modèles expérimentés lors de la campagne d'évaluation illustre l'importance des caractéristiques des

produits (plutôt que de la requête) dans la prise en compte du score de pertinence. Ainsi, dans ce qui suit, nous nous intéressons à étudier les relations potentielles entre les caractéristiques des produits et l'efficacité. Plutôt que de déduire l'efficacité à partir de la mesure « ratio de victoires » estimée au niveau de la requête et dénotant plutôt une préférence de l'utilisateur au système de référence, nous souhaitons exploiter les jugements de pertinence des utilisateurs pour chaque produit retourné par notre modèle. Cette analyse permet d'identifier une relation entre les caractéristiques des produits et les préférences des utilisateurs vis-à-vis de chaque produit afin d'approximer au mieux les spécificités de la recherche produit. Pour cette raison, plutôt que d'utiliser l'historique des jugements de pertinence qui sont fortement corrélés à la temporalité (« effet de mode » d'un produit) et à la disponibilité des produits, nous utilisons une mesure fournie par les organisateurs du « Living Lab » correspondant à la probabilité d'un produit d'être cliqué. Cette probabilité est estimée durant la période de test par le ratio entre le nombre de *clics* d'un produit et le nombre de fois où le produit a été présenté à l'utilisateur.

Le Tableau 5 présente les caractéristiques qualitatives des produits. Les variables numériques sont présentées dans le Tableau 6. Les produits généralement retournés par notre modèle sont des produits assexués qui ne bénéficient pas spécifiquement de réduction. Le site étant un site de vente de jouets, les produits recommandés sont généralement préférés pour des enfants entre 3 et 6 ans, présentés avec 2 photos et rarement reliés à des personnages connus. Le prix moyen de ces produits est de 348.7 forints hongrois (équivalent à 1.10€ ) avec un taux moyen de réduction à 13.14%.

|       | Caractéristique | Nombre de produits |
|-------|-----------------|--------------------|
| Genre | Neutre          | 2397               |
|       | Fille           | 368                |
|       | Garçon          | 267                |
| Bonus | Avec réduction  | 292                |
|       | Sans réduction  | 2740               |

**Tableau 5.** *Analyse descriptive des caractéristiques qualitatives des produits*

| Caractéristique          | Min | Moyenne | Max  |
|--------------------------|-----|---------|------|
| Age minium               | 0   | 3.387   | 10   |
| Age maximum              | 0   | 6       | 10   |
| Date d'arrivée           | 20  | 932.1   | 2098 |
| Prix                     | 1   | 348.7   | 712  |
| Réduction appliquée      | 0   | 309.3   | 3000 |
| Pourcentage de réduction | 1   | 13.14   | 252  |
| Nombre de photos         | 0   | 1.968   | 50   |
| Nombre de personnages    | 0   | 0.12    | 2    |

**Tableau 6.** *Analyse descriptive des caractéristiques numériques des produits*

De façon analogue à l'analyse des requêtes, notre objectif est de construire le modèle statistique qui permettrait d'expliquer quels sont les produits que les utilisateurs ont tendance à acheter sur les sites de e-commerce. Étant donné les deux types de variables (qualitatives et quantitatives), nous avons effectué une analyse de covariance où nous avons construit le modèle itérativement en enlevant à chaque itération le facteur avec la *p-value* la plus importante jusqu'à n'inclure que des caractéristiques avec des *p-values* inférieures ou égales à 0.5. Les résultats sont présentés dans le Tableau 7.

| caractéristiques         | coefficient | p-value      |
|--------------------------|-------------|--------------|
| Nombre de photos         | 2.367e-3    | 8.01e-07 *** |
| Date d'arrivée           | 5.242e-07   | 0.001 **     |
| Pourcentage de réduction | -4.349e-02  | 0.009 **     |
| Réduction appliquée      | 1.744e-02   | 0.008 **     |

**Tableau 7.** *Modèle descriptif des préférences utilisateurs lors de la recherche de produits sur des sites de e-commerce*

L'analyse de covariance révèle que trois classes de caractéristiques de produits impactent les préférences des utilisateurs : la présence de photo(s), l'ancienneté d'un produit et également les informations relatives à son prix. En effet, la présence de photos semble susciter l'intérêt des utilisateurs, comportement traduit par un coefficient positif dans l'analyse de covariance. Très simplement, ce résultat est expliqué par le fait que la présence de photos apporte plus de crédibilité au produit et permet aux utilisateurs d'avoir une image du produit qu'ils souhaitent acheter. En effet, la présence de photos est un facteur déclenchant d'achat bien connu des stratégies de marketing avec certaines études mettant en avant le fait que 67% de consommateurs considèrent que les images des produits sont extrêmement importantes lors de leurs achats en ligne<sup>8</sup>. La date d'arrivée du produit semble également impacter positivement le comportement des utilisateurs. Les résultats semblent suggérer que les utilisateurs apprécient les produits avec un peu d'ancienneté dans la mesure où ils attendent un petit peu de temps avant de les acheter afin d'avoir l'avis des autres utilisateurs qui auront acheté le produit dès les premiers jours de sa mise à disposition.

Une autre conclusion qu'il est intéressant de souligner réside dans le fait que les caractéristiques liées au prix et à la réduction du prix d'un produit impactent également les préférences des utilisateurs. Dans le même esprit, bien que l'analyse de covariance révèle que la présence d'une réduction génère une appétence vers le produit plus forte, la valeur de la réduction, quant à elle, semble être négativement corrélée avec les préférences des utilisateurs. D'une façon intéressante, le prix semble positivement corrélé ainsi que le taux de réduction négativement corrélé aux préférences des utilisateurs permettent de déduire que la recherche de produits est particulièrement orientée vers des

8. <http://blog.lemonstand.com/7-ways-optimize-product-page-conversions/>

produits spécifiques (par exemple « Playmobil ») et que les utilisateurs ne recherchent donc pas de produits similaires moins onéreux.

Il nous semble également important de souligner que certaines caractéristiques n'ont pas été identifiées comme significatives en termes d'impact sur les préférences des utilisateurs. Il s'agit principalement des caractéristiques de genre, d'âge recommandé et des personnages connus reliés au produit.

### 4.3. *Discussion*

L'analyse des facteurs de pertinence repose sur des métriques qui compare le modèle évalué avec le système de production. Ces mesures comportent certaines limites dans la mesure où elles ne sont pas basées sur des mesures formelles et ne prennent pas en compte le nombre de clics global des utilisateurs. Malgré cela, notre analyse a permis de mettre en évidence les conclusions suivantes :

- *Des requêtes non discriminantes pour la recherche de produits.* Les analyses descriptives semblent montrer dans un premier temps que les requêtes formulées par les utilisateurs n'incluent pas spécifiquement des caractéristiques « produit » (telles que la marque ou le personnage). Cette intuition a été démontrée par l'analyse de variance qui a approuvé qu'il est difficile de mesurer l'inefficacité de la recherche, voire qu'il n'existe pas de relation entre les caractéristiques des requêtes et l'efficacité de la recherche. Cependant, nous sommes conscients qu'il aurait fallu étayer ces résultats par une analyse comparative des requêtes réalisées dans le cas d'une recherche produit et une recherche documentaire évaluée selon un même cadre d'évaluation ; ce qui n'a pas été mis en place dans le cas de la campagne d'évaluation « Living Lab ».

- *Des préférences utilisateurs orientées sur une gamme de produits plutôt que la recherche de produits génériques.* Les résultats mettant en avant l'impact du prix et du taux de réduction sur la probabilité de pertinence d'un produit ont mis en évidence le besoin des utilisateurs d'identifier des produits correspondant exactement à leur requête plutôt que de favoriser une recherche de produits similaires et moins onéreux. De façon similaire, l'absence de relation entre la pertinence du produit et les caractéristiques de genre et d'âge suggèrent que les utilisateurs recherchent des produits bien spécifiques et que favoriser un genre ou un âge en particulier ne permettra pas de provoquer une déviance dans le comportement de l'utilisateur. En effet, une requête orientée produit est souvent reliée à un produit ou type de produit en particulier (par exemple « doll »), ainsi l'utilisateur a déjà une idée du genre, de l'âge recommandé ou des personnages reliés au produit. Cependant, cette explication est difficile à démontrer dans la mesure où les requêtes ne sont rattachées à aucune donnée de ce type, empêchant la mise en correspondance entre les préférences des utilisateurs et les produits jugés comme pertinents.

- *La présentation du produit est prépondérante.* La pertinence des produits étant impactée par la présence de photos dans la description du produit signifie la prépondérance de l'aspect visuel sous-jacent à la présentation du pro-



duit. Par conséquent, il serait intéressant de prendre en compte la présentation des caractéristiques des produits comme facteur explicatif de la pertinence du produit. En effet, les réductions sont généralement affichées sous forme d'un pictogramme qui signifie à l'utilisateur que le prix est soldé. Ceci peut expliquer les coefficients opposés obtenus pour les facteurs liés à l'absence (ou la présence) d'une réduction et le taux de réduction. Un aspect intéressant aurait été d'analyser l'impact de ce pictogramme sur l'appréciation de la présence de réduction, en effectuant la même évaluation sur un site ne favorisant pas cette mise en avant de la réduction.

Les conclusions précédentes nous permettent d'identifier le besoin de formaliser des modèles de recherche de produits plus ciblés et moins orientés similarité. En effet, dans ce cas de requêtes orientées produits, l'objectif d'un modèle d'ordonnancement n'est donc pas, comme dans le cas d'une tâche de recherche d'information, d'identifier des produits fortement similaires (de marques différentes par exemples) qui gravitent autour de la description, mais plutôt de fournir à l'utilisateur l'ensemble des produits répondant spécifiquement au besoin. Par exemple, dans le cas de la requête « Playmobil », l'utilisateur attendra plus particulièrement tous les jouets de la gamme « Playmobil » et non des jouets génériques, tels que des bonhommes en plastiques du type soldat, animaux, etc... De même, un modèle retournant des produits favorisant la diversité des genres ou des âges recommandés ne sera probablement pas pertinent pour ce type de besoin, ce qui est différent en recherche d'information classique où l'utilisateur a besoin de documents diversifiés pour assurer une couverture du besoin en information.

## **5. Conclusion**

Dans cet article, nous nous sommes intéressés à comprendre les facteurs de pertinence pour la recherche de produits. Cette étude fait suite à notre participation à la campagne « Living Lab » qui propose un protocole expérimental novateur basé sur des utilisateurs en situation de recherche réelle. Lors de cette campagne d'évaluation, nous avons proposé un modèle probabiliste basé sur la recherche de produits reposant sur les caractéristiques descriptives des produits et l'engagement des utilisateurs. Les résultats ont montré l'efficacité de notre modèle vis-à-vis des modèles proposés par l'ensemble des participants.

Forts de ces résultats, nous nous sommes appuyés sur les fichiers *logs* de notre modèle afin d'identifier les facteurs de pertinence liés à la requête et aux produits. Les résultats n'ont montré aucune relation entre l'efficacité d'une requête et ses caractéristiques, suggérant une formulation du besoin utilisateur similaire à un besoin en information classique. De par la nature de la tâche orientée vers un acte d'achat de produits, nous avons pu révéler que les facteurs de pertinence sont fortement dépendant des caractéristiques des produits, plutôt que de la requête. Parmi les facteurs significatifs, nous avons identifié des facteurs liés à la description visuelle du produit (présence de photo), son an-

cienneté sur le marché et également son prix. Ce dernier révèle que l'utilisateur ne cherche pas nécessairement un produit similaire à la requête (et moins onéreux) comme dans le cas d'une recherche classique, mais plus particulièrement une gamme de produits. Ainsi, nous espérons que ces conclusions permettront de formaliser des modèles de recherche de produits plus ajustés aux spécificités de cette tâche.

## 6. Bibliographie

- Ben Jabeur L., Soulier L., Tamine L., « IRIT at CLEF 2015 : A product search model for head queries », *CLEF 2015 Workshop*, 2015.
- Cappellato L., Ferro N., Jones G., San Juan E. e. ., « CLEF 2015 Labs and Workshops, Notebook Papers », *CEUR Workshop*, 2015.
- Castagnos S., Jones N., Pu P., « Eye-tracking product recommenders' usage », *Proceedings of the fourth ACM conference on Recommender systems*, ACM, 2010.
- Chen L., « Social influence of product popularity on consumer decisions : usability study of Flickr camera finder », *IUI*, p. 297-300, 2010.
- Chen X., Wang H., Sun X., Pan J., Yu Y., « Diversifying Product Search Results », *SIGIR*, ACM, p. 1093-1094, 2011.
- Craswell N., Robertson S., Zaragoza H., Taylor M., « Relevance Weighting for Query Independent Evidence », *SIGIR*, ACM, p. 416-423, 2005.
- Dakka W., Gravano L., Ipeirotis P., « Answering General Time-Sensitive Queries », *IEEE ToKDE*, vol. 24, p. 220-235, 2012.
- Detlor B., Sproule S., Gupta C., « Pre-Purchase Online Information Seeking : Search versus Browse. », *J. Electron. Commerce Res.*, vol. 4, n° 2, p. 72-84, 2003.
- Duan H., Zhai C., Cheng J., Gattani A., « Supporting Keyword Search in Product Database : A Probabilistic Approach », *VLDB*, vol. 6, n° 14, p. 1786-1797, 2013.
- Ghirmatsion A. B., Balog K., « Probabilistic field mapping for product search », *CLEF 2015 Workshop*, 2015.
- Joachims T., « Optimizing Search Engines Using Clickthrough Data », *KDD*, p. 133-142, 2002.
- Lehmann J., Lalmas M., Yom-Tov E., Dupret G., « Models of User Engagement », *UMAP*, Springer, p. 164-175, 2012.
- Mothe J., Savoy J., Kamps J., Pinel-Sauvagnat K., Jones G., San Juan E., Cappellato L., Ferro N. e., « Experimental IR meets Multilinguality, Multimodality, and Interaction », *CLEF*, LNCS, vol. 9283, Springer, 2015.
- Schaer P., Tavakolpoursaleh N., « Historical Clicks for Product Search : GESIS at CLEF LL4IR 2015 », *CLEF 2015 Workshop*, 2015.
- Schuth A., Balog K., Kelly L., « Overview of the Living Labs for Information Retrieval Evaluation (LL4IR) CLEF Lab 2015 », *CLEF*, Springer, 2015.
- Vandic D., van Dam J.-W., Frasincar F., « Faceted product search powered by the Semantic Web », *Decision Support Systems*, vol. 53, n° 3, p. 425 - 437, 2012.
- Zaragoza H., Craswell N., Taylor M. J., Saria S., Robertson S. E., « Microsoft Cambridge at TREC 13 : Web and Hard Tracks. », Citeseer, 2004.

---

# RI-TAL : le TAL au service de la RI

**Laurent Candillier — Julien Hénot**

*TokTokTok , <https://toktoktok.com> , [prenom@toktoktok.com](mailto:prenom@toktoktok.com)*

---

*RÉSUMÉ. Comment le Traitement Automatique des Langues peut-il servir la Recherche d'Information? Cet article apporte des éléments de réponse à cette question dans le cadre de la mise en place d'un moteur de recherche au sein d'une application industrielle, TokTokTok, qui réunit un ensemble important de données hétérogènes sur des produits de tout type. Nous démontrons que l'enrichissement de la base de données par des traitements sémantiques améliore les résultats du moteur de recherche, mais que l'intégration de ces données sémantiques au coeur même du système de Recherche d'Information s'avère moins efficace. Utiliser le TAL en pré-traitement de la RI se révèle donc plus pertinent que son intégration plus profonde.*

*ABSTRACT. How may Natural Language Processing serve Information Retrieval? This paper provides some clues in the context of the implementation of a search engine within an industrial application, TokTokTok, that gathers a large set of heterogeneous data about products of many kinds. We demonstrate that the enrichment of the database by semantic treatments improves the results of the search engine. On the other side, the integration of these semantic data into the core of the Information Retrieval system proves to be less effective. Using the NLP as preprocessing of IR thus reveals itself as more relevant than its deeper integration.*

*MOTS-CLÉS : Recherche d'Information, Traitement Automatique des Langues, ElasticSearch, Application industrielle.*

*KEYWORDS: Information Retrieval, Natural Language Processing, ElasticSearch, Industrial Application.*

---

## 1. Introduction

Si l'utilisation des techniques de Traitement Automatique des Langues (TAL) pour aider à la Recherche d'Information (RI) convainc intuitivement nombre de chercheurs des deux domaines, leur intégration pratique s'est souvent heurtée à de bien moins bons résultats qu'attendus. Il y a vingt ans déjà, (Strzalkowski, 1995) menait une tentative relativement infructueuse et soulevait que non seulement l'utilisation de techniques de TAL dans la RI alourdisait considérablement les traitements en comparaison avec les techniques statistiques, mais aussi que dans de nombreux cas, ces traitements pouvaient dégrader les résultats. Un peu plus tard, (Smeaton, 1997) puis (Liddy, 1998) et (Brants, 2003) arrivaient à des conclusions similaires.

Le problème majeur lié à l'intégration des systèmes de TAL dans ceux de RI provient de la difficulté à gérer les ambiguïtés du langage, synonymies et polysémies en particulier. Pour contourner ce problème, les études citées précédemment semblent donc plaider pour un usage du TAL dans le cadre de l'enrichissement du contenu en pré-traitement de l'usage du système de RI, plutôt que l'intégration plus profonde des techniques de TAL au cœur même du système de RI, exception faite des traitements de plus bas niveau du TAL, à savoir les manipulations morphologiques des mots.

Plus récemment, (Moreau *et al.*, 2007) ont de nouveau mené une étude visant à formaliser l'évaluation de l'apport des techniques de TAL en RI dans un cadre unifié, et démontré que l'utilisation de ces techniques au niveau morphologique des textes améliore systématiquement les résultats, qu'au contraire, les traitements au niveau syntaxique n'apportaient jamais d'amélioration sur l'approche statistique, et que l'apport des traitements au niveau sémantique était plus mitigé. Cependant, si ces conclusions sont valides pour la RI sur des textes importants et avec des ressources générales comme *WordNet*, il est observé que le TAL pourrait toutefois être plus efficace dans des domaines d'application plus spécifiques (Bannour et Zargayouna, 2012).

### 1.1. Application

TokTokTok offre un service de livraison instantanée de tout produit que l'on peut trouver à Paris, Toulouse, Lyon ou Lille. Chez vous, chez des amis ou dans un parc, vous pouvez vous faire livrer quand vous le voulez un burger, un hamburger, des fleurs, des roses, du champagne, du mousseux, ou tout autre produit disponible à l'achat dans votre ville. Pour ce faire, TokTokTok doit regrouper un ensemble important de données hétérogènes, et y fournir un accès facile pour que les utilisateurs trouvent aisément ce dont ils ont envie au moment où ils se connectent au service, que ce soit via l'application mobile ou le site web. Deux approches peuvent être utilisées pour donner accès au catalogue de produits aux utilisateurs : la navigation au travers d'un arbre des catégories organisant les produits, ou la réponse à des requêtes ouvertes au travers d'un moteur de recherche.

La première difficulté à laquelle on se confronte intervient au niveau de la qualité des contenus associés aux différents produits formant la base de données, car ceux-ci sont souvent basiques (peu renseignés), mal organisés (des détails inutiles sur le produit sont présentés avant sa description générale), ou ambigus (les mots utilisés représentent mal le produit), tel ce jeu de société uniquement décrit par :

*Asmodée - ali baba. Ali baba accompagné de son dromadaire. Aide ali baba à charger son dromadaire. Contenu : 1 dromadaire, 20 cartes, 15 marchandises, 1 selle, ali baba et 1 notice. 2 à 4 joueurs. Age minimum requis 4 ans.*

Puis les difficultés classiques en Recherche d'Information apparaissent lors des traitements des requêtes des utilisateurs : gestion de la polysémie, de la synonymie, de la subsomption ou des erreurs orthographiques, typiquement ; et plus généralement, compréhension fine des attentes des utilisateurs. Pour bien répondre à la recherche d'un utilisateur, il faut en effet comprendre la différence entre son besoin et sa requête, le premier étant son objectif final alors que le second est le moyen d'atteindre son objectif, qui peut être formulé de différentes manières. Prenons l'exemple d'une recherche de *viande* par un utilisateur. Est-il en recherche de :

- 1) empanadas carne : viande de boeuf ?
- 2) boulettes de viande hachée ?
- 3) moussaka à la viande de boeuf ?
- 4) sauce tomate cuisinée à la viande ?
- 5) pizza bolognaise, viande de boeuf, tomate et fromage ?
- 6) onglet, viande bovine ?

Les résultats 2 et 6 sont sûrement ceux qui répondent le plus directement à la requête : de la viande, sans plus. Les résultats 1, 3 et 5 peuvent répondre à l'attente de l'utilisateur si celui-ci est en exploration : il sait qu'il veut un plat à base de viande mais ne sait pas lequel, car sinon, il aurait recherché *empanada*, *moussaka* ou *pizza*. Enfin le résultat n°4 répond moins probablement à la requête car même s'il contient bien de la viande, il ne s'agit que d'un composant d'un article différent : de la sauce. Par ailleurs, qu'en est-il du *bifteck extra tendre de charal*, pour lequel le mot viande n'apparaît pas, ou du *couscous viande et légume*, qui contient une erreur d'orthographe ?

Dans cet article, nous présentons nos travaux menés dans les domaines du TAL et de la RI dans ce cadre d'application industrielle. Nous montrons que l'enrichissement de la base de données par l'association automatique des produits aux concepts d'une ontologie dédiée améliore de manière significative les résultats du moteur de recherche, mais qu'à l'inverse, nos tentatives pour rendre ce moteur de recherche sémantique par l'utilisation de ces mêmes informations ontologiques au niveau de la transformation des requêtes aboutissent à des résultats moins pertinents.

Nous organisons notre article comme suit : dans la section 2 nous décrivons nos travaux en Traitement Automatique des Langues portant sur la catégorisation automatique des produits du catalogue de TokTokTok ; puis dans la section 3 nous abordons

la partie Recherche d'Information de nos travaux, en présentant tout d'abord comment l'outil *ElasticSearch* a été intégré au projet, et comment nous avons traité le problème de la correction orthographique, avant d'aborder le sujet de l'intégration des données riches de notre ontologie au sein du moteur de recherche qui serait alors plus sémantique.

## 2. TAL : catégorisation automatique des produits

### 2.1. Ontologie

Contrairement à nombre d'acteurs du secteur de la livraison instantanée qui se focalisent sur la livraison de plats à emporter, TokTokTok propose à la livraison tout type de produit qu'il est possible d'acheter en ville : il recense donc non seulement les plats à emporter des restaurants de chaque ville, mais aussi l'ensemble des produits des minimarchés (boîtes de thon, serviettes, produits ménagers...), des magasins de mode (pantalons, maillots, gants...), de produits sportifs (raquettes, ballons...), de haute technologie (téléphones, câbles, disques durs...), et bien plus encore.

Le nombre de catégories représentant les dizaines de milliers de produits pouvant se retrouver dans la base de données de TokTokTok est donc très important, de même que leur diversité. Nous avons identifié 9 catégories principales : *alimentation, boisson, beauté, santé, maison, high-tech, loisir, mode et fleurs*. Puis 122 sous-catégories leur ont été associées. Au total, 1827 catégories sont aujourd'hui présentes dans notre hiérarchie de catégories de profondeur maximale 4. L'établissement de cette ontologie a été de longue haleine et est allée de pair avec la mise au point de notre algorithme de catégorisation automatique.

### 2.2. Algorithme

À partir des données associées à un produit, à savoir son titre, sa marque, sa description et éventuellement une liste de tags, nous devons être capables d'associer automatiquement le produit à l'une des catégories de notre ontologie. Mais nous devons opérer avec des contenus très divers : parfois seul le titre est renseigné (*cheese delice*), parfois c'est dans la description que se trouve l'information de catégorie (pour les vins par exemple), et surtout, de nombreuses ambiguïtés peuvent apparaître : un *crayon* peut par exemple être associé à la catégorie *maison – papeterie*, ou à *beauté – visage* ; la *tomate* doit être différenciée de la *sauce tomate* ou de la *salade tomate* ; le *jean* doit être différencié de *Jean Foillard* ; etc.

Aucune méthode existante en Traitement Automatique des Langues ne répond complètement à notre problématique, tout d'abord parce que nos données couvrent l'ensemble des produits qui peuvent être achetés alors que de nombreuses méthodes s'appliquent sur des données appartenant à un domaine spécifique, ensuite parce que nos contenus représentent des textes de très petite taille comparés aux corpus qui sont

généralement utilisés dans le domaine, enfin parce qu’au début de notre étude, nous ne disposions pas d’une base d’apprentissage qui aurait été suffisante pour couvrir l’ensemble des associations entre produits et catégories qui permette à une approche automatique d’opérer efficacement.

Pour mener à bien cette étude, nous avons commencé par instancier une telle base d’apprentissage : plusieurs produits ont été manuellement catégorisés afin d’évaluer la pertinence des algorithmes mis au point. Composée de quelques dizaines d’exemples seulement au début de l’étude, cette base d’apprentissage réunit aujourd’hui plusieurs milliers d’exemples, qualifiés par étapes en fonction des avancées de la recherche présentée ici.

Des algorithmes très simples ont été développés dans un premier temps pour servir d’étalon : par exemple, on peut simplement déterminer la catégorie d’un produit en fonction du premier mot rencontré dans son titre qui appartienne à l’une des catégories de l’ontologie (mais cette approche associera le CD *Double chill burger* d’*Akhenaton* à la catégorie *burger*) ; une autre approche relativement simple consiste à compter le nombre d’occurrences de chacun des mots de l’ontologie dans le titre des produits, et à sélectionner la catégorie la plus représentée (cette solution associera cependant un *burger tomate sauce tomate* à la catégorie *tomate*) ; cette solution peut ensuite être affinée en pondérant l’importance de chaque mot en fonction de sa position dans le titre (*burger* ayant alors plus de poids que *tomate* dans l’exemple précédent) ; enfin, on peut chercher à établir la catégorie principale d’un produit avant de l’associer à l’une de ses sous-catégories (ainsi, si davantage de mots font référence à la musique qu’à l’alimentation dans le cas du CD d’*Akhenaton*, alors l’erreur d’association avec la catégorie *burger* sera évitée). Évidemment, ce traitement effectué au niveau du titre d’un produit peut de même être effectué au niveau de sa description, du nom de la marque associée, ou de sa liste de tags. Le choix du poids associé à chacun de ces champs a cependant son importance : c’est en effet souvent le titre d’un produit qui contient ses informations les plus pertinentes, bien que parfois ces informations soient présentes dans les autres champs.

### 2.3. Racinisation

Un premier problème de base qu’il a fallu régler pour nous lancer dans ces Traitements Automatiques des Langues concerne la racinisation (ou stemming) des mots qui composent notre langage. En effet, que l’on parle de *burger* ou de *burgers* au pluriel dans un titre de produit ne doit pas en modifier sa catégorisation automatique. Nous avons donc mené une première étude sur les solutions utilisables à ce niveau. Nous avons comparé trois solutions : le stemmer de *Porter*, le plus reconnu et utilisé bien que dédié à la langue anglaise ; le stemmer de *Paice-Husk*, dédié au français ; et l’utilisation du lexique libre pour le français *Morphalou*. Les deux premières solutions sont basées sur des approches automatiques pour la racinisation des mots, les transformant en utilisant des règles de suppressions de préfixes ou de suffixes, alors que la dernière solution se base sur une association directe entre différentes assertions d’un

mot : *burger* et *burgers* par exemple. Au final, cette dernière solution nous a apporté les résultats les plus satisfaisants, non seulement au niveau du taux d'erreur calculé sur notre base d'apprentissage, mais aussi dans notre objectif de maîtriser au mieux l'ensemble de notre processus de catégorisation automatique. En effet, un inconvénient majeur des algorithmes de racinisation automatiques est qu'il est très difficile de contrôler leurs erreurs.

L'intégration du lexique Morphalou s'est donc faite effective au sein de notre projet, et nous nous sommes lancés dans son affinage pour faire face à notre problématique. Notre lexique s'organise donc comme suit : chaque mot est associé à son lemme, c'est-à-dire son assertion de base : *burger* est ainsi le lemme associé aux mots *burger* et *burgers* ; et chaque lemme peut être associé à une catégorie : le lemme *burger* est ainsi associé à la catégorie *alimentation – fast food – burger*. Pour affiner ce lexique, nous nous sommes dans un premier temps appuyés sur les cas d'erreur que nous détectons dans notre base d'apprentissage. Les lemmes *hamburger*, *cheeseburger* ou *bigmac* ont ainsi été associés à la catégorie *burger*, par exemple. L'association entre le mot *poivrons* et le lemme *poivrer* a été modifiée pour préférer une association avec le lemme *poivron*, plus appropriée dans notre cadre. Nous avons également enrichi notre lexique en créant des exceptions qui spécifient quelles associations entre lemmes doivent mener vers certaines catégories plutôt que d'autres : par exemple, si le lemme *eau* est par défaut associé à la catégorie *boisson – eau*, l'association entre les lemmes *eau* et *apaiser* mène, elle, à la catégorie *beauté – visage – lotion*.

## 2.4. Résultats

La méthode de catégorisation automatique a alors pu être affinée à son tour. Les meilleurs résultats ont été obtenus en considérant les titres des produits comme prioritaires dans les décisions de catégorisation, puis les descriptions, suivies des tags (ces derniers étant de moindre confiance, mais apportant parfois l'information qu'il manque dans le reste des champs), suivis des marques (celles-là se révélant souvent trop générales, et ambiguës, *nestlé* pouvant aussi bien proposer des boissons que de l'alimentation, et *philips* proposant aussi bien des produits high-tech que de l'électroménager de maison). La technique de pondération des mots en fonction de leur position dans le texte s'est révélée positive, et la fonction de pondération a été affinée sur la base d'apprentissage. La technique de recherche initiale de la catégorie principale des produits, avant leur affinage en sous-catégories, a également démontré son intérêt, mais elle a aussi soulevé certains cas problématiques : dans l'alimentation, les ingrédients d'un plat principal peuvent alors prendre le pas sur le plat en lui-même : une *lasagne* décrivant l'ensemble de ses ingrédients peut alors se retrouver associée à la catégorie *légume*. Pour contourner ce problème, nous avons ajouté une technique de priorisation des catégories entre elles : un poids supplémentaire est ainsi associé à des catégories générales comme *lasagne*, *burger* ou *sandwich*, tandis que les ingrédients comme les *légumes* ou les *épices* se voient attribuer un poids moins important.



Ce système, affiné sur la base d'apprentissage, nous a alors permis d'améliorer encore notre prédiction en catégorisation automatique.

En parallèle de ces recherches, la base d'apprentissage s'est peu à peu enrichie de nouveaux exemples, rendant l'algorithme plus précis et général, et soulevant de nouveaux cas spécifiques nous permettant d'affiner notre ontologie ainsi que notre lexique et sa base d'exceptions associée. Un nouvel affinage de l'algorithme a alors vu le jour : un poids a été associé à l'émergence d'une exception en fonction de la proximité entre les lemmes représentant l'exception. Ainsi, l'association entre *crème* et *frais*, menant à la catégorie *alimentation - crèmerie - crème fraîche*, ne s'active que si les termes sont relativement proches, laissant la *crème du soir qui vous donne un teint frais* s'associer à la catégorie *beauté - soin du corps – crème*.

Nous avons atteint à ce jour la quantité de 1827 catégories, 2751 lemmes associés à ces catégories, et 1213 exceptions gérant des associations spécifiques entre lemmes devant mener à une correction de leur association aux catégories. Notre base d'apprentissage réunit 5094 exemples. 93,56% d'entre eux sont correctement associés aux catégories de premier niveau de notre arbre. 82,55% sont correctement associés au deuxième niveau de l'arbre. 83,30% au troisième niveau. 90,67% au quatrième niveau. Et le *bifteck extra tendre de charal*, donné en exemple précédemment, est désormais bien associé à la catégorie *alimentation - boucherie*.

### 3. RI : moteur de recherche de produits

#### 3.1. Données

TokTokTok réunit donc des données hétérogènes sur des dizaines de milliers de produits de tout type issus de plusieurs milliers de magasins, et notre algorithme de catégorisation automatique associe chacun d'entre eux à une entrée de notre ontologie. Cette association permet déjà de fournir un premier accès au catalogue, pertinent pour les utilisateurs : à travers le parcours de l'arbre des catégories, il peut décider de manger > du fast food > de la pizza, et entrer ainsi dans un univers précis. Si cette façon de naviguer est assez suivie sur mobile, c'est moins le cas pour la version web du site, sur laquelle les utilisateurs ont davantage tendance à utiliser le moteur de recherche pour préciser leurs attentes. La mise en place d'un système de Recherche d'Information efficace est donc indispensable.

Dans le même objectif de rigueur cherchant à assurer que la solution mise en place pour cela soit pertinente, nous avons démarré notre étude dans ce deuxième domaine de la même façon que dans le premier : en établissant un jeu de données qui nous permette d'évaluer la précision des méthodes développées. Celui-ci est structuré de la manière suivante :

- les requêtes les plus souvent émises sur notre moteur de recherche ont été sélectionnées ;

- pour chacune d’entre elles, un ensemble de produits répondant ou non à la requête ont été choisis ;

- et à chacun de ces exemples de couples requête - produit, un score de pertinence a été attribué, compris entre -1 et 1 ; un score négatif signifie que le produit n’est pas considéré comme pertinent pour la requête tandis qu’un score positif signifie au contraire que le produit est attendu comme résultat de la requête ; enfin, on offre la possibilité d’accorder plus ou moins d’importance à un résultat donné en permettant d’attribuer des scores inférieurs à 1 aux exemples : si l’utilisateur cherche des *frites*, étant donné que le catalogue est rempli de restaurants offrant des *frites*, le *coupe frites* n’est pas considéré comme répondant de manière pertinente à la recherche, son score associé est donc négatif, mais cette réponse n’étant pas non plus complètement absurde, nous préférons par exemple lui associer un score de -0,2 plutôt que -1.

171 requêtes ont ainsi été listées, associées à 1346 produits attendus ou non en résultats. Différents types de requêtes ont été considérées :

- certaines plutôt générales : *fast food / vetement* ;
- certaines concernant des marques : *big fernand / studio 5* ;
- certaines plus spécifiques : *android / evian 1L* ;
- certaines contenant des fautes d’orthographe : *buger / citronade* ;
- et certaines faisant davantage référence à des attributs de produits : *halal / gouter*.

Nous avons adopté le score DCG (Discounted Cumulative Gain) pour évaluer la pertinence d’une solution. Ce score est largement utilisé dans le domaine de la Recherche d’Information, mais il ne couvre pas la possibilité d’exclure des résultats de recherches, ce que nous codons par nos scores négatifs. Nous l’avons donc adapté pour y ajouter cette possibilité. Notons  $E$  l’ensemble de nos exemples  $e_i$  de triplets (requête  $r_i$ , produit  $p_i$ , score  $s_i$ ). Notons  $K_S(e_i)$  le rang attribué au produit  $p_i$  pour la requête  $r_i$  par le système de RI à évaluer, noté  $S$ . Alors, le score de pertinence du système  $S$  par rapport au jeu de données  $E$  est donné par :

$$DCG_{S,E} = \sum_{e_i \in E} \text{sign}(s_i) \times \frac{2^{|s_i|} - 1}{\log_2(K_S(e_i) + 1)}$$

Si le système ne renvoie pas le produit  $p_i$  parmi les 20 premiers résultats pour la requête  $r_i$ , alors le  $\text{sign}(s_i)$  est inversé, et on pose  $K_S(e_i) = 1$ . Ainsi, si le produit ne devait pas être retrouvé pour la requête, le DCG est incrémenté de manière optimale, et inversement, s’il devait être retrouvé mais ne l’est pas, le DCG est décrémenté au maximum.

### 3.2. Algorithme

Nous avons décidé de nous baser sur le système de RI libre *ElasticSearch* pour développer le moteur de recherche de TokTokTok et mener cette étude.

Dans la littérature en Recherche d'Information, on relève typiquement les problèmes suivants pour faire face à la bonne compréhension des besoins des utilisateurs (Manning *et al.*, 2008) :

- 1) considérer différentes assertions d'un terme : le pluriel *burgers* de *burger*, typiquement ;
- 2) traiter les fautes d'orthographe : *buger* au lieu de *burger* par exemple ;
- 3) reconnaître des synonymes, ou mots de la même famille : *hamburger* et *burger* ;
- 4) comprendre les concepts de la recherche : un *boeuf* s'avérant une bonne réponse pour une recherche de *viande* par exemple ;
- 5) et traiter la requête comme un tout : faire la différence entre une *sauce tomate* et un *poulet en sauce aux épices accompagné de tomates*, par exemple.

La racinisation des mots, vue dans la partie précédente, répond à la première problématique, mais sa version automatique n'est pas toujours reconnue comme pertinente, en particulier parce qu'elle peut faire apparaître des fausses corrections. Des approches sémantiques sont nécessaires pour répondre aux problématiques 3 et 4. Le calcul de proximité entre termes peut être utilisé pour répondre au dernier point.

Mais c'est le problème de la correction orthographique qui connaît une attention toute particulière dans la littérature du domaine. Des analyses montrent qu'entre 10 et 12% des termes fournis en entrée des requêtes sur les moteurs de recherche sont mal orthographiés, et que les contenus des pages contiennent également des erreurs (Martins et Silva, 2004). Les corrections automatiques sont donc nécessaires et doivent être transparentes pour l'utilisateur. Les solutions classiques pour y faire face sont les suivantes :

- 1) utiliser la distance d'édition entre mots, qui compte le nombre de transformations nécessaires pour passer d'un mot à un autre : la distance d'édition entre *buger* et *burger* se révélant par exemple réduite : une seule transformation est nécessaire pour passer de l'un à l'autre : ajouter un r ;
- 2) utiliser les n-grams : un mot est alors divisé en groupes de n caractères, et la distance entre deux mots correspond à la proportion de ces groupes qui sont identiques : pour n=3, *buger* est ainsi divisé en *bug* + *uge* + *ger*, alors que *burger* est divisé en *bur* + *urg* + *rge* + *ger*, les deux mots partageant donc seulement le 3-grams *ger* ;
- 3) utiliser les corrections phonétiques : *burrger* se révélant alors complètement similaire à *burger* par exemple ;
- 4) enfin la correction entre termes peut être effectuée à partir d'un corpus : dans ce cas, on analyse les logs de requêtes des utilisateurs, et si deux requêtes de *buger* puis *burger* se succèdent fréquemment, alors on en déduit que *burger* correspond à la correction de *buger*.

L'atout important de cette dernière approche réside dans son aspect universel : elle permet de faire face à un lexique qui évolue rapidement, avec des abréviations qui peuvent être largement partagées, ou des noms de personnalités ou de marques qui

peuvent apparaître alors qu’elles ne seraient pas présentes dans un lexique de base (Cucerzan et Brill, 2004). Cette approche est cependant prématurée pour TokTokTok car elle requiert de nombreuses données de recherches utilisateurs, données dont nous ne disposons pas encore en quantités suffisantes. Aussi, ElasticSearch fournit des outils permettant de mettre en place les deux premières solutions citées : la distance d’édition et les n-grams. Nous nous sommes donc focalisés sur ces solutions.

Les informations recherchées par les utilisateurs de TokTokTok peuvent se trouver dans plusieurs champs : les titres de produits, leur description, leurs tags, le nom de leur marque, ou bien le nom de leur catégorie, ou d’une catégorie parente.

Les requêtes comme les contenus des produits sont naturellement normalisés : tous encodés dans le même système, en lettres minuscules, et séparés par les caractères non alphanumériques qui les composent. Viennent alors les choix plus fins de formatage : utilise-t-on la racinisation des mots ? élimine-t-on les mots creux du langage (*stop words*) ? utilise-t-on une représentation par n-grams ? ou la distance d’édition ?

Dans un premier temps, nous avons comparé ces différentes solutions, mais rapidement, nous avons fait le choix de les mixer car elles sont complémentaires. Chaque requête utilisateur a donc été confrontée à tous les champs des produits (titre, description, tags, marque, catégories), formatés de différentes façons (avec ou sans racinisation, élimination des mots creux et technique de n-grams ou de distance d’édition). L’inconvénient de cette approche est que chaque relation entre un champ et un formatage augmente d’autant le temps d’exécution de la solution. En particulier, les approches n-grams et par distance d’édition sont couteuses en temps d’exécution. Aussi, le poids des méthodes de corrections orthographiques est parfois trop important dans ce cadre, renvoyant des résultats de *gin* ou d’*orangina* pour une recherche de *gini*.

En fait, il existe naturellement des priorités entre ces relations entre champs et formatages : une adéquation entre la requête et le titre peu formaté d’un produit est logiquement plus importante qu’une adéquation avec un tag formaté par n-grams. Des *boosts* (ou poids) ont donc été associés à ces relations, et optimisés par validation croisée face à notre ensemble d’apprentissage (le jeu de données de tests).

16 clauses ont ainsi été finalement sélectionnées. Le tableau 1 présente les caractéristiques retenues pour encoder notre système de RI. Pour chaque champ associé aux produits, il détaille :

- le nombre de clauses retenues qui y sont associées ;
- la somme des poids associés à ces clauses ;
- la quantité de ces clauses utilisant le stemming (racinisation automatique) ;
- la quantité de ces clauses utilisant la suppression de mots creux (stopwords) ;
- la quantité de ces clauses utilisant la distance d’édition (technique de correction orthographique) ;
- et la quantité de ces clauses utilisant la technique de n-grams (pour la correction orthographique).

| champ       | nb clauses | poids | stemming | stopwords | édition | n-grams |
|-------------|------------|-------|----------|-----------|---------|---------|
| titre       | 4          | 8     | 3        | 3         | 1       | 1       |
| description | 2          | 2     | 1        | 1         | 0       | 0       |
| tags        | 3          | 3     | 0        | 0         | 1       | 2       |
| marque      | 4          | 6     | 2        | 2         | 2       | 0       |
| catégories  | 3          | 5     | 1        | 1         | 2       | 0       |

**Tableau 1.** Configuration optimale du système de RI.

On remarque ainsi que le système considère comme prioritaires les informations liées aux titres des produits : 4 clauses y sont associées, avec des poids doublant leur importance. Ce sont ensuite les informations de marques qui prennent le plus de poids : 6, pour 4 clauses. Puis les informations de catégories, déduites par Traitement Automatique des Langues, révèlent une grande importance, boostées à 5 pour 3 clauses, alors que les informations de tags et de descriptions ne réunissent que 3 puis 2 clauses, respectivement, et qu’aucun poids supplémentaire ne leur est attribué.

La racinisation des mots et le filtrage des mots creux sont utilisés une fois sur deux, confirmant leur intérêt, mais relativisant l’importance qu’il faut leur accorder, et confirmant surtout l’intérêt de mixer les approches pour bénéficier des atouts de chacune. Elles ne semblent toutefois pas nécessaires au niveau des informations de tags, ce qui se comprend relativement bien car il s’agit justement de listes de mots déjà naturellement formatés.

Enfin, l’importance des techniques de corrections orthographiques est aussi mise en avant par leur utilisation intensive. La technique par distance d’édition semble prendre plus de poids que la technique par n-grams, mais dans deux cas (pour les titres et les tags), elles sont utilisées de manière complémentaire. On peut s’étonner qu’elles ne soient pas utilisées au niveau des informations de descriptions des produits, on peut même s’étonner plus largement du faible poids attribué à ces données descriptives dans l’ensemble du système, mais cela peut s’expliquer par la présence d’un contenu de plus faible confiance. En effet, dans certains cas, ces informations sont redondantes avec celles du titre ; parfois elles ne sont pas directement liées au produit lui-même mais plutôt à sa marque ; parfois aussi, elles utilisent des termes qui décrivent mal le produit, et qui sont sujet à davantage d’ambiguïtés que dans le reste du corpus, contrairement aux catégories, qui auront aidé, grâce à nos travaux de TAL, à lever les ambiguïtés, comme pour les vins par exemple.

### 3.3. Résultats

Au final, notre système de RI configuré ainsi atteint le score suivant :

$$DCG_{S,E} = 596,4$$

| champs supprimés         | $DCG_{S,E}$  |
|--------------------------|--------------|
| aucun                    | <b>596,4</b> |
| titre                    | 57,7         |
| description              | 508,2        |
| tags                     | 515          |
| marque                   | 409,5        |
| <b>catégories</b>        | <b>389,2</b> |
| catégories & titre       | -162,4       |
| catégories & description | 267,8        |
| catégories & tags        | 294,2        |
| catégories & marque      | 198,1        |

**Tableau 2.** Résultats du système de RI selon la configuration.

À noter qu'il nécessite seulement 4,7 secondes pour traiter les 171 requêtes du jeu de données test, ce qui le rend tout à fait utilisable en pratique, et ce malgré les coûts accrus requis par les techniques de corrections orthographiques. Et à comparer au maximum possible de ce score pour un système  $S^*$  qui serait complètement adapté au jeu de données  $E$  :

$$DCG_{S^*,E} = 739,2$$

Si maintenant on retire du système les informations issues de l'étape précédente de TAL, à savoir les informations de catégories des produits, le score  $DCG_{S,E}$  chute à 389,2. Pour apprécier plus formellement l'apport de chaque champ associé aux produits dans les performances du système, nous présentons dans le tableau 2 les scores qu'il atteint lorsque les contenus de chacun de ces champs sont masqués.

On remarque alors que les informations contenues dans les titres des produits sont de loin les plus pertinentes pour répondre aux requêtes des utilisateurs, puisque sans elles, le système chute à un score proche de zéro de 57,7. À l'inverse, si les informations de descriptions ou de tags ne sont pas considérées, le score diminue de moins de 100 points par rapport à l'optimal. Les informations de marques sont davantage utilisées car sans elles, le score diminue de manière plus conséquente. Mais après les informations contenues dans les titres des produits, ce sont celles contenues dans les catégories qui sont les plus pertinentes, car leur absence fait chuter le score  $DCG_{S,E}$  à 389,2, soit moins de 200 points par rapport à l'optimal.

Nous avons répété ces expériences en observant l'effet de la suppression de chacun des champs associés aux produits en plus de la suppression des informations issues de nos traitements de TAL, et les conclusions se répètent : en particulier, on observe que la suppression des informations contenues dans les titres de produits combinée à l'éviction de celles contenues dans les catégories aboutit à des résultats très mauvais du système sur le jeu de données, qui affiche un score négatif :  $DCG_{S,E} = -162,4$  ; si l'on compare ce chiffre aux 57,7 points du système n'utilisant pas les informations

contenues dans les titres, on remarque que les informations de catégories avaient pour partie compensé cette perte de manière conséquente : plus de 200 points.

On démontre donc ici que nos Traitements Automatiques des Langues enrichissent de manière conséquente la base de données de produits de TokTokTok, de telle manière que le système de Recherche d'Information se voit ensuite amélioré de manière significative par l'utilisation de ces informations au niveau du traitement des requêtes utilisateurs.

### 3.4. RI sémantique

Notre système de RI appliqué à notre base de données enrichie par TAL a ainsi atteint des performances tout à fait intéressantes. Il est capable de traiter de manière pertinente des requêtes générales portant sur les catégories de produits ou les marques de notre catalogue, et répond également de manière pertinente à des requêtes plus spécifiques, et ce même si des fautes d'orthographe sont présentes dans la requête ou le contenu.

La plupart des requêtes que nous traitons sont des requêtes simples, courtes, et nous remarquons que notre système réagit moins bien en cas de requêtes plus complexes, plus longues. Aussi, si nous prenons l'exemple d'une recherche de *pull coton*, si bien les premiers résultats concernent effectivement les *pulls en coton* de notre catalogue, la suite des résultats est moins satisfaisante car elle mélange les résultats de *pulls* avec ceux de *coton-tiges* ou de *gants en coton*. Or, naturellement, on s'attendrait plutôt à ce que la suite des résultats concerne davantage les *pulls* que les *cotons* car le *pull* est l'objet principal de la recherche, le *coton* n'étant qu'une qualité associée à cet objet principal. On retrouve cette problématique dans d'autres exemples comme la recherche de *burger bio* ou de *glace vanille*.

Dans le cas du *burger bio*, la solution pourrait venir du fait que *burger* est identifié comme une catégorie de notre ontologie alors que *bio* non. On pourrait alors donner plus de poids au terme *burger*, ou même rechercher directement les produits dont la catégorie est *burger*. Cependant, pour la *glace vanille*, le traitement est plus compliqué car la *vanille* correspond également à une catégorie de notre ontologie. Une solution serait alors de donner la priorité aux premiers mots de la requête qui sont détectés comme faisant partie de l'ontologie. On pourrait aussi s'appuyer sur les priorités entre catégories établies à l'étape de TAL pour faire le choix. Cependant, *pull* et *coton* sont aussi tous deux des catégories de l'ontologie, mais jusqu'à présent, aucune valeur de priorité n'a été attribuée à l'une sur l'autre. Et surtout, on voit bien ici que le système se complique rapidement ainsi.

Enfin, nous avons également cherché à traiter de manière plus sémantique les requêtes contenant des noms de marques, l'idée intuitive étant toujours la même : si je sais identifier que dans la requête *burger big fernand*, *big fernand* correspond à la marque recherchée, et *burger* est l'objet recherché, une catégorie d'objet recherché en l'occurrence, alors je peux coder cette compréhension de l'attente de l'utilisateur

en une requête plus précise, et éviter de retourner des résultats non pertinents du type *CD Double chill burger, a big hit from my friend Fernand*. Cependant, dans cette tentative, nous nous sommes de nouveau confrontés à des effets de bord indésirables : la recherche de *crème solaire*, par exemple, n’aboutissait plus à aucun résultat car la *crème* était identifiée comme une catégorie de l’ontologie, et *solaire* était identifié comme une marque.

Ce que nous retenons de cette série d’expériences est que la transformation de notre moteur de Recherche d’Information en un moteur qui semblerait plus puissant de Recherche d’Information Sémantique se fait au détriment d’une flexibilité naturelle du système classique statistique. L’approche sémantique est trop contraignante et oblige à trop d’effort pour traiter tous les cas d’ambiguïtés présents dans les requêtes. Au final, le meilleur résultat que nous avons atteint avec nos différentes tentatives de moteurs de recherche sémantiques  $S^S$  est bien inférieur au résultat du moteur de recherche initial utilisant simplement la base enrichie par le TAL ; il est même moins bon que si l’on n’utilise aucune information supplémentaire issue de TAL :

$$DCG_{S^S, E} = 343, 1$$

## 4. Conclusions et perspectives

### 4.1. RI-TAL

Utiliser le TAL pour enrichir la base de données de produits de TokTokTok et mettre en place un système de RI optimisé sur ces données fournit d’excellents résultats. Aujourd’hui, la recherche de produits dans le catalogue se fait de manière pertinente, les erreurs d’orthographe sont bien traitées, ainsi que les recherches conceptuelles : l’exemple problématique présenté au début de cet article sur la *viande* est résolu.

Nos tentatives pour rendre le moteur davantage sémantique se sont par contre heurtées à de bien moins bons résultats, et nous pensons que cette observation va dans le sens de ce qui est généralement défendu dans les recherches du domaine : le TAL échoue à améliorer la RI si on tente de l’intégrer de manière trop profonde dans le système, alors qu’il s’avère tout à fait positif s’il se contente d’accompagner le processus de RI par des traitements sur le contenu en amont.

Ce dernier point, évoqué par (Brants, 2003) ou (Bannour et Zargayouna, 2012), a en effet été démontré dans nos expériences : si bien l’intégration d’informations sémantiques dans un cadre général (typiquement avec des ontologies de type WordNet) ne permet pas d’améliorer de manière significative les résultats d’un système de RI, lorsque ces informations sémantiques sont spécifiques à un domaine (médical ou juridique par exemple), alors les résultats sont plus probants. Nous démontrons ici qu’il en est de même dans notre cadre d’application industrielle spécifique.



## 4.2. TAL

Pour approfondir davantage notre étude sur la catégorisation automatique de produits, nous pourrions maintenant considérer de comparer notre approche à des approches plus automatiques, car nous disposons désormais d'une base d'apprentissage plus conséquente qui nous le permet. L'approche par plus proches voisins de (Rosso *et al.*, 2004) pourrait par exemple être testée.

Cependant, notre méthode actuelle possède un avantage important sur ces approches automatiques qui nécessitent des exemples d'apprentissage : il s'agit de la facilité de sa généralisation au cas d'autres langues. En effet, nous avons déjà généralisé notre méthode au cas de l'anglais, à l'aide d'un simple dictionnaire de traductions mot à mot, et au faible coût de quelques affinages de traductions spécifiques à notre cas d'application.

En fait, nous pensons que notre base d'apprentissage pourrait plus efficacement nous aider à affiner nos lexique et ontologie. En détectant les mots les plus fréquents de notre collection qui ne sont pas encore dans notre ontologie, celle-là pourrait s'affiner de manière semi-automatique. Inversement, il pourra être intéressant d'analyser les catégories peu ou pas utilisées dans notre ontologie, pour la remettre en question. Aussi, et surtout, davantage d'associations entre lemmes et catégories pourraient être proposées automatiquement afin d'améliorer notre système. Enfin, l'analyse des requêtes postées par nos utilisateurs pourrait être approfondie afin d'affiner encore nos lexique et ontologie en fonction de leurs attentes.

## 4.3. RI

En ce qui concerne la partie Recherche d'Information de notre étude, nous pourrions dans un premier temps formaliser davantage notre jeu de données de couples requêtes - produits. En effet, ce jeu de données a été créé manuellement par seulement deux annotateurs, et sans qu'ils travaillent sur les mêmes requêtes. Or on sait que les attentes de résultats de requêtes sont hautement subjectives. Une solution consisterait donc à demander à un ensemble plus conséquent d'annotateurs de fournir leurs attentes sur l'ensemble des requêtes du jeu de données. L'autre solution consiste à récupérer automatiquement ces évaluations implicites faites par les utilisateurs du service à partir de logs (Joachims *et al.*, 2005). En créant un outil de suivi des actions des utilisateurs de TokTokTok, nous pourrions mettre en relation leurs requêtes avec leurs actions sur les produits proposés : de la visualisation du produit à son achat. Un tel jeu de données serait alors plus conséquent, davantage focalisé sur les attentes réelles des utilisateurs, et en constante évolution.

Nous pourrions ensuite travailler sur trois aspects connexes aux travaux du domaine, à savoir :

- 1) l'affinage de requêtes : proposer des réécritures de requêtes pour aider les utilisateurs à préciser leur attente ;

2) la diversification des résultats : éviter de n'avoir que des *burgers* comme résultats d'une recherche de *fast food*, mais plutôt retourner un éventail de l'offre du catalogue : *pizzas, hot dogs, empanadas, etc.* ;

3) la recherche personnalisée via la recommandation : adapter les résultats de la recherche en fonction de chaque individu, en s'appuyant sur des profils utilisateurs, ce qui permettrait de différencier les résultats entre un utilisateur qui préférerait le *coca-cola* et un autre le *pepsi*, même si les deux formulent leur requête par *soda*.

## 5. Bibliographie

- Bannour I., Zargayouna H., « Une plate-forme open-source de recherche d'information sémantique », *Actes de la neuvième Conférence en Recherche d'Information et Applications (CORIA)*, p. 167-178, 2012.
- Brants T., « Natural Language Processing in Information Retrieval », *CLIN*, Google Inc., 2003.
- Cucerzan S., Brill E., « Spelling Correction as an Iterative Process that Exploits the Collective Knowledge of Web Users », *EMNLP*, vol. 4, p. 293-300, 2004.
- Joachims T., Granka L., Pan B., Hembrooke H., Gay G., « Accurately interpreting clickthrough data as implicit feedback », *28th annual international SIGIR Conference on Research and Development in Information Retrieval*, p. 154-161, 2005.
- Liddy E. D., « Enhanced text retrieval using natural language processing », *Bulletin of the American Society for Information Science and Technology*, vol. 24, n° 4, p. 14-16, 1998.
- Manning C. D., Raghavan P., Schütze H., *Introduction to information retrieval*, vol. 1, Cambridge university press Cambridge, 2008.
- Martins B., Silva M. J., « Spelling correction for search engine queries », *Advances in Natural Language Processing*, Springer, p. 372-383, 2004.
- Moreau F., Claveau V., Sébillot P., « Intégrer plus de connaissances linguistiques en recherche d'information peut-il augmenter les performances des systèmes ? », *Actes de la quatrième Conférence en Recherche d'Information et Applications (CORIA)*, p. 223-238, 2007.
- Rosso P., Ferretti E., Jimenez D., Vidal V., « Text categorization and information retrieval using wordnet senses », *The Second Global Wordnet Conference GWC*, 2004.
- Smeaton A. F., « Using NLP or NLP resources for information retrieval tasks », *Natural language information retrieval*, Kluwer Academic Publishers, p. 99-111, 1997.
- Strzalkowski T., « Natural language information retrieval », *Information Processing & Management*, vol. 31, n° 3, p. 397-417, 1995.

---

# Répondre à des requêtes cliniques PICO<sup>1</sup>

**Eya Znaïdi\*** — **Lynda Tamine\*** — **Chiraz Latiri\*\***

\* IRIT, Université de Toulouse, CNRS, INPT, UPS, UT1, UT2J, France, 118 Route Narbonne, Toulouse, France

{znaïdi,tamine}@irit.fr

\*\* Laboratoire LIPAH, Faculté des Sciences de Tunis, Campus Universitaire Tunis El Manar, 1060 Tunis, Tunisie

{chiraz.latiri}@gnet.tn

---

*RÉSUMÉ. Dans cet article, nous nous intéressons à l'évaluation de requêtes cliniques exprimées avec les facettes PICO (Population/Problem (P), Intervention (I), Comparaison (C) et Outcome (O)). Nous proposons l'application d'un opérateur d'agrégation prioritaire des scores qui permet : (1) d'agréger les scores de pertinence partiels issus de l'évaluation de représentations sémantiques associées aux sous-requêtes facettes et (2) contextualiser le score d'importance des facettes au document et requête en cours d'évaluation. Les expérimentations menées sur la collection standard CLIREC, comprenant 423 requêtes cliniques et plus de 1.2 millions de documents PubMed, mettent en évidence l'efficacité de notre approche comparativement aux autres modèles de l'état de l'art.*

*ABSTRACT. In this paper, we address the issue of answering PICO (Patient/Problem, Intervention, Comparison, Outcome) clinical queries. The contributions of this work include (1) a new document ranking model based on a prioritized aggregation operator that computes the global relevance score based on the relevance estimation of the semantic facet sub-queries and (2) leverages the importance of the facets according to the document and query under evaluation. The effectiveness of our clinical search approach is empirically evaluated using a clinical retrieval collection including 423 queries and more than 1.2 million of medical abstracts from PubMed. The experimental results show that our approach for PICO query answering significantly overpasses state-of-the-art document ranking models.*

*MOTS-CLÉS : RI Médicale, Requêtes Cliniques, Agrégation de scores, Représentation Sémantique*

*KEYWORDS: Medical IR, Clinical Queries, Semantic Query representation, Aggregation Scoring*

---

1. Population/Problem (P), Intervention (I), Comparaison (C) et Outcome (O)

## 1. Introduction

De nombreuses études ont clairement montré que la recherche d'information médicale est largement pratiquée aussi bien par des experts que par des novices (Fox et Duggan, 2013 ; Zhang, 2013). Plus spécifiquement, la recherche d'information médicale pratiquée par les experts, est un type de recherche verticale basé sur l'usage de ressources, comme les dossiers médicaux de patients ou la littérature scientifique médicale (eg. MEDLINE<sup>1</sup>, Cochrane<sup>2</sup>), en vue de répondre à différents objectifs comme l'aide au diagnostic, l'aide à la prescription médicale ou encore la recherche clinique. Ce type de besoins en information induit en grande partie l'évaluation de requêtes cliniques ayant pour objectif de sélectionner, à partir de la littérature scientifique du domaine, des procédés cliniques et/ou des preuves d'études systématiques (Yang *et al.*, 2011). Plus globalement, les tâches de recherche d'information cliniques expertes, sont généralement conduites par les cliniciens dans le cadre de la médecine basée sur les faits connue sous l'acronyme EBM (*Evidence-Based Medicine*) (Sackett *et al.*, 1996).

Cette dernière consiste à utiliser de manière rigoureuse, explicite et judicieuse, les preuves scientifiques les plus récentes et plus pertinentes lors de la prise de décision concernant les soins à prodiguer à chaque patient. Sa pratique implique que l'on conjugue l'expertise clinique individuelle avec les meilleures preuves cliniques externes obtenues actuellement par la recherche systématique (Sackett *et al.*, 1996). La recherche de ces meilleures preuves d'études cliniques, à partir de la littérature scientifique, est à juste titre, l'objet de notre travail présenté dans cet article. Cela suppose d'abord la formulation d'un besoin en informations cliniques. Selon l'approche EBM, un moyen qui a été proposé aux experts en vue de clarifier leur besoin est de structurer leur requête selon la forme PICO, à savoir : Patient/Problem (P), Intervention (I), Comparison (C) et Outcome (O), appelés les éléments ou facettes PICO (Schardt *et al.*, 2007). A titre d'exemple, la formulation PICO de la question clinique *Q* : "*In people with recurrent aggression having any antiepileptic drug in any dosage, what is length of time of placebo for observer reported aggression ?*" est comme suit : *P* ⇒ "*people with recurrent aggression*", *I* ⇒ "*any antiepileptic drug in any dosage*", *C* ⇒ "*length of time of placebo*", *O* ⇒ "*reported aggression*".

En plus du verrou largement reconnu lié à l'ambiguïté des expressions médicales et acronymes (Trieschnigg, 2010), l'évaluation de ce type de requêtes pose deux difficultés supplémentaires et non triviales à résoudre : (1) considérer la structure de la requête en facettes dans le processus d'appariement avec des documents qui ne sont pas ainsi structurés et (2) considérer leur importance relative dans l'estimation du score de pertinence. À notre connaissance, c'est un champ d'investigation peu exploré ; on recense en effet peu de travaux qui ont abordé spécifiquement le problème d'évaluation des questions cliniques PICO (Boudin *et al.*, 2010c ; Boudin *et al.*, 2010b ; Demner-Fushman et Lin, 2007). Dans l'ensemble des travaux précédents, une étape prélimi-

1. <https://www.nlm.nih.gov/bsd/pmresources.html> accessible à l'aide PubMed

2. <https://www.ncbi.nlm.nih.gov/pubmed>, Cochrane <http://www.cochranelibrary.com/>

naire à la recherche, est la détection des facettes PICO dans les documents. Au niveau de l'appariement requête-document, les auteurs dans (Demner-Fushman et Lin, 2007), se sont basés sur une approche d'appariement sémantique entre les types sémantiques extraits d'UMLS de chacune des facettes détectées dans les requêtes et les documents. Cependant, cette méthode ne prend pas en compte l'importance des facettes PICO dans le texte, lors du calcul des scores de pertinence. Boudin et al. (Boudin *et al.*, 2010c) ont proposé un modèle d'appariement qui considère l'importance de chaque facette PICO pour calculer le score de pertinence des documents. Toutefois, les scores d'importance sont calculés d'une manière statique, sur la base de la distribution des mots appartenant à chaque facette sur l'ensemble de la collection de documents. Dans cet article, nous proposons un modèle d'appariement requête PICO-document, qui à la différence des précédents travaux : (1) ne requiert pas l'identification préalable des facettes *P*, *I*, *C* et *O* dans les documents, (2) utilise un opérateur d'agrégation prioritaire (Pereira *et al.*, 2010) dans le calcul des scores d'appariement requête-document en personnalisant les poids de chaque facette selon la requête et le document en cours d'évaluation. Comme dans (Demner-Fushman et Lin, 2007), nous privilégions une représentation sémantique des requêtes et documents, qui est cependant basée sur la génération de graphes sémantiques par facette, développé dans notre précédente contribution (Znaidi *et al.*, 2015).

L'article est organisé comme suit. La section 2 présente une synthèse des travaux portant sur l'évaluation de requêtes PICO. La section 3 donne un large aperçu du processus d'évaluation de requête, annonce nos hypothèses de recherche puis détaille le modèle d'agrégation de scores de pertinence pour l'évaluation de requêtes PICO. Dans la section 4, nous présentons le cadre expérimental puis détaillons et analysons les résultats. Enfin, la section 5 conclut l'article et annonce les pistes de travaux futurs.

## 2. Evaluation de requêtes cliniques PICO

De nombreux précédents travaux ont montré que les requêtes médicales sont particulièrement complexes (Natarajan *et al.*, 2010 ; Suominen *et al.*, 2013). L'un des facteurs de complexité les plus abordés est incontestablement celui du fossé sémantique entre besoins en information (experts ou novices) et documents. Les solutions ont porté essentiellement sur l'utilisation de ressources sémantiques comme MeSH ou UMLS pour l'enrichissement des requêtes et/ou documents (Stokes *et al.*, 2009 ; Dinh et Tamine, 2012) ou alors dans le modèle d'appariement requête-document (Trieschnigg, 2010 ; Mao *et al.*, 2015). Parmi les requêtes médicales, les requêtes cliniques sont considérées, sous l'angle du système de recherche d'information, comme particulièrement difficiles car elles sont de nature exploratoire et l'évaluation de la pertinence des réponses candidates requiert des facettes d'informations contextuels difficiles à identifier (Francke *et al.*, 2008 ; Natarajan *et al.*, 2010). Sous l'angle de l'expert et du système de recherche d'information, une façon de préciser le besoin en information induit par une requête clinique est de la structurer en facettes PICO. Les travaux sur l'évaluation automatique des requêtes PICO sont peu abondants et se scindent

en deux volets. Dans la première catégorie de travaux (Boudin *et al.*, 2010a ; Zhao *et al.*, 2010), le problème principal adressé par les auteurs est la détection des facettes PICO, comme une étape en amont à la sélection de documents pertinents. La plupart des approches sont basées sur des techniques d'apprentissage supervisé afin d'identifier les éléments PICO à partir du texte. Par exemple, dans (Boudin *et al.*, 2010a), le processus d'identification des éléments PICO à partir du document a été conduit selon deux étapes : une première étape pour la segmentation du texte des documents en plusieurs phrases, puis dans une seconde étape, chaque phrase est transformée en un vecteur de propriétés utilisant les caractéristiques statistiques et linguistiques pour désigner les facettes  $P$ ,  $IC$  et  $O$ . Les expérimentations sur un ensemble de 260000 résumés de *PubMed* ont montré que la combinaison linéaire de plusieurs classifieurs est l'approche la plus efficace pour la détection des éléments PICO.

Plus proche de notre contribution, la deuxième catégorie de travaux (Boudin *et al.*, 2010c ; Boudin *et al.*, 2010b ; Demner-Fushman et Lin, 2007) concerne la définition de modèles de recherche d'information qui exploitent les facettes PICO pour calculer les scores de pertinence des documents. Pour atteindre cet objectif, Boudin et al. (Boudin *et al.*, 2010c ; Boudin *et al.*, 2010b) ont proposé une extension de la version basique du modèle de langue (Song et Croft, 1999). Les auteurs ont modifié le modèle de pondération basé sur les termes des documents en tenant compte de la distribution des éléments PICO dans les différents passages de documents ainsi que la distribution des termes dans les différentes parties PICO. L'évaluation expérimentale conduite sur une collection de 1.5 millions de documents et 423 requêtes a montré que le modèle proposé a permis une amélioration de 28% de la  $MAP^3$  sur l'ensemble des modèles de référence. Demner-Fushman et Lin (Demner-Fushman et Lin, 2007) ont également proposé un modèle unifié pour détecter et utiliser les éléments PICO dans une fonction de calcul de pertinence des documents  $S_{EBM}$ . Cette dernière est basée sur une combinaison linéaire des scores de pertinence partiels des documents, considérant trois éléments de l'EBM, à savoir, la structure PICO ( $S_{PICO}$ ), la crédibilité de la preuve médicale ( $S_{SoE}$ ) et le type de la tâche ( $S_{task}$ ). Par exemple, le score  $S_{PICO}$  est basé sur une combinaison linéaire des scores des facettes  $P$ ,  $I$ ,  $C$  et  $O$  en prenant en compte l'appariement des mots entre le document et la facette de la question. Les expérimentations sur 24 questions cliniques ont montré que cette approche dépasse, en termes de performance, la recherche classique dans *PubMed*.

### 3. Aperçu général de l'approche

#### 3.1. Hypothèses de recherche

Soit une requête clinique  $Q$  avec les annotations PICO associées donnant lieu aux sous-requêtes  $Q_P$ ,  $Q_{IC}$  et  $Q_O$ , manuellement ou automatiquement identifiées. Comme dans de précédents travaux (Boudin *et al.*, 2010c), nous considérons les fa-

---

3. Mean Average Precision

cettes  $I$  et  $C$  de façon regroupée comme elles sont associées au même type sémantique. Notre objectif est de sélectionner les documents  $d$  qui sont pertinents au besoin en information expert véhiculé derrière la requête  $Q$  et ce, en accord avec les facettes  $P$ ,  $I/C$  et  $O$ . On s'appuie pour cela sur une représentation sémantique de la requête, issue d'un processus de génération de graphes sémantiques exploitant la ressource MeSH (Znaidi *et al.*, 2015). Nous proposons d'appliquer un opérateur d'agrégation prioritaire (Da Costa Pereira *et al.*, 2009) afin de calculer le score global de pertinence du document  $d$  par combinaison de ses scores partiels d'appariement avec les sous-requêtes facettes  $Q_P$ ,  $Q_{IC}$  et  $Q_O$ . Pour cela, nous nous appuyons sur les deux hypothèses suivantes :

- **H1.** Un document est d'autant plus pertinent qu'il s'apparie avec plus de facettes de la requête (Boudin *et al.*, 2010c ; Demner-Fushman et Lin, 2007).
- **H2.** En phase d'évaluation de la pertinence des résultats, l'expert médical n'accorde pas la même importance à l'adéquation des facettes. La facette  $I/C$  est plus importante que la facette  $P$  qui, à son tour, est plus importante que la facette  $O$  (Weifield et Finkelstein, 2005 ; Boudin *et al.*, 2010c).

### 3.2. Processus de traitement de requêtes PICO

Comme le montre la Figure 1, notre approche est basée sur quatre (4) étapes principales : (1) un appariement préliminaire document-requête basé sur les mots, qui retourne une liste initiale de documents pertinents candidats, (2) représentation sémantique des requêtes, qui retourne les graphes sémantiques conceptuels associés à chaque facette de la requête, (3) un appariement requête-document qui permet de sélectionner les concepts les mieux pondérés de chaque facette du graphe, et (4) calcul de pertinence des documents basé sur un opérateur d'agrégation prioritaire (Da Costa Pereira *et al.*, 2009). Une description détaillée des étapes (2) et (3) est donnée dans (Znaidi *et al.*, 2015). Le principe général de l'algorithme sous-jacent est le suivant :

- *Etape 2* : A partir des sous-requêtes facettes  $Q_P$ ,  $Q_{IC}$  et  $Q_O$ , un algorithme génère des graphes sémantiques de requêtes  $Q_P^c$ ,  $Q_{IC}^c$ ,  $Q_O^c$ , en effectuant une extraction de concepts MeSH (Znaidi *et al.*, 2015) pour chacune des facettes en remontant de proche en proche des concepts extraits à partir des mots de la requête jusqu'à atteindre le plus proche concept commun de la hiérarchie de MeSH.
- *Etape 3* : Pour chaque arbre sous-requête associé à une facette  $Q_P^c$ ,  $Q_{IC}^c$ ,  $Q_O^c$  et chaque document  $d$  résultat de l'étape préliminaire (1), identifier les  $N_c$  meilleurs concepts associés en appliquant un algorithme de propagation de scores d'appariement concept-document par accumulation des scores des concepts associés au document  $d$  jusqu'aux concepts feuilles de MeSH.

Une illustration de la représentation sémantique issues des étapes (2) et (3), appliquée à la requête  $Q$  présentée en exemple "*In people with recurrent aggression having any antiepileptic drug in any dosage, what is length of time of placebo for observer reported aggression ?*" est donnée, respectivement, dans les Figure 2 et Figure 3.

Dans le présent article, notre principale contribution porte sur le calcul du score de pertinence des documents en réponse à une requête clinique, correspondant à l'étape (4).

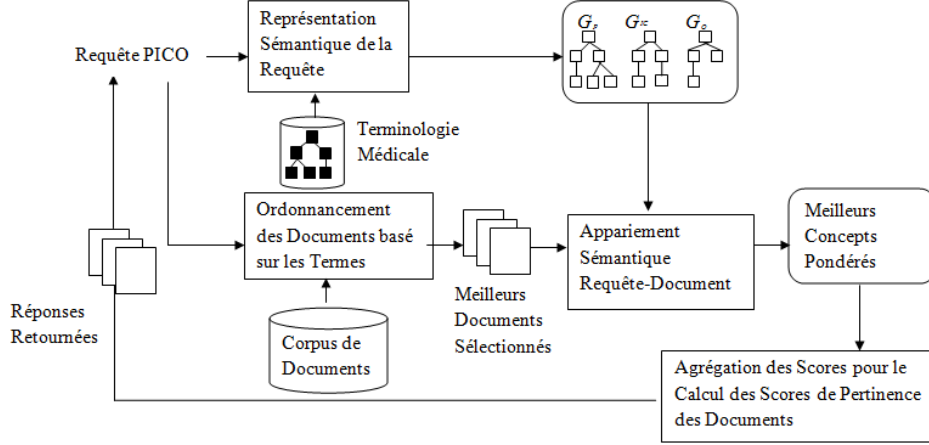


Figure 1 – Architecture du modèle pour le traitement des questions PICO.

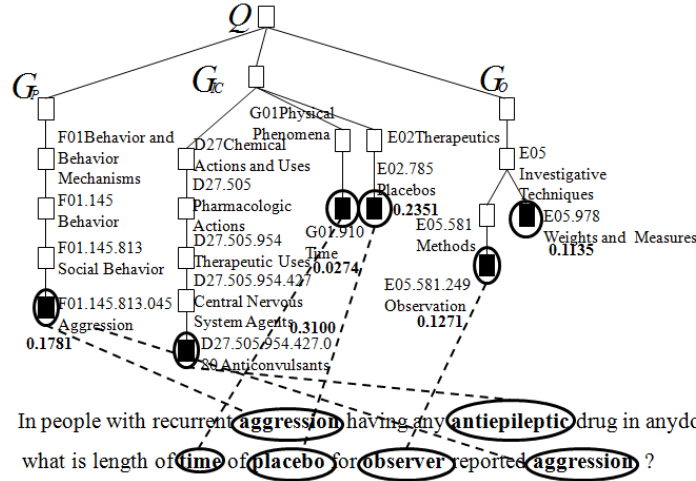


Figure 2 – Exemple de graphe sémantique d'une requête PICO

### 3.3. Modèle d'agrégation pour l'estimation de la pertinence en réponse à des requêtes PICO

Dans l'étape (4), représentée dans la Figure 1., nous disposons d'une représentation sémantique de chacune des requêtes facettes  $Q_P$ ,  $Q_{IC}$ ,  $Q_O$  à l'aide des  $N_c$  concepts pertinents issus de l'étape (2) et (3) (résumées dans la section précédente). Nous considérons que chaque facette  $P$ ,  $I/C$  et  $O$  constitue un critère de pertinence et proposons une fonction d'agrégation  $F$  qui calcule, pour chaque document  $d$ , un score global de pertinence comme suit :

$$RSV_{PICO}(Q, d) = F(RSV_P(Q_P, d), RSV_{IC}(Q_{IC}, d), RSV_O(Q_O, d)) \quad [1]$$



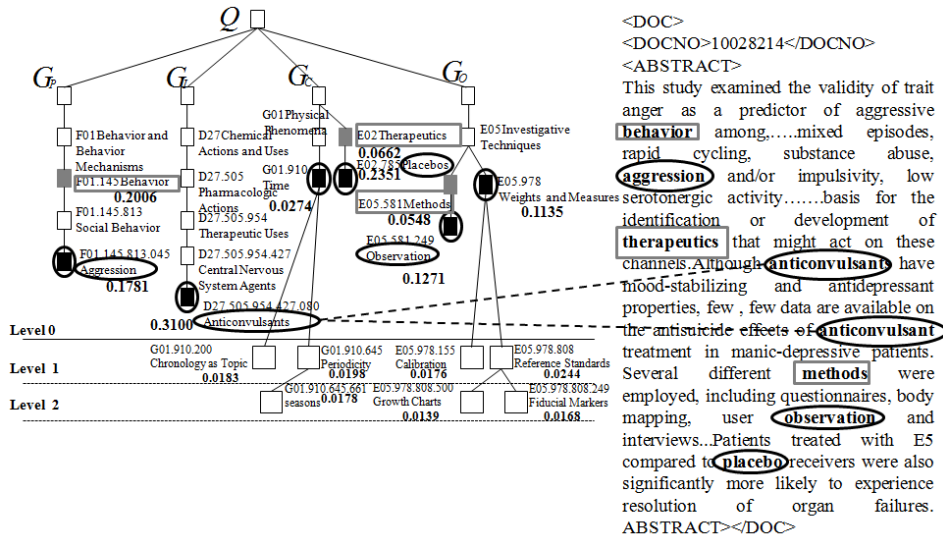


Figure 3 – Illustration de l'appariement requête-document

où  $F$  représente la fonction d'agrégation et  $P$ ,  $IC$  et  $O$  sont les critères de pertinence PICO. Nous calculons pour chacun de ces critères un score d'importance qui reflète son poids ou contribution dans l'estimation de la pertinence du document. Nous soutenons l'idée que les scores d'importance obéissent globalement à la hiérarchie posée par l'hypothèse H2 (Section 3.1) et confirmée par les travaux de l'état de l'art (Weifield et Finkelstein, 2005 ; Boudin *et al.*, 2010c) mais ne sont cependant pas fixes pour toutes les requêtes et documents. En effet, nous jugeons opportun de contextualiser les scores d'importance de ces facettes aux requêtes et documents en cours d'évaluation, à plus forte raison, si elles ne sont pas explicitement identifiées dans les documents. En clair, plus le document est supposé pertinent pour la facette la plus importante, moins il convient de considérer le score de son appariement avec la facette la moins importante. Pour répondre à cette intuition, nous proposons l'utilisation d'un opérateur prioritaire d'agrégation de scores (Da Costa Pereira *et al.*, 2009), qui permet de calculer un score global en tenant compte des scores partiels obtenus pour chaque critère ainsi que d'un ordonnancement dans l'importance de ces critères. Le poids de chaque critère est calculé en fonction des poids ainsi que le degré de satisfaction du critère le plus important. L'intuition sous-jacente est la suivante : plus le degré de satisfaction du critère le plus important est élevé, moins le degré de satisfaction du critère le moins important influencera le score global de pertinence.

En accord avec l' hypothèse H1 (Section 3.1), on calcule le score PICO comme suit :

$$RSV^4_{PICO}(Q, d) = \lambda_P * RSV(Q_P, d) + \lambda_{IC} * RSV(Q_{IC}, d) + \lambda_O * RSV(Q_O, d) \quad [2]$$

#### 4. Relevance Status Value

où  $x \in \{P, IC, O\}$  et  $RSV_x(Q, d)$  est calculé comme suit :

$$RSV_x(Q, d) = \frac{\sum_{C \in Q_x^C} SIM(C, d)}{\sum_{C \in \cup_x Q_x^C} SIM(C, d)} \quad [3]$$

où :

–  $Q_x^C$  : est l'ensemble  $N_c$  de concepts pondérés, associés à chaque facette de la requête  $Q_x$ , résultat de l'étape 3.

–  $SIM(C, d)$  : est le degré de similarité entre les vecteurs  $TF/IDF$  du document  $d$  et concept  $C$  représenté par ses entrées préférées dans la terminologie MeSH.

Les poids d'importance des éléments PICO sont calculés selon le principe de l'opérateur d'agrégation des scores (Da Costa Pereira *et al.*, 2009), comme suit :

1) Hiérarchisation des poids d'importance des facettes en tenant compte de l'hypothèse H2 :

$$\lambda_P, \lambda_{IC}, \lambda_O \in [0..1], \quad \text{where } \lambda_{IC} > \lambda_P > \lambda_O \quad \text{and} \quad \lambda_{IC} = 1 \quad [4]$$

2) Calcul de scores contextualisés de l'importance des facettes :

$$\begin{aligned} \lambda_P &= \lambda_{IC} * RSV(Q_{IC}^C, d) \\ \lambda_O &= \lambda_P * RSV(Q_P^C, d) \end{aligned} \quad [5]$$

Pour prendre en compte la pertinence du document en se basant sur les mots et sur les concepts comme recommandé dans (Stokes *et al.*, 2009), le score de pertinence du document  $d$  par rapport à la requête  $Q$  est la combinaison des scores de pertinence basés sur les concepts ( $Score_{PICO}(Q, d)$ ) et le score de pertinence basé sur les mots ( $Score_w(Q, d)$ ). Le score global de pertinence est calculé comme suit :

$$RSV(Q, d) = \alpha * RSV_{PICO}(Q, d) + (1 - \alpha) * RSV_w(Q, d) \quad [6]$$

où  $\alpha \in [0..1]$  est un paramètre fixé de façon empirique.

#### 4. Evaluation expérimentale

Les objectifs de l'évaluation expérimentale sont : (1) mesurer l'efficacité du modèle d'évaluation des requêtes PICO en analysant l'effet de chacun des éléments contributifs (représentation conceptuelle, principe de combinaison des facettes, principe de pondération des facettes) ; (2) analyser la robustesse du modèle en identifiant les raisons possibles d'échec vs. succès des requêtes.

|                                        |                          |
|----------------------------------------|--------------------------|
| Nombre de documents                    | 1.212.040 résumés PubMed |
| Longueur moyenne de document           | 246 termes               |
| Nombre de requêtes                     | 423                      |
| Nombre moyen de termes de la requête   | 4.3 termes               |
| Longueur moyenne de la requêtes (PICO) | 18.7 termes              |
| Nombre moyen de doc pertinents         | 19                       |

Tableau 1 – Statistiques de la collection de test CLIREC

#### 4.1. Cadre d'évaluation

##### 4.1.1. Collection de données

Nous utilisons la collection de données CLIREC (*CLinical Information Retrieval Evaluation Collection*), construite dans le but d'évaluer la recherche d'information clinique (Boudin *et al.*, 2010c). Pour atteindre cet objectif, les auteurs ont construit la collection de test d'une manière semi-automatique à partir d'un ensemble de revues systématiques de la ressource *Cochrane*. Plus spécifiquement, les auteurs ont demandé à un groupe d'experts de générer les requêtes qui correspondent aux questions cliniques fournies par un sous-ensemble du répertoire *Cochrane*. Chaque question est manuellement annotée avec les parties *P* (Patient/Problème), *I* (Intervention), *C* (Comparaison) et *O* (Outcome, résultats). Pour constituer la vérité terrain, les auteurs ont tout d'abord extrait, pour chaque question clinique, les citations des documents associés à la section référence attachée à chaque revue ; cette section énonce en effet l'ensemble des études pertinentes qui traitent la question clinique considérée. Ensuite, une liste de documents pertinents (articles de journaux) associés à ces citations sont extraits de *PubMed* pour chaque question clinique.

Quelques caractéristiques et statistiques de la collection sont représentés dans le Tableau 1. Nous avons utilisé *MeSH*, qui est la terminologie la plus utilisée pour indexer les citations biomédicales (Stokes *et al.*, 2009). Chaque nœud de la terminologie représente un concept qui fait référence à une entrée préférée dans la terminologie.

##### 4.1.2. Protocole expérimental

Nous avons adopté un protocole de validation croisée à base de 10 essais (*10-fold-cross validation*) pour le paramétrage de la fonction d'appariement puis le test de son efficacité. Nous avons calculé l'efficacité moyenne de la performance de recherche sur l'ensemble des 10 sous-collections d'essais construits en amont, après avoir identifié les valeurs optimales qui maximisent la *MAP*. Pour la mesure de l'efficacité de la recherche, nous utilisons les mesures de la précision moyenne (MAP), et  $P@X$  ( $X = 5, 10$ ) avec l'outil TREC-eval<sup>5</sup>.

5. <http://trec.nist.gov/trec eval>

#### 4.1.3. Modèles de référence

Nous avons comparé notre modèle de recherche d'information PICO, notée *PSM*, aux modèles de référence suivants, considérés avec leurs paramètres optimaux (optimisation de la MAP) :

- Deux modèles de recherche d'information de l'état de l'art : (1) le modèle probabiliste Okapi (*BM25*) avec es valeurs recommandées :  $k_1 = 1.2$ ,  $k_3 = 7$  et  $b = 0.75$  (Robertson et Sparck Jones, 1988) et (2) le modèle de langue (*LM*) avec le lissage de Dirichlet avec le paramètre  $\mu = 1000$  comme recommandé dans (Song et Croft, 1999).

- Une approche de reformulation sémantique de requêtes notée *CQE*, où nous avons étendu les requêtes initiales PICO  $Q$  avec les entrées préférées des meilleurs concepts MeSH retournés par l'étape 3 du processus d'extraction des concepts pertinents (Section 3.2). Nous avons utilisé le nombre de concepts qui optimise la MAP, soit  $N_C = 4$  (Znaidi *et al.*, 2015).

- Un opérateur d'agrégation prioritaire, noté (*PSBM25*), appliqué aux ordonnancements issus de l'application du modèle *BM25* pour l'évaluation de chacune des sous-requêtes  $Q_P$ ,  $Q_{IC}$  et  $Q_O$ .

- Deux modèles de recherche d'information de l'état de l'art, spécifiques à l'évaluation de requêtes PICO : (1) un modèle d'agrégation de scores sans pondération, noté (*AGM*), proche du modèle présenté dans (Demner-Fushman et Lin, 2007), où le modèle *BM25* est utilisé pour calculer la similarité entre les facettes des requêtes et les documents et, (2) *Positional Language Model (PLM)* présenté dans (Boudin *et al.*, 2010c), basé sur une extension du modèle de langue (LM). Les résultats présentés sont ceux rapportés dans (Boudin *et al.*, 2010c) sur la même collection de test et qui incluent uniquement les mesures de la MAP et la  $P@5$ .

## 5. Résultats

### 5.1. Efficacité du modèle de pertinence basé sur l'agrégation des scores

Dans un premier temps, nous avons identifié la valeur optimale du paramètre  $\alpha$  utilisé dans l'équation 6 et ce en le variant dans l'intervalle  $[0..1]$ . Comme le montre la Figure 4, la valeur optimale du paramètre  $\alpha$  est de 0.7 permettant d'atteindre un score de précision MAP égale à 0.170. Ces résultats montrent que le score basé sur les éléments PICO contribuent de façon significative au calcul du score global optimal. Nous retenons cette valeur pour le reste des expérimentations. Pour rappel, nous procédons à la variation du paramètre  $\alpha$  dans le cadre de la validation croisée.

Le Tableau 2 présente les résultats de l'efficacité de recherche de notre modèle d'agrégation prioritaire sémantique, notée *PSM*, comparativement avec ceux des six modèles de référence cités ci-dessus, selon les mesures de précision (MAP et  $P@X$ ) et du nombre de documents pertinents sélectionnés ( $\#RR$ ). Les résultats montrent d'importantes améliorations significatives pour toutes les mesures d'évaluation et modèles, sauf pour le modèle *PLM* où l'amélioration est faible et non significative. Plus spécifiquement, pour la mesure de la MAP, les améliorations varient de +4.60% à +52.36%. De plus, le nombre de documents pertinents retournés ( $\#RR$ ) est plus im-

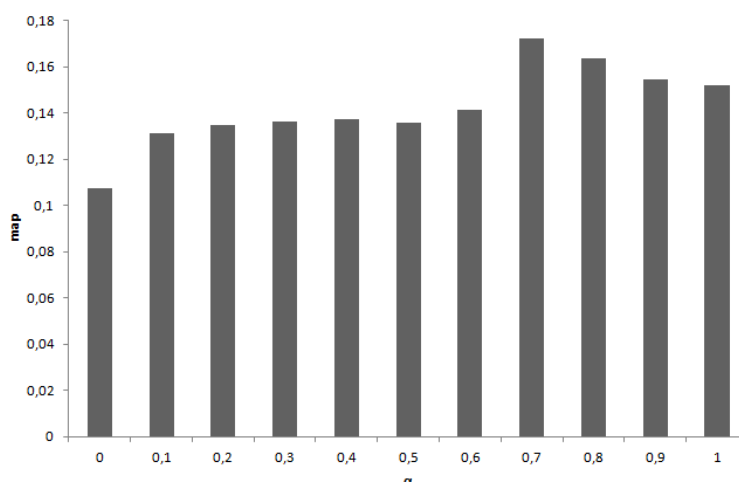


Figure 4 – Variation du paramètre de combinaison des scores basés mots et basés PICO (Formule 6)

portant en faveur du modèle *PSM* en comparaison avec l'ensemble des modèles de référence. D'une manière générale, ces résultats mettent en avant quatre (4) observations majeures :

- 1) Exploiter la sémantique cachée derrière les facettes des requêtes à travers les concepts, est efficace vu que le modèle *PSM* dépasse les modèles *BM25* et *LM* mais aussi le modèle *PSBM25* qui se base sur l'opérateur d'agrégation prioritaire.
- 2) L'importante amélioration de performance du modèle *PSM* par rapport au modèle *CQE* montre l'intérêt d'intégrer la structure de la requête en facettes *P*, *I/C* et *O* dans le calcul des scores de pertinence des documents.
- 3) Le modèle *PSM* donne de meilleurs résultats que le modèle *AGM* basé sur la représentation sémantique de la requête ainsi que l'agrégation additive des scores d'importance des facettes ; ces résultats montrent l'avantage d'assigner des scores partiels de pertinence, en adéquation avec l'importance des facettes PICO avec le document en cours d'évaluation.
- 4) Le modèle *PSM* est légèrement mais pas significativement plus pertinent que le modèle *PLM* (+4.60%). Ce point sera particulièrement exploré lors de l'analyse de robustesse du modèle *PSM* présentée dans ce qui suit.

## 5.2. Analyse de la robustesse du modèle d'agrégation de pertinence

Ici, notre objectif est d'analyser la robustesse du modèle *PSM*. Pour rappel, un modèle de recherche d'information robuste doit impacter positivement la plupart des requêtes (Wang *et al.*, 2012). Pour cela, nous menons tout d'abord une analyse globale d'amélioration/baisse des performances sur l'ensemble des requêtes en comparaison avec les modèles de recherche d'information PICO de l'état de l'art en l'occurrence les modèles *PLM* et *AGM*. Ensuite, nous nous focalisons sur l'étude de cas de requêtes typiques du succès vs. échec du modèle *PSM* comparativement au modèle de

| MODÈLE        | Précision    |              |              | % Change   | # RR        |
|---------------|--------------|--------------|--------------|------------|-------------|
|               | MAP          | P@5          | P@10         |            |             |
| <i>BM25</i>   | 0.112        | 0.1561       | 0.127        | +51.42%**  | 4574        |
| <i>LM</i>     | 0.111        | 0.156        | 0.130        | +52.36%*** | 4491        |
| <i>CQE</i>    | 0.14         | 0.163        | 0.142        | +47.71%**  | 4625        |
| <i>PSBM25</i> | 0.123        | 0.151        | 0.139        | +37.94%**  | 4904        |
| <i>AGM</i>    | 0.121        | 0.148        | 0.135        | +40.09%**  | 4835        |
| <i>PLM</i>    | 0.163        | 0.240        | —            | +4.60%     | 5770        |
| <b>PSM</b>    | <b>0.170</b> | <b>0.254</b> | <b>0.198</b> | —          | <b>5894</b> |

Tableau 2 – Evaluation comparative du modèle *PSM*. %Change indique le taux de changement du modèle *PSM* en termes de *MAP*. Les symboles \*, \*\* and \*\*\* indiquent le degré de significativité du test de student : \* :  $0.01 < t \leq 0.05$ ; \*\* :  $0.001 < t \leq 0.01$ ; \*\*\* :  $t \leq 0.001$ .

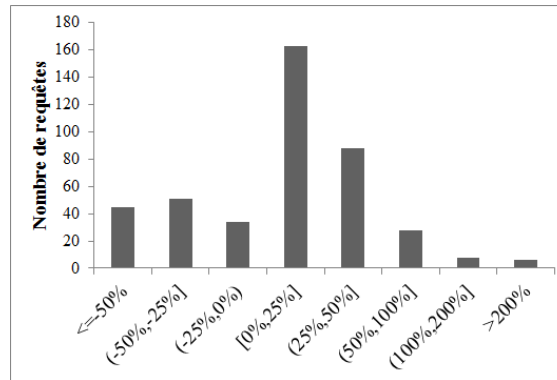


Figure 5 – Statistiques sur l'amélioration/dégradation en MAP comparé au modèle *PLM*

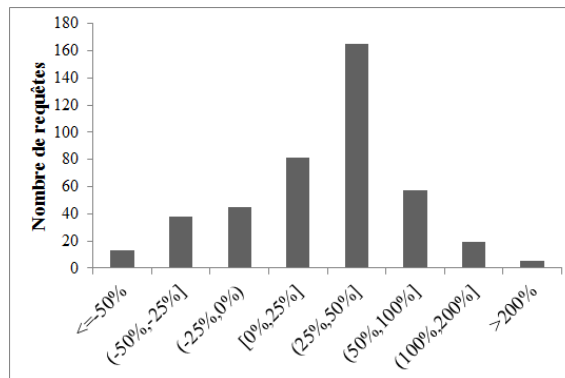


Figure 6 – Statistiques sur l'amélioration/dégradation en MAP comparé au modèle *AGM*

référence le plus performant pour lequel les améliorations observées sont non significatives (Section 5.1, Tableau 2).

### 5.2.1. Analyse globale

Les Figures 5 et 6 résument les statistiques concernant le nombre de requêtes dont la performance du modèle *PSM* a été plus élevée vs. dégradée en comparaison avec les modèles *PLM* et *AGM*. L'axe des abscisses représente le taux l'amélioration (+) vs. dégradation (-) de la performance en termes de la mesure *MAP*. L'axe des ordonnées représente le nombre de requêtes concernées par cette amélioration vs. dégradation. Les barres à gauche de [0 25%] représentent les requêtes dont les performances sont plus élevées en considérant les modèles de comparaison *AGM* et *PLM*. Les barres à droite (incluant [0 25%]) représentent les requêtes dont la performance est plus élevée pour le modèle *PSM*. On peut observer d'après ces figures que la proportion des requêtes avec une amélioration de la performance plus élevée en faveur du modèle *PSM*, est plus importante. A partir de la Figure 5, nous constatons plus particulièrement un grand nombre de requêtes avec une amélioration de performance dans l'intervalle [0%, 25%] suivi de (25%, 50%] par rapport au modèle *PLM*. Dans la Figure 6, nous observons un plus grand nombre de requêtes avec une amélioration de performance dans (25% 50%]) suivi de l'intervalle [0% 25%] par rapport au modèle *AGM*. Nous remarquons également que l'amélioration est plus importante par rapport au modèle *AGM* avec un plus grand nombre de requêtes dans [25% 50%]. Cela montre l'utilité et l'efficacité d'appliquer l'opérateur d'agrégation de scores avec des poids d'importances variables d'une requête à l'autre. La performance de quelques requêtes dépasse les 100%. Pour résumer, comparé aux modèles *PLM* et *AGM*, les résultats montrent que le modèle *PSM* est robuste.

### 5.2.2. Analyse au niveau requête

Dans le but de comprendre les raisons de l'amélioration vs. dégradation de la performance des requêtes du modèle *PSM* comparativement au modèle de référence *PSM*, nous avons analysé les deux meilleures requêtes *R+* et les deux plus faibles *R-* en termes de performance *MAP*. Le Tableau 3 présente les ensembles *R+* et *R-* en y mentionnant l'identifiant de la requête (*Id*), sa description (*Des*) ainsi que le taux de changement (augmentation vs. dégradation) par rapport au modèle *PLM* (*%Change*), la longueur en nombre de mots (*#T*), le nombre de concepts extraits (*#C*) et le score de clarté (*#Cla*) de la requête. Ce score traduit le degré d'appariement de la requête *Q* avec la collection. Il est calculé sur la base d'une mesure de divergence entre le modèle de langue de la requête et celui de la collection (Steve et Croft, 2002) :

$$Cla(Q) = \sum_{t \in V} P(t|Q) \log_2 \frac{P(t|Q)}{P_{coll}(t)} \quad [7]$$

où *V* est le vocabulaire de la collection, *t* est un mot,  $P_{coll}(t)$  est la fréquence relative du mot *t* et  $P(t|Q)$  est estimée  $P(t|Q) = \sum_{d \in R} P(w|D)P(D|Q)$  où *d* est un document, *R* est l'ensemble des documents indexés par au moins un mot de la requête *Q*.

| Q     | Id    | Desc                                                                                                                                                                                                                                       | %Change | #T | #C | #Cla  |
|-------|-------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|----|----|-------|
| $R^+$ | M6.2  | (P) In obese patients diabetes(\P)(IC) orlistat Placebo(\IC) (O)Weight loss(\O).                                                                                                                                                           | +78.08% | 7  | 4  | 0.080 |
|       | Q37.2 | (P) Adults 18 years or more migraine(\P)(IC)aspirin plus an antiemetic placebo (\IC) (O)pain free(\O).                                                                                                                                     | +74.60% | 8  | 5  | 0.071 |
| $R^-$ | C21.1 | (P) Adults 14 years and older GORD (\P)(IC)Medical management : proton pump inhibitors/histamine receptor antagonists Laparoscopic fundoplication surgery(\IC)(O) Health-related quality of life (\O).                                     | -45.07% | 18 | 7  | 0.062 |
|       | Q48.3 | (P) Adults 18 years or older rheumatoid arthritis (\P)(IC) methotrexate combined with other non-biologic disease modifying anti-rheumatic drugs (DMARDs) methotrexate alone(\IC)(O)ACR response of non-MTX DMARDS inadequate response(\O). | -83.21% | 26 | 6  | 0.020 |

Tableau 3 – Analyse comparative de cas types de requêtes types *PSM* vs. *PLM*

Les résultats présentés dans le Tableau 3 montrent que le modèle *PSM* présente une meilleure performance pour les requêtes relativement plus courtes (*M6.2* et *Q37.2*) ; inversement, le modèle *PLM* est plus performant pour les requêtes longues (*C21.1* et *Q48.3*). L'amélioration de la *MAP* atteint 78.08% et 74.60% respectivement pour les requêtes *M6.2* et *Q37.2*. Cela peut être expliqué par le fait qu'en comparaison avec le modèle *PLM*, le risque d'ambiguïté étant plus élevé pour les requêtes plus courtes, la représentation conceptuelle des facettes des requêtes proposée dans le modèle *PSM* permet de réduire l'effet négatif du défaut de correspondance entre les représentations document-requête sur les mots comme proposé dans le modèle *PLM*. À l'opposé, pour les requêtes longues, la dégradation de la *MAP* atteint respectivement -45.07% et -83.21% pour les requêtes *C21.1* et *Q48.3*. On remarque que les requêtes sont bien plus longues en mots (7, 8 vs. 18, 26) mais pas autant plus longues en concepts avec le même ordre de grandeur (4, 5 vs. 7, 6). Même si on constate que ces requêtes sont moins claires (0.062 et 0.020) que les requêtes plus longues (0.080 et 0.071), en défaveur du modèle *PLM*, ce dernier s'avère plus performant. Ceci peut être expliqué par le fait que plus la requête est longue, plus le risque d'appariement des mots avec les documents candidats est grand ; cependant la non prise en compte du contexte de la facette, comme cela est fait dans le modèle *PSM*, conduit au calcul d'un score de pertinence selon la formule 3 qui pourrait être erroné. A titre d'exemple, pour la requête *Q48.3*, des mots comme *older*, *rheumatoid* et *arthritis methotrexate* présents dans un document candidat peuvent s'apparier indifféremment avec les facettes *IC* et *P* dans le cas du modèle *PSM*. En revanche, dans



le modèle *PLM*, ce défaut d'appariement lié à la facette ne peut survenir puisque les documents sont préalablement annotés et l'appariement est effectué facette-facette.

## 6. Conclusion et perspectives

Dans cet article, nous avons proposé l'application d'un opérateur d'agrégation prioritaire pour l'évaluation de requêtes cliniques PICO. L'opérateur ne requiert pas une annotation préalable des facettes PICO dans les documents et permet d'adapter le score d'importance des facettes aux documents et requêtes en cours d'évaluation. Les expérimentations conduites sur la collection *CLIREC* ont montré que l'opérateur proposé est significativement plus performant que la majorité des modèles de référence basés sur l'appariement mot-mot, la reformulation sémantique des requêtes et des modèles d'agrégation classiques des ordonnancements issus de l'évaluation de chaque sous-requête associée à une facette. Bien que robuste, l'analyse des performances de recherche au niveau requête montre que l'opérateur *PSM* présente des limites. En effet, le modèle d'ordonnement des documents ne prend pas en compte le lien entre le contexte des mots dans les documents et leur contexte dans la requête, représenté par les facettes auxquelles les mots font référence. Nous planifions de vérifier le bien fondé de ces limites en menant une analyse statistique sur l'ensemble de la collection qui permettrait de déterminer les facteurs d'échec des requêtes. Suivra, une réflexion quant à l'intégration de ces facteurs comme éléments contextuels dans l'opérateur d'agrégation prioritaire des scores.

## 7. Bibliographie

- Boudin F., Nie J.-Y., Bartlett J., Grad R., Pluye P., Dawes M., « Combining classifiers for robust PICO element detection », *BMC Medical Informatics and Decision Making*, vol. 10, n° 1, p. 29, 2010a.
- Boudin F., Nie J. Y., Dawes M., « Clinical information retrieval using document and PICO structure », *NAACL HLT*, p. 822-830, 2010b.
- Boudin F., Nie J.-Y., Dawes M., « Positional language models for clinical information retrieval », *EMNLP*, p. 108-115, 2010c.
- Da Costa Pereira C., Dragoni M., Pasi G., « Multidimensional Relevance : A New Aggregation Criterion », in M. Boughanem, C. Berrut, J. Mothe, C. Soule-Dupuy (eds), *Advances in Information Retrieval*, vol. 5478 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, chapter 25, p. 264-275, 2009.
- Demner-Fushman D., Lin J., « Answering Clinical Questions with Knowledge-Based and Statistical Techniques », *Comput. Linguist.*, vol. 33, n° 1, p. 63-103, 2007.
- Dinh D., Tamine L., « Towards a context sensitive approach to searching information based on domain specific knowledge sources », *Web Semantics : Science, Services and Agents on the World Wide Web*, 2012.
- Fox K., Duggan M., « Health online 2013 », 2013.
- Francke A., Smit M., de Veer A., « Factors influencing the implementation of clinical guidelines for health care professionals : a systematic meta-review », *BMC Medical Information Decision Making*, vol. 8, n° 38, p. 1-11, 2008.

- Mao J., Lu K., Mu X., Li G., « Mining Document, Concept, and Term Associations for Effective Biomedical Retrieval : Introducing MeSH-enhanced Retrieval Models », *Inf. Retr.*, vol. 18, n° 5, p. 413-444, 2015.
- Natarajan K., Stein D., Jain S., Elhadad N., « An analysis of clinical queries in an electronic health record search utility », *International journal of medical information*, vol. 79, n° 7, p. 515-522, 2010.
- Pereira C., Dragoni D., Pasi G., « Multidimensionnal relevance : a new aggregation criterion », *ECIR 2010*, p. 264-275, 2010.
- Robertson S. E., Sparck Jones K., « Document Retrieval Systems », Taylor Graham Publishing, London, UK, UK, chapter Relevance Weighting of Search Terms, p. 143-160, 1988.
- Sackett D. L., Rosenberg W. M. C., Gray J. A. M., Haynes R. B., Richardson W. S., « Evidence based medicine : what it is and what it isn't », *BMJ*, vol. 312, n° 7023, p. 71-72, 1996.
- Schardt C., Adams M., Owens T., Keitz S., Fontelo P., « Utilization of the PICO framework to improve searching PubMed for clinical questions », *BMC Medical Informatics and Decision Making*, vol. 7, n° 1, p. 16+, 2007.
- Song F., Croft W. B., « A General Language Model for Information Retrieval », *CIKM '99*, p. 316-321, 1999.
- Steve C. R., Croft W., « Quantifying query ambiguity », *Proceedings of the second international conference on Human Language Technology Research*, HLT '02, p. 104-109, 2002.
- Stokes N., Cavedon Y., Zobel J., « Exploring criteria for succesful query expansion in the genomic domain », *Information retrieval*, vol. 12, p. 17-50, 2009.
- Suominen H., Salanter S., Velupillai S., Chapman W., Savova G., Elhadad N., Pradhan S., South B., Mowery D., Jones G., Leveling J., Kelly L., Goeuriot L., Martinez D., Zuccon G., *Information Access Evaluation. Multilinguality, Multimodality, and Visualization*, vol. 8138 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, p. 212-231, 2013.
- Trieschnigg D., Proof of concept : concept-based biomedical information retrieval, PhD thesis, University of Twente, 2010.
- Wang L., Bennett P. N., Collins-Thompson K., « Robust Ranking Models via Risk-sensitive Optimization », *SIGIR '12*, p. 761-770, 2012.
- Weifield J., Finkelstein K., « How to answer your clinical questions more efficiently », *Family practice management*, vol. 12, n° 7, p. 37, 2005.
- Yang L., Mei Q., K.Zheng, Hanauer D. A., « Query log analysis of an electronic health record search engine », *Proceedings of the annual symposium AMIA*, AMIA '11, p. 915-924, 2011.
- Zhang Y., « Searching for specific health-related information in MedlinePlus : behavioral patterns and user experience », *Journal of the American Society for Information Science and Technology (JASIST)*, 2013.
- Zhao J., yen Kan M., Procter P. M., Zubaidah S., Yip W. K., Li G. M., « Improving Search for Evidence-based Practice using Information Extraction », *BMC Medical Informatics and Decision Making*, 2010.
- Znaidi E., Tamine L., Latiri C., « Answering PICO Clinical Questions : a Semantic Graph-Based Approach (short paper) », *Conference on Artificial Intelligence in Medicine (AIME)*, 2015.



CORIA – CIFED 2016

Documents anciens  
CIFED

Président : *XX*

Répondre à des requêtes cliniques PICO

Papier absent

Papier absent

*Maroua Mehri, Pierre Héroux, Mickaël Coustaty, Petra Gomez-Krämer,  
Julien Lerouge, Rémy Mullot*

## Papier absent

Papier absent



*Maroua Mehri, Pierre Héroux, Mickaël Coustaty, Petra Gomez-Krämer,  
Julien Lerouge, Rémy Mullot*

## Papier absent



Papier absent

Papier absent

# Papier absent

Papier absent

# Papier absent





Papier absent

Papier absent

# Papier absent

Papier absent

# Papier absent





# CORIA – CIFED 2016

Microblogs

CORIA

Président : *Patrice Bellot*





---

# Synthèse de flux de messages en temps réel

**Abdelhamid Chellal — Mohand Boughanem — Bernard Dousset**

*IRIT, Université de Toulouse, CNRS, INPT, UPS, UT1, UT2J, France  
118 route de Narbonne 31062 Toulouse Cedex 9, France  
{abdelhamid.chellal, Mohand.Boughanem, Bernard.Dousset} @irit.fr*

---

*RÉSUMÉ. La supervision d'événements à travers les réseaux sociaux ont connu un engouement certain ces dernières années. Cependant, le nombre considérable de messages publiés rend difficile, voire impossible, pour une personne de suivre "ce qui se passe autour de l'événement". Le filtrage et la génération en temps réel d'une synthèse de messages importants portant sur l'événement permet de répondre à ce besoin. La génération de cette synthèse pose plusieurs problèmes qui rend cette tâche difficile. En effet, la synthèse doit être concise, non redondante et couvrant le maximum de sous événements, et ce, en sus de la pertinence des informations qu'elle contient. Dans cet article, nous proposons une nouvelle approche de sélection des messages pour la génération, en temps réel, de résumés de flux de messages courts. La décision de sélectionner un message est prise instantanément, un nouveau message est ajouté au résumé si ses scores afférents à l'informativité et à la non redondance sont supérieurs à un seuil dynamique. L'approche proposée a été évaluée sur la collection TREC 2014 TS et elle a été comparée avec trois approches de l'état de l'art. Le résumé généré est de meilleure qualité que celui généré par les approches de "base" avec un nombre réduit de phrases.*

*ABSTRACT. Monitoring stream of social media posts has attracted significant attention in the last view years. Real time summarization in microblog aims at providing new relevant and non redundant information about an event as soon as it occurs. In this paper, we propose a novel continuous summarization approach in which information updates are pushed in real time. The decision to select/ignore incoming information is based on its informativeness and redundancy scores. The on-hand post is added to a summary only if the aforementioned scores are above a parametric free threshold. Our strategy was evaluated on a TREC Temporal Summarization 2014 data-set and it was compared with a well known baselines. The results reveal that our method outperforms all baselines and runs of the aforementioned task. The generated summaries are shorter and have higher precision than summaries generated by the baselines.*

*MOTS-CLÉS : filtrage temps réel, résumé temporel, informativité, redondance,*

*KEYWORDS: real time filtering, temporal summarization, informativeness, redundancy.*

---

## 1. Introduction

Partager des informations portant sur un événement est devenu une pratique très courante voire un réflexe dans les réseaux sociaux tels que Twitter. Les utilisateurs publient des informations actualisées sur l'évolution d'un événement comme un match de football ou une catastrophe naturelle. En effet, au 30 juin 2015, Twitter comptait 316 millions d'utilisateurs actifs par mois avec 500 millions de tweets envoyés par jour<sup>1</sup>. La supervision et le suivi d'événements à travers les réseaux sociaux peuvent être bénéfiques à plus d'un titre pour un journaliste ou une entreprise. A priori, tout le monde peut tirer profit de la disponibilité de cette masse d'information fraîche, diversifiée et presque "gratuite". Cependant, au regard de l'évolution exponentielle du nombre de messages publiés, il est très difficile pour un humain de suivre le flux d'information et construire un aperçu général sur l'évolution de l'événement objet d'intérêt, et ce, en particulier, lorsque il s'agit d'un événement qui dure dans le temps.

De plus, dans le cas d'événement inattendu à l'instar d'un attentat terroriste ou d'une catastrophe naturelle, l'utilisateur est à la recherche d'information actualisée et d'une mise à jour au fil de temps. Dans un tel scénario, une synthèse concise, non redondante mise à jour en temps réel qui met en évidence les sous événements les plus importants en présentant le développement de l'événement, serait très utile pour répondre au besoin d'information de l'utilisateur (Zubiaga *et al.*, 2012).

La tâche de filtrage et de synthèse en temps réel d'événement est différente de la recherche d'information classique dans laquelle les documents (ressources) sont triés en fonction de leur score de pertinence vis-à-vis de la requête. En effet, outre la pertinence qui est dans les deux tâches, le filtrage et la synthèse en temps réel exigent de retourner les nouvelles informations relatives à un événement dès qu'une information pertinente apparaît dans le flux des messages (Aslam *et al.*, 2014), et le résultat est une synthèse regroupant les messages pertinents non redondant.

Cette thématique de recherche a rencontré un réel engouement dans la communauté de Recherche d'Information ces dernières années (Sharifi *et al.*, 2010a), (Sharifi *et al.*, 2010b), (Ren *et al.*, 2013), (Xu *et al.*, 2013), (Olariu, 2014), (Mackie *et al.*, 2014). Récemment, une nouvelle tâche dédiée à cette thématique (Real-Time Filtering) a été introduite dans le cadre des campagnes TREC<sup>2</sup>. Le défi majeur sous-jacent à cette thématique porte sur le fait qu'en plus de la nécessité de retourner des tweets pertinents vis-à-vis de l'événement objet d'intérêt, la synthèse générée doit satisfaire les contraintes suivantes : éviter la redondance, être diversifiée en couvrant tous les sous-événements, et réduire la latence entre le temps de publication de l'information et le temps de notification. En effet, être en mesure de détecter l'information pertinente dès sa publication est très important dans le cadre de cette tâche, et ce, car l'information en tant que telle a une durée de vie, une date d'expiration, au-delà de laquelle elle n'est plus pertinente pour l'utilisateur. Ensuite, la redondance dans le

1. <https://fr.wikipedia.org/wiki/Twitter>

2. <https://github.com/lintool/twitter-tools/wiki/TREC-2015-Track-Guidelines>

flux de tweets est omniprésente vu que la même information est reliée par plusieurs utilisateurs à des moments différents.

Ainsi, une synthèse optimale devrait couvrir tous les sous événements sans redondance où chaque information est ajoutée dans la synthèse dès son apparition dans le flux. Il a été démontré dans (Shou *et al.*, 2013) que la synthèse dans les microblogs est un problème NP-difficile.

Par ailleurs, la synthèse (résumé) d'événements dans les flux des messages courts est différente de la synthèse (résumé) de documents, et ce pour les raisons suivantes : (i) les messages sont horodatés et l'information véhiculée est actualisée, elle représente une évolution temporelle de l'événement ; (ii) le flux de messages contient beaucoup de redondance, la même information est reliée par plusieurs utilisateurs sur des intervalles de temps différents (iii) la taille d'un message est généralement petite (limitée à 140 caractères dans le cas des tweets) ce qui fait que le contenu est souvent mal orthographié avec une utilisation fréquente des abréviations.

Dans cet article, nous présentons une nouvelle approche de génération de synthèse incrémentale de messages saillants (importants) portant sur un événement spécifié publié dans un flux de messages. Un message peut être soit une phrase extraite d'un document horodaté ou un blog soit un tweet. Il est considéré important s'il contient des informations nouvelles par rapport à celles déjà rencontrées dans le flux et non redondantes vis-à-vis des messages déjà sélectionnés dans la synthèse. L'importance d'un message est évaluée à travers trois critères, sa pertinence vis à vis l'événement recherché, son informativité représentée par la quantité d'informations apportées et estimée à travers l'entropie de Shannon (Shannon, 1948) et sa non redondance par rapport à la synthèse estimée, à travers la mesure de divergence Kullback-Leibler (Kullback et Leibler, 1951). Notre contribution porte sur :

- 1) La décision de sélectionner/ignorer un message est prise en temps réel. Cela permet d'éviter la mise en mémoire "buffering" du flux de messages et par la même de réduire le délai entre le temps de publication et le temps de notification à l'utilisateur ;
- 2) La sélection des messages est basée sur trois critères : la pertinence, l'informativité et la non redondance ;
- 3) Le seuil sur lequel se base la sélection des messages n'est pas un paramètre préfixé mais estimé d'une manière automatique.

Afin d'évaluer l'approche proposée, des expérimentations ont été réalisées sur la version filtrée de la collection TREC Temporal Summarization 2014 (TREC-TS-2014F)<sup>3</sup>.

Cet article est organisé comme suit : Nous présentons dans la section 2 les travaux liés à la synthèse des microblogs et à la détection de la redondance. La section 3 décrit l'approche proposée. Dans la section 4, nous présentons et discutons les résultats de notre approche et nous les comparons aux résultats des trois approches de l'état de

3. <http://s3.amazonaws.com/aws-publicdatasets/trec/ts/index.html>

l'art et aux systèmes ayant participé à la tâche TREC TS 2014. Nous terminons par une conclusion dans la section 5.

## 2. Etat de l'art

Dans cette section, nous présentons un bref aperçu sur les travaux relatifs à la génération automatique du résumé dans les microblogs suivi des approches de détection de la redondance en recherche d'information.

### 2.1. Résumé automatique dans les microblogs

D'une manière générale, les techniques du résumé automatique des documents peuvent être regroupées en deux catégories : abstractives et extractives (sélectives) (Ren *et al.*, 2013). Dans la première catégorie, le résumé généré contient des phrases, construites par le système de résumé automatique, qui n'apparaissent pas forcément dans les documents originaux. Tandis que dans la seconde catégorie, le résumé est construit à partir des phrases importantes sélectionnées directement des documents originaux. Pour cela, une fonction d'évaluation de l'importance des phrases est définie et les Top-K phrases sont sélectionnées. Ces deux stratégies ont été adoptées dans la littérature pour la génération automatique de résumé dans les microblogs. En ce qui concerne l'approche proposée dans cet article, elle appartient à la seconde catégorie. Ce choix d'adopter une approche sélective est motivé par le fait que c'est la stratégie la mieux adaptée à un scénario temps réel dans lequel le résumé est construit d'une manière incrémentale.

**Les méthodes abstractives :** La majorité des approches abstractives proposées dans la littérature se basent sur les graphes où les nœuds correspondent aux termes des tweets et les arrêtes modélisent leur ordre d'adjacence. La première approche appartenant à cet axe, est l'algorithme Phrase Reinforcement (PR) proposé par (Sharifi *et al.*, 2010b) qui génère une seule phrase pour résumer un événement. Dans cette approche un graphe ayant comme racine les mots clés de la requête est construit à partir des termes qui apparaissent dans les tweets. Un poids est attribué à chaque nœud (terme) en fonction de sa distance par rapport à la racine et sa fréquence d'occurrence dans le flux. Le chemin ayant le poids le plus élevé constitue la phrase "résumé".

Dans (Olariu, 2013) les auteurs proposent l'approche *Multi Sentence Compression* (MSC) qui représente les tweets collectés par un graphe orienté où chaque mot est représenté par un nœud et les arrêtes modélisent la relation d'adjacence entre les termes (bigramme). A chaque arrête est associée un poids calculé en fonction des occurrences des termes dans le flux des tweets. Le résumé est donné par le chemin ayant la plus petite moyenne des poids des arrêtes et dont la taille (nombre de mots) est supérieure à un certain seuil. Ainsi dans cette approche, le résumé généré contient des mots bigrammes peu fréquents dans le flux. Dans (Olariu, 2014) une extension du modèle précédent a été introduite dans laquelle le graphe est construit à partir des occur-

rences des trigrammes au lieu des bigrammes. Chaque nœud représente un bigramme et une arrête est ajoutée entre deux nœuds si les deux bigrammes afférents ont un terme en commun. A titre d'exemple, le graphe correspondant au tweet  $(w_1, w_2, w_3)$  est  $(Debut, w_1) \rightarrow (w_1, w_2), (w_1, w_2) \rightarrow (w_2, w_3)$  et  $(w_2, w_3) \rightarrow (w_3, Fin)$ . Les poids des nœuds et des arrêtes correspondent aux occurrences des bigrammes et des trigrammes respectivement. Dans cette approche, les top-k nœuds ayant le poids le plus fort sont sélectionnés comme noyau. Ensuite pour chaque nœud sélectionné, une phrase "résumé" est construite à partir du chemin qui contient ce nœud et qui maximise un score d'importance estimé à partir des poids des nœuds et des arrêtes ainsi que la fréquences des termes. L'inconvénient majeur de cette approche est que l'utilisation de trigramme augmente de manière significative le nombre de nœuds dans le graphe.

**Les méthodes sélectives :** Concernant la synthèse extractive, les approches proposées peuvent être classées en deux catégories : Dans la première catégorie, les tweets sont modélisés dans des graphes où les nœuds représentent les tweets et les arrêtes correspondent à la similarité entre deux tweets. Pour chaque arrête un poids relatif au degré de similarité entre les tweets afférents est calculé. Dans (Liu *et al.*, 2012), d'autres caractéristiques liées au réseau social comme le nombre de retweets et la lisibilité du texte sont combinés avec la mesure de similarité pour évaluer le poids des arrêtes. Un score d'importance est calculé pour chaque nœud en fonction de sa similarité par rapport aux autres nœuds. Ainsi, le résumé est construit à partir des nœuds ayant le plus grand score d'importance.

Dans la seconde catégorie basée sur les caractéristiques des tweets, une variété de caractéristiques textuelles a été exploitée telles que le modèle de langue (O'Connor *et al.*, 2010), la fréquence des termes (Liu *et al.*, 2012) et TF-IDF (Chakrabarti et Punera, 2011). Dans certaines approches, la problématique de synthèse sélective de tweets a été formalisée sous forme d'un problème d'optimisation combiné avec des techniques de classification. Dans (Liu *et al.*, 2011) la synthèse automatique de tweets a été modélisée sous la forme du programme linéaire suivant : Pour un sujet donné, il faut sélectionner un nombre k de tweets qui maximise la somme des poids des concepts (les n-gramme les plus fréquents) sous deux contraintes : la première contrainte porte sur le nombre de tweets à sélectionner et la seconde concerne le nombre de termes total dans le résumé.

Dans Zubiaga *et al.* (2012) l'importance des tweets est évaluée en fonction des fréquences de ses termes et sa non redondance par rapport aux tweets déjà sélectionnés dans la synthèse. La non redondance est estimée par une divergence de Kullback-Leibler (Kullback et Leibler, 1951). Il s'agit de la première approche qui propose la génération en temps réel de résumé dans les microblogs. Cependant, elle est dédiée aux événements programmés comme les matchs de football.

Dans (Sharifi *et al.*, 2010c) les auteurs ont introduit l'approche hybride TF-IDF dans laquelle les tweets sont triés en fonction de leur score de pertinence estimé à travers la moyenne des poids de leurs termes. Le poids d'un terme est évalué par la fonction hybride TF-IDF. Les Top-k tweets sont sélectionnés en excluant ceux qui ont une similarité par rapport aux tweets déjà sélectionnés supérieure à un seuil prédéfini.

Dans ce contexte, la fonction cosinus a été utilisée pour estimer la similarité entre les tweets. Dans cette approche, les auteurs suggèrent de considérer la collection de tweets comme un seul document pour calculer la fréquence du terme TF. L'approche Sumbasic (Nenkova et Vanderwende, 2005) initialement proposée pour le résumé des documents s'est avérée aussi efficace pour les microblogs selon (Karkali *et al.*, 2014). L'intuition sous-jacente à cette approche est que les mots les plus fréquents dans les documents ont une probabilité plus élevée d'être sélectionnés dans le résumé que les mots les moins fréquents.

La première comparaison en termes de performance absolue des différentes approches de synthèse automatique des microblogs, a été réalisée par Inouye et Kalita, (2011). Cette étude a fait ressortir que les méthodes basées sur la fréquence des termes obtiennent des performances meilleures, et ce, au regard de la spécificité de tweets (non structurés et courts). Dans ce contexte, il a été rapporté que l'approche hybride TF-IDF est l'une des meilleures approches. Récemment, dans (Mackie *et al.*, 2014) les auteurs ont comparé onze approches de génération automatique de synthèse en utilisant quatre corpus de microblog. Les résultats révèlent que SumBasic (Nenkova et Vanderwende, 2005) et hybride TF-IDF (Sharifi *et al.*, 2010c) sont les plus efficaces. Partant de ce résultat et conformément aux recommandations des auteurs, nous avons adopté lesdites méthodes comme ligne de base dans l'évaluation expérimentale de notre approche.

Notre travail s'inscrit dans la ligne de recherche portant sur l'adoption de la stratégie sélective pour la construction de résumé en exploitant des statistiques textuelles. Notre approche diffère des approches citées ci-dessus par : (i) Le résumé est construit d'une manière incrémentale où la décision de sélectionner / ignorer un message est prise instantanément en temps réel et sans utilisation de connaissances externes. (ii) En outre de la pertinence, la sélection est basée sur deux critères (informativité et non redondance) combinées sous forme de contraintes conjonctives. (iii) Dans notre approche, le seuil de sélection est estimé d'une manière automatique. Noter aussi, que l'approche proposée est indépendante du type d'événement alors que l'approche proposée dans (Olariu, 2014) est applicable uniquement pour des événements planifiés.

## 2.2. Détection de la redondance

La détection de redondance repose sur des mesures de similarité / divergence comme la distance de Manhattan, la fonction cosinus ou la comparaison des distributions des probabilités. Elle est habituellement utilisée avec la nouveauté dans les travaux relatifs à la détection et le suivi des topics (Topic Detection et Tracking) (Markou et Singh, 2003). Deux classes d'approche peuvent être distinguées. Dans la première, dite document-document, le nouveau document est comparé à tous ceux de la collection. L'inconvénient majeur de ce type d'approche réside dans la complexité des calculs dans le cas de collections volumineuses. Dans la seconde classe, et pour diminuer cette complexité, un nombre réduit de documents résumant la collection est généré, la similarité du nouveau document est alors mesurée uniquement avec le sous

ensemble de documents (Karkali *et al.*, 2014). Notre approche se base sur cette seconde classe. Ce choix est motivé par la nécessité de réduire la complexité du calcul dans le traitement temps réel du flux.

### 3. Génération en temps réel de résumé basée sur l'informativité et la redondance

L'approche que nous proposons traite en temps réel le flux des messages puis décide de garder ou d'ignorer le message en cours en fonction de plusieurs critères. Les messages sont traités selon leur ordre chronologique. Le problème de génération en temps réel de synthèse peut être défini comme suit :

1)  $\forall T_i, T_j \in R, t_i < t_j$ . où  $R$  est le résumé et  $t_i, t_j$  sont les temps de publication des messages  $T_i$  et  $T_j$  respectivement. Autrement dit, les messages sont collectés et traités selon leur ordre chronologique ;

2)  $\forall T_i \in R, \Delta t = \tau_i - t_i \simeq 0$ , où  $\tau_i$  est le temps de notification (temps de prise de décision de sélectionner le message) ;

3)  $\forall T_i, T_j \in R, T_i \approx T_j$  ; c'est à dire que les deux messages  $T_i$  et  $T_j$  apportent des informations significativement différentes afin d'éviter la redondance dans la synthèse et d'assurer une couverture maximale de tous les sous-événements ;

4)  $R \prec R'$ , le résumé  $R$  est préféré par rapport au résumé  $R'$  si  $R$  couvre au moins le même nombre de sous-événements (aspects) que  $R'$  avec un nombre réduit de messages (résumé concis).

Ces critères sont satisfaits dans l'approche proposée comme suit : (i) La décision de sélectionner/ignorer un message est prise en temps réel dès la réception d'un message. (ii) Le processus de prise de décision se base sur trois dimensions : la pertinence, l'informativité et la redondance. La première dimension permet de réduire le bruit et de limiter le traitement aux messages potentiellement importants. La seconde dimension détecte les messages qui apportent une quantité d'information significative par rapport à ce qu'il a été déjà vu dans le flux. Tandis que la troisième dimension permet d'éviter de sélectionner des messages similaires à ceux déjà ajoutés dans le résumé ce qui permet de réduire la redondance.

Autrement dit, un nouveau message candidat est ajouté au résumé si et seulement si ses scores afférents à l'informativité et la non redondance sont supérieurs à un certain seuil. Cela nous ramène à la question suivante : comment définir les seuils de sélection ? En effet, les statistiques (fréquences des termes) varient dans le temps ce qui rend difficile la définition à priori des seuils de sélection. En outre, apprendre ces seuils à partir d'une collection d'entraînement pose le problème de dépendance vis à vis aux types d'événements de la collection d'entraînement. Pour remédier à cette problématique, nous proposons de fixer ces seuils d'une manière adaptative en fonction des valeurs précédemment observées.

### 3.1. Filtrage de messages

Pour atteindre les exigences citées ci-dessus, notre approche agit comme un filtre à trois niveaux. Le premier filtre porte sur la pertinence vis à vis la requête. Tous les messages qui ne contiennent pas au moins deux mots de la requête sont ignorés. Les tweets qui passent ce premier filtre sont considérés comme des tweets candidats à la sélection dans le résumé. Les deux filtres suivants portent respectivement sur l'informativité et la non redondance. Si on considère que  $R^{t_i}$  est l'ensemble des messages résumant un événement à l'instant  $t_i$ , un nouveau message candidat  $T_i$  sera ajouté à  $R^{t_i}$  si et seulement si :

$$\begin{cases} IS(T_i) \geq \max_{\forall T_j \in F^{t_i}, t_i < t_j} [IS(T_j)] \\ RS(T_i) \geq \max_{\forall T_j \in R^{t_i}, t_i < t_j} [RS(T_j)] \end{cases} \quad [1]$$

Où  $IS(T_i)$  et  $RS(T_i)$  sont les scores d'informativité et de non-redondance de message  $T_i$  respectivement.  $F^{t_i}$  et  $R^{t_i}$  sont le flux de messages et le résumé à l'instant  $t_i$  (temps de publication du message  $T_i$ ).

Ainsi  $T_i$  est ajouté au résumé  $R^{t_i}$  si ses scores d'informativité et de non-redondance sont respectivement supérieurs à ceux obtenus par tous les messages du flux (pour l'informativité) et ceux du résumé (pour la non redondance).

D'autres options auraient pu être considérées pour combiner les deux critères. En particulier, on aurait pu combiner linéairement les deux scores ; on aurait eu un seul seuil à contrôler. Cette alternative n'a pas été considérée car elle pourrait conduire à l'ajout dans le résumé des messages redondants ayant une informativité élevée où bien des messages non redondants mais peu informatifs. Nous proposons la combinaison de ces deux critères sous forme d'une condition conjonctive qui permet d'avoir une complémentarité entre eux.

### 3.2. Score d'informativité d'un message

En théorie de l'information, la quantité d'information apportée par un message est évaluée à travers l'entropie de Shannon (Shannon, 1948). Pour évaluer l'importance d'un message, nous proposons l'utilisation de cette mesure afin d'estimer son informativité par rapport au flux d'information. L'intuition sous-jacente à cette proposition est qu'un nouveau message est considéré comme pertinent (peut d'être ajouté au résumé) s'il apporte une information différente où complémentaire par rapport aux précédents messages dans le flux. L'informativité, plus précisément la quantité d'information, apportée par le message, mesurée par son entropie, est supérieure à la quantité d'infor-



mation apportée par ses prédécesseurs. Ainsi, le score d'informativité d'un message  $T$  publié à l'instant  $t$  est mesuré comme suit :

$$IS(T) = - \sum_{w_i \in T} P^t(w_i) \times \log_2(P^t(w_i)) \quad [2]$$

Où  $P^t(w_i)$  représente la probabilité d'occurrence de terme  $w_i$  à l'instant  $t$  (timestamps de la phrase  $T$ ) dans le flux. Cette probabilité est estimée comme suit :

$$P^t(w_i) = \frac{\#Messages\ ou\ w_i\ apparait\ à\ l'instant\ t}{\#Messages\ dans\ F^t} \quad [3]$$

La probabilité  $P^t(w_i)$  peut être vue comme une DF (Document Frequency) inverse où une fréquence documentaire relative. Il convient de noter que dans les microblogs, la fréquence d'occurrence d'un mot est généralement égale à 1 ou 0. Ainsi, pour un terme donné, sa fréquence d'occurrence (TF) dans le flux (la collection) est égale au nombre de messages où il apparaît (fréquence documentaire DF). De ce constat, il ressort que la notion de TF est peu pertinente.

Un message long contenant des termes importants sera préféré par rapport un message court. A titre d'exemple, supposons que  $R = \{t_1 = \{w_1, w_2\}\}$  et que nous avons trois nouveaux messages en entrée  $t_2 = \{w_1, w_3\}$ ,  $t_3 = \{w_4, w_5\}$  et  $t_4 = \{w_2, w_3, w_4, w_5\}$ . Le message  $t_4$  obtiendra les scores d'informativité et de non-redondance les plus élevés.

### 3.3. Score de non redondance

Pour évaluer la divergence entre deux messages, nous proposons de mesurer l'écart entre leur modèle de langue respective en utilisant la divergence de Kullback-Leibler (KL) (Kullback et Leibler, 1951) définie comme suit :

$$KL(T, T') = \sum_{w_i \in T \cap T'} \theta_T(w_i) \log \frac{\theta_T(w_i)}{\theta_{T'}(w_i)} \quad [4]$$

où  $\theta_T$  est le modèle de langue uni-gramme de message  $T$  et  $\theta_T(w_i)$  est la probabilité d'occurrence de terme  $w_i$  dans le message  $T$ .

Pour évaluer le score de non redondance d'un message candidat ayant passé les deux premiers filtres relatifs à la pertinence et l'informativité, nous proposons de mesurer la non redondance vis à vis aux messages déjà sélectionnés dans le résumé. Les scores de divergence entre le nouveau message et tous les messages du résumé actuel peuvent être agrégés dans un seul score global. Cependant, un message peut avoir une grande divergence moyenne tout en être très proche à un message particulier dans

le résumé. Pour cela, nous proposons de considérer le score de divergence minimum comme score de non redondance du nouveau message.

Plus précisément, ce que nous proposons est de considérer la divergence entre le nouveau message et le message le plus proche parmi les messages déjà sélectionnés. Le message le plus proche (le moins divergeant) est celui dont la divergence est la plus petite. Ainsi, le score de non redondance d'un message vis à vis du résumé est défini par le minimum de divergence KL obtenue par rapport à tous les messages du résumé actuel (à l'instant  $t_i$ )  $R^{t_i}$  tel que mentionné par la formule suivante :

$$RS(T_i) = \min_{\forall T_j \in R^{t_i}} KL(T_i, T_j) \quad [5]$$

Pour remédier au problème de probabilité nulle, nous pouvons lisser les probabilités en utilisant le lissage de Jelinek-Mercer (JM) ou de Dirichlet (D). Le résultat de nos expérimentations révèle que le lissage de Dirichlet convient mieux au contexte de synthèse temps réel des documents courts. Ce dernier est défini par la formule suivante :

$$\theta_T(w_i) = \frac{TF_t(w_i) + \mu P_F^t(w_i)}{|T| + \mu} \quad [6]$$

Où  $TF_t(w_i)$  est la fréquence d'occurrence du mot  $w_i$  dans le message  $T$  et  $\mu$  est le paramètre de lissage.  $P_F^t(w_i)$  est la probabilité d'occurrence de terme  $w_i$  dans le flux  $F$  à l'instant  $t$ . Elle est évaluée en utilisant l'estimation du maximum de vraisemblance (ML), et ce, comme suit :

$$P_F^{t_i}(w_i) = \frac{\#(w_i) \text{ dans } F^{t_i}}{|F^{t_i}|} \quad [7]$$

Dans nos expérimentation, le paramètre de lissage  $\mu$  a été fixé à 1000. Cette valeur a été estimée d'une manière expérimentale en variant  $\mu$  de 10 à 2000 par un pas de 100.

#### 4. Évaluation expérimentale

Afin de valider notre approche, une série d'expérimentations a été effectuée sur la collectionne TREC Temporal summarization 2014. Trois objectifs principaux sont identifiés :

- 1) Évaluer l'impact des critères relatifs à l'informativité et non redondance : combinés où utiliser séparément ;
- 2) Évaluer l'impact du seuil dans l'équation [1] ;
- 3) Comparer les performances de l'approche proposée avec ceux de l'état de l'art et ceux obtenus dans TREC TS 2014.

A cet effet, deux versions de l'approche proposée ont été testées. Dans la première, les deux filtres relatifs à l'informativité et la redondance sont appliqués dans le processus de fabrication du résumé. Tandis que dans la seconde version, un seul filtre est appliqué à la fois. De plus, deux manières différentes d'estimer le seuil de sélection (équation 1) ont été testées, la moyenne et le maximum des scores précédemment observés.

Les résultats obtenus par notre approche ont été confrontés à ceux de trois approches de l'état de l'art. La première est l'approche SumBasic (Nenkova et Vanderwende, 2005) recommandée pour être adoptée comme ligne de base par (Mackie *et al.*, 2014). La deuxième ligne de base est l'approche hybrideTF-IDF (Sharifi *et al.*, 2010c) qui a été rapporté comme étant une des meilleures approches de résumé dans les microblogs. Aussi, nous avons implémenté l'approche TF-IDF combinée avec la fonction cosinus pour l'estimation du score de non redondance. Il convient de signaler que ces lignes de bases ont été ajustées pour correspondre au traitement en temps réel et en utilisant le même seuil de sélection adopté par notre approche (équation N°1). Les équations ci-dessous décrivent les fonctions hybrideTF-IDF et Sumbasic pour un message donné  $T$ , respectivement :

$$HybridTF - IDF(T) = \frac{\sum_{w_i \in T} TF(w_i) \times IDF(w_i)}{\max[Minimum\ seuil, |T|]} \quad [8]$$

$$TF(w_i) = \frac{\#(w_i)\ dans\ F}{\#mot\ dans\ F}, IDF(w_i) = \log_2\left(\frac{\#Messages}{\#Messages\ ou\ w_i\ apparait}\right) \quad [9]$$

$$Sumbasic(T) = \sum_{w_i \in T} \frac{P(w_i)}{|T|}, P(w_i) = \frac{\#w_i\ dans\ F}{\#T\ ou\ w_i\ apparait} \quad [10]$$

#### 4.1. La collection des données

Les expérimentations ont été menées sur la collection des documents TREC *Temporelle Summarization* (TS) 2014 qui contient des documents horodatées en anglais issus de différentes sources du Web (presse, blog, réseaux sociaux, etc.) et collectés durant la période allant d'octobre 2011 à avril 2013. Le but de cette tâche est d'évaluer les approches de supervision des événements en se focalisant sur la détection le plutôt possible des nouvelles informations publiées dans un flux. Dans nos expérimentations, nous avons utilisé la version filtrée de la collection (TREC-TS-2014F) qui regroupe les documents susceptibles de contenir au moins une phrase pertinente. Pour l'année 2014, quinze (15) sujets ont été définis par les organisateurs de TREC. En plus, les fragments d'informations vitaux (pertinent et apporte une information nouvelle), qu'un résumé doit contenir, ont été fournis pour chaque sujet. Une phrase d'un document est jugée pertinente si elle est associée à au moins une information vitale. Les documents sont traités par ordre chronologique et le système de génération de résumé enregistre le temps de notification par rapport au flux de document au moment de la sélection d'une phrase dans le résumé. Les messages traités sont soit des tweets soit

**Tableau 1.** Comparaison entre les différentes configurations de l'approche proposée.

| Metrique                 | H             | nEG(S)        | C(S)          | Nb de phrases |
|--------------------------|---------------|---------------|---------------|---------------|
| <b>KL-Entropie-MAX</b>   | <b>0.1449</b> | <b>0.0793</b> | 0.3167        | 171           |
| <b>Entropie-Only-MAX</b> | 0.1191        | 0.0562        | 0.3925        | 333           |
| <b>KL-Entropie-AVG</b>   | 0.0878        | 0.0383        | 0.4151        | 758           |
| <b>KL-Only-MAX</b>       | 0.0846        | 0.0382        | 0.3427        | 477           |
| <b>Entropie-Only-AVG</b> | 0.0606        | 0.0239        | 0.528         | 1812          |
| <b>KL-Only-AVG</b>       | 0.0179        | 0.0064        | <b>0.5400</b> | 7986          |

des phrases extraites des documents horodatés. Dans l'approche décrite dans cet article, la décision de sélectionner/ignorer une phrase est prise immédiatement. Ainsi, le temps de notification correspond au temps de publication du document.

#### 4.2. Métriques d'évaluation

Dans la tâche TREC TS, les performances sont évaluées en fonction de la pertinence et la couverture lesquelles sont estimées à travers les métriques suivantes : (i) le gain escompté normalisé (nEG) qui permet d'évaluer la pertinence des phrases sélectionnées dans le résumé. (ii) la compréhension "Comprehensiveness" (C) qui estime la couverture du résumé. Ces métriques sont similaires aux mesures de précision et rappel respectivement (Aslam *et al.*, 2014), mais en pénalisant la redondance et la latence lors de la détection d'informations. (iii) la moyenne harmonique (H) entre nEG et C a été définie comme étant la métrique officielle. Les résultats sont classés en fonction de cette mesure. Tous les détails concernant ces métriques sont disponibles sur <http://www.trec-ts.org/downloads>.

#### 4.3. Résultats

Dans nos expérimentations, nous avons évalué plusieurs configurations de l'approche proposée. Dans la première configuration notée (KL-Entropie), les deux filtres (informativité et non-redondance) ont été utilisés dans le processus de sélection des phrases tandis que dans la seconde configuration un seul critère a été utilisé (KL) ou (Entropie). Les suffixes "MAX" et "AVG" représentent le seuil utilisé (le maximum ou la moyenne des scores d'informativité et de non redondance respectivement).

**Impact de la combinaison de deux dimensions :** Le tableau 1 liste les performances des différentes configurations. En ressort que la combinaison des deux composants augmente de manière significative le gain (nEG) tandis que la couverture (C) est légèrement réduite (par rapport à la couverture obtenue lorsque les critères de sélection sont utilisés séparément) ce qui permet d'avoir une moyenne harmonique

(H) relativement élevée. Cette réduction du couverture dans la version KL-Entropie est attendue vu que l'application des deux filtres réduit le nombre de phrases dans le résumé généré. Toutes les phrases que contient le résumé généré en utilisant les deux filtres informativité (entropie) et non redondance (KL) sont incluses dans les résumés générés lorsque un seul filtre est utilisé  $R(KL - Entropie) \subset R(Entropie - Only)$  et  $R(KL - Entropie) \subset R(KL - Only)$ . Ce résultat confirme notre intuition relative à la nécessité de combiner plusieurs dimensions pour la sélection de fragment d'information lors de la construction d'une synthèse en temps réel. De plus, il semble y avoir un compromis entre le gain (nEG) et la couverture (C) où il est plus difficile d'améliorer la couverture que le gain, et ce, au regard du fait que le résumé est construit d'une manière incrémentielle et il doit avoir une taille réduite.

En outre en comparant l'impact de chaque dimension seule, nous constatons que l'informativité joue un rôle plus important par rapport à la redondance. En effet, le filtrage selon le score d'informativité (entropie) permet d'avoir un résumé meilleur en terme de gain (nEG) et de couverture (ligne 2). Le score d'informativité représente la quantité d'information apportée par une phrase par rapport au flux. Ce score, tel que défini dans l'équation 2, augmente avec la fréquence des termes dans le flux. Ainsi une phrase est ajoutée au résumé si elle contient des termes importants (fréquents) ce qui permet de sélectionner des phrases pertinentes. Ce résultat signifie que le filtre relatif à l'informativité pré-sélectionne les informations candidates et le filtre de non-redondance approuve ou désapprouve ce choix.

**Impact de seuil :** Le seuil de sélection dans les deux filtres (équation 1) porte sur la valeur maximale des valeurs précédentes. Nous avons testé la moyenne comme seuil pour étudier l'impact de ce paramètre. Le résultat obtenu (Tableau 1) montre que l'utilisation de la moyenne comme seuil donne de faible performance en termes de gain et de moyenne harmonique (H) entre (nEG) et la couverture. Cependant, il donne une couverture meilleure. Cela peut être expliqué par le nombre relativement élevé de phrases sélectionnées dans le résumé en raison du fait que l'utilisation de la moyenne comme seuil est moins restrictive par rapport au maximum. Le nombre moyen de phrases dans le résumé généré par le KL-Entropie-AVG est de 758 alors que en moyenne le nombre de phrases dans le résumé généré par KL-Entropie-MAX est de 171. Nous observons que la couverture est proportionnelle à la taille du résumé (nombre de phrase).

**Comparaison des résultats par rapport à l'état de l'art :** Dans le tableau 2, nous comparons les performances de notre approche par rapport aux résultats obtenus par les 3 approches de l'état l'art et aux résultats officiels de TREC TS 2014 en terme de gain (precision) et de couverture (rappel). Cette comparaison révèle que l'approche KL-Entropie donne une meilleure moyenne harmonique (H) entre la couverture (C) et le gain (nEG). Elle surpasse l'approche qui a été classée la première dans TREC TS 2014 (2APSal) (Aslam *et al.*, 2014) de 24% et l'approche hybride TF-IDF de 19%. Le gain de l'approche proposée est constamment supérieur par rapport aux autres approches tandis que la couverture est plus petite. Ce résultat peut être expliqué par le fait que la combinaison des deux scores dans une contrainte conjonctive et l'utilisation

**Tableau 2.** Comparaison des performances de l'approche proposée par rapport aux lignes de bases et les résultats des participants à TREC TS 2014

| Metrique                         | H             | nEG(S)        | C(S)          |
|----------------------------------|---------------|---------------|---------------|
| <b>KL-Entropy-MAX</b>            | <b>0.1449</b> | <b>0.0793</b> | 0.3167        |
| <b>HybridTF-IDF</b>              | 0.1207        | 0.0593        | <b>0.3796</b> |
| <b>SumBasic</b>                  | 0.0904        | 0.0402        | 0.3351        |
| <b>TF-IDF</b>                    | 0.0987        | 0.0483        | 0.3672        |
| <b>TREC TS 2014 task results</b> |               |               |               |
| <b>2APSal</b>                    | <b>0.1162</b> | 0.0631        | 0.3220        |
| <b>Q1</b>                        | 0.1110        | 0.0657        | <b>0.4088</b> |
| <b>Q2</b>                        | 0.1091        | 0.0632        | 0.3979        |
| <b>Average all runs</b>          | 0.0620        | 0.0327        | 0.3615        |

du maximum des valeurs précédentes comme seuil de sélection réduit le nombre de phrases sélectionnées ce qui augmente considérablement le gain. L'entropie donne un score élevé pour les phrases contenant des termes fréquents dans le flux et le filtre de redondance rejettera les phrases qui contiennent des mots fréquents dans le résumé. Ainsi, seules les phrases contenant un bon mélange de termes fréquents dans le flux et nouveaux par rapport au résumé seront sélectionnées.

Enfin, dans la figure 1, nous comparons la taille moyenne du résumé généré par chaque approche par rapport au nombre moyen d'informations vitales à retrouver pour les 15 topics. Nous observons que le résumé généré par l'approche proposée est plus petit (plus concis) que le résumé généré par les trois approches de l'état de l'art et l'approche 2APSal selon le résultat obtenus lors de la participation à TREC TS 2014 (Aslam *et al.*, 2014). La taille moyenne du résumé produit par KL-Entropie est de 171 alors qu'elle est de 381 et 340 pour 2APSal et hybrideTF-IDF respectivement. On note aussi que le nombre moyen d'informations vitales "golden nuggets" par topic est de 94. Ce résultat montre que notre approche génère un résumé concis ayant un gain meilleur avec une réduction de la couverture relativement peu significative.

## 5. Conclusion

La synthèse en temps réel d'événement est un sujet qui a connu récemment un engouement dans la recherche d'information. Dans cet article, nous avons proposé une nouvelle méthode de génération de résumé en temps réel. Les expérimentations ont été réalisées sur la collection de documents filtrés de la compagnie TREC TS 2014. Les résultats révèlent que l'approche décrite donne des meilleures performances en terme de précision et de moyenne harmonique entre le rappel et la précision par rapport aux systèmes ayant participé à TREC TS 2014 et trois approches de l'état de l'art.

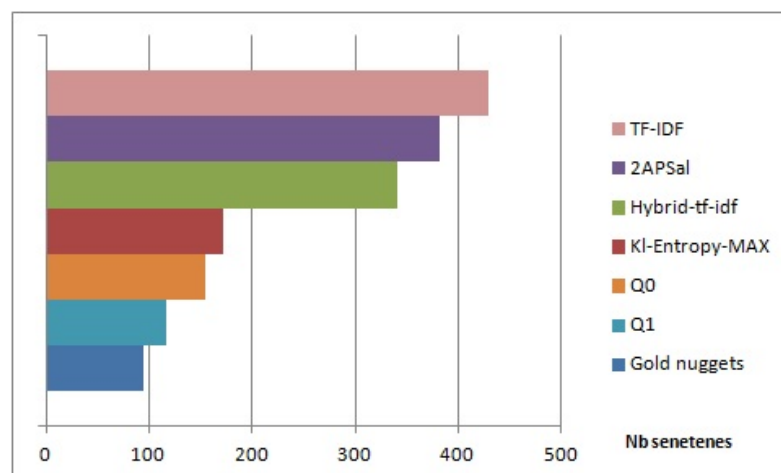


**Figure 1.** Nombre moyen de phrases dans le résumé par sujet

Notre approche est indépendante de l'événement, la génération de résumé est réalisée d'une manière incrémentale en temps réel au lieu de catégoriser les messages en sous-événements.

Ces résultats montrent que les statistiques textuelles sur le flux peuvent être exploités dans le filtrage et la génération de synthèse en temps réel. Le résumé obtenu est court et compact avec une bonne précision. Cependant la couverture diminue avec l'augmentation de la précision. Pour y remédier, d'autres pistes sont à explorer notamment en ce qui concerne le choix du seuil de sélection et l'identification d'autres dimensions tel que la diversité comme filtre à ajouter pour améliorer la qualité du résumé généré.

## 6. Bibliographie

- Aslam J. A., Ekstrand-Abueg M., Pavlu V., Diaz F., McCreadie R., Sakai T., « TREC 2014 Temporal Summarization Track Overview », *Proceedings of The Twenty-Third Text REtrieval Conference, TREC, 2014, Gaithersburg, Maryland, USA, November 19-21, 2014*, 2014.
- Chakrabarti D., Punera K., « Event Summarization Using Tweets », *Proceedings of the Fifth International Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, July 17-21, 2011*, 2011.
- Karkali M., Rousseau F., Ntoulas A., Vazirgiannis M., « Using temporal IDF for efficient novelty detection in text streams », *CoRR*, 2014.
- Kullback S., Leibler R. A., « On Information and Sufficiency », *The Annals of Mathematical Statistics*, vol. 22, n° 1, p. 79-86, 03, 1951.
- Liu F., Liu Y., Weng F., « Why is "SXSW" Trending ? : Exploring Multiple Text Sources for Twitter Topic Summarization », *Proceedings of the Workshop on Languages in Social Media, LSM '11*, p. 66-75, 2011.

- Liu X., Li Y., Wei F., Zhou M., « Graph-Based Multi-Tweet Summarization using Social Signals », *Proceedings of COLING 2012*, The COLING 2012 Organizing Committee, Mumbai, India, p. 1699-1714, December, 2012.
- Mackie S., McCreadie R., Macdonald C., Ounis I., « Comparing Algorithms for Microblog Summarisation », *Information Access Evaluation. Multilinguality, Multimodality, and Interaction - 5th International Conference of the CLEF Initiative, CLEF 2014, Sheffield, UK, September 15-18, 2014. Proceedings*, p. 153-159, 2014.
- Markou M., Singh S., « Novelty Detection : A Review - Part 1 : Statistical Approaches », *Signal Process.*, vol. 83, n° 12, p. 2481-2497, December, 2003.
- Nenkova A., Vanderwende L., The Impact of Frequency on Summarization, Technical Report n° MSR-TR-2005-101, MSR-TR-2005-101, January, 2005.
- O'Connor B., Krieger M., Ahn D., « TweetMotif : Exploratory Search and Topic Summarization for Twitter », *Proceedings of the Fourth International Conference on Weblogs and Social Media, ICWSM 2010, Washington, DC, USA, May 23-26, 2010*, 2010.
- Olariu A., « Hierarchical Clustering in Improving Microblog Stream Summarization », *Computational Linguistics and Intelligent Text Processing - 14th International Conference, CILing 2013, Samos, Greece, March 24-30, 2013, Proceedings, Part II*, p. 424-435, 2013.
- Olariu A., « Efficient Online Summarization of Microblogging Streams », *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2 : Short Papers*, Association for Computational Linguistics, Gothenburg, Sweden, p. 236-240, April, 2014.
- Ren Z., Liang S., Meij E., de Rijke M., « Personalized Time-aware Tweets Summarization », *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '13*, ACM, New York, NY, USA, p. 513-522, 2013.
- Shannon C. E., « A Mathematical Theory of Communication », *Bell System Technical Journal*, vol. 27, p. 379-423 , 623-656, October, 1948.
- Sharifi B., anthony Hutton M., Kalita J., « Automatic Summarization of Twitter Topics », in *National Workshop on Design and Analysis of Algorithm*, 2010a.
- Sharifi B., Hutton M.-A., Kalita J., « Summarizing Microblogs Automatically », *Human Language Technologies : The Conference of the North American Chapter of the Association for Computational Linguistics, HLT '10*, Association for Computational Linguistics, Stroudsburg, PA, USA, p. 685-688, 2010b.
- Sharifi B., Hutton M.-A., Kalita J. K., « Experiments in Microblog Summarization », *Proceedings of the 2010 IEEE Second International Conference on Social Computing, SOCIALCOM '10*, IEEE Computer Society, Washington, DC, USA, p. 49-56, 2010c.
- Shou L., Wang Z., Chen K., Chen G., « Sumblr : Continuous Summarization of Evolving Tweet Streams », *the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '13*, ACM, New York, NY, USA, p. 533-542, 2013.
- Xu W., Grishman R., Meyers A., Ritter A., « A Preliminary Study of Tweet Summarization using Information Extraction », *Proceedings of the Workshop on Language Analysis in Social Media*, Association for Computational Linguistics, Atlanta, Georgia, p. 20-29, June, 2013.
- Zubiaga A., Spina D., Amigó E., Gonzalo J., « Towards Real-time Summarization of Scheduled Events from Twitter Streams », *Proceedings of the 23rd ACM Conference on Hypertext and Social Media, HT '12*, ACM, New York, NY, USA, p. 319-320, 2012.



---

# Représentation Temporelle des Mots : Application au Clustering de Micro-Blogs

**Želko Kraljevic\*** — **Nicolas Baskiotis \*** — **Benjamin Piwowarski \***  
— **Patrick Gallinari\***

\* Sorbonne Universités, UPMC Univ Paris 06, CNRS, LIP6 UMR 7606, 4 place Jus-sieu 75005 Paris. Email: prenom.nom@lip6.fr

---

*RÉSUMÉ.* Les modèles distribués de mots sont un outil précieux pour la classification, le clus-tering, et plus généralement la représentation des documents. Pour des collections dynamiques, il est nécessaire de prendre en compte l'évolution temporelle de ces représentations. Dans cet article, nous présentons un modèle qui représente les mots sous la forme de trajectoires dans un espace de représentation, trajectoires qui sont déterminées par les groupes auxquels les mots appartiennent. Des expériences préliminaires en clustering sur des micro-blogs montrent l'intérêt de ce type de modèle.

*ABSTRACT.* Distributed term models are a powerful tool for classifying, clustering and represent-ing documents. For dynamic collections, we need to model both temporal and topical evolution. In this work, we present a model that uses a continuous time distribution and that represent words as trajectories in a continuous space. We perform some preliminary experiments on Twit-ter data, showing the potential of our model.

*MOTS-CLÉS :* représentation distribuée, temps, clustering, micro-blogs.

*KEYWORDS:* distributed representation, time, clustering, micro-blogs.

---

## 1. Introduction

La représentation distribuée (ou latente) des mots est un outil précieux pour explorer et analyser les collections de documents textuels. Pour des collections dynamiques, il faut prendre en compte à la fois les dimensions thématiques et temporelles. Ceci permet par exemple d'étudier l'évolution des idées dans la littérature scientifique (Hall *et al.*, 2008) ou des controverses (Beykikhoshk *et al.*, 2015). La plupart des modèles existants reposent sur une discrétisation du temps, ce qui limite leur utilité : ces modèles ne peuvent pas s'adapter de manière fine aux données car le choix de la granularité (heure, jour, etc.) a un impact important sur les solutions trouvées. Une granularité trop large (ex. un mois) ne permettra pas de détecter des événements courts, alors qu'une granularité trop fine (ex. une heure) peut à la fois introduire du bruit et nécessiter de plus grandes ressources (mémoire et temps de calcul).

Dans cet article, nous proposons un modèle dynamique de représentation des mots qui prédit la trajectoire de la représentation de mots dans un espace vectoriel thématique, et où les groupes (clusters) sont définis par une distribution sur le temps, ce qui définit leur influence temporelle, ainsi que par une transformation dans l'espace de représentation. Dans cet article, nous considérons deux transformations qui sont toutes deux des translations. Les transformations ainsi définies sont continues, i.e. la position d'un mot dans l'espace thématique est une fonction continue du temps.

(Mikolov *et al.*, 2013) a proposé un critère qui permet d'apprendre de manière non supervisée la représentation de mots dans un espace vectoriel. Dans cet article, nous proposons d'étendre ce critère pour intégrer le temps dans le contexte. Il est ainsi possible d'apprendre de manière non supervisée les paramètres de chaque groupe (pour le temps et la transformation), ainsi que les degrés d'appartenance des mots à ces groupes. Dans cet article, nous utilisons une collection de micro-blogs qui nous permet d'évaluer les qualités des représentations obtenues.

Dans la suite, nous présentons tout d'abord les travaux liés (section 2), avant de présenter le principe de l'apprentissage des représentations "Word2Vec" (section 3). Puis nous présentons notre modèle de représentation temporel (section 4) avant de présenter des résultats expérimentaux en clustering (section 5).

## 2. Travaux connexes

La représentation distribuée des mots a une longue histoire dans les domaines de l'accès à l'information et du traitement automatique du langage naturel. Des méthodes de factorisation matricielle (SVD ou LSA) ont été proposées dans (Deerwester *et al.*, 1990) pour trouver des thèmes latents dans une collection de documents. Plus récemment, porté par les succès des approches basées sur des réseaux de neurones pour les modèles de langues, (Mikolov *et al.*, 2013) ont proposé un modèle probabiliste simple, où la probabilité qu'un mot apparaisse à proximité d'un autre dépend directement du produit scalaire entre les représentations vectorielles des deux mots.

Il a été démontré empiriquement que ce modèle, en plus d’être rapide à apprendre, permettait de capturer de façon implicite des relations linéaires entre les mots.

Aucun de ces modèles ne prend par contre en compte le temps, ce qui est une limite pour étudier des collections qui évoluent avec le temps, comme les informations, les (micro-)blogs, ou la littérature scientifique. La plupart des extensions proposées supposent un temps discret en utilisant des fenêtres glissantes sur le temps (qui peuvent se chevaucher ou former une partition). (Saha et Sindhwani, 2012) utilisent une factorisation matricielle non négative pour chaque temps  $t$  avec une contrainte de régularité qui force les thèmes entre les temps  $t$  et  $t + 1$  à être proches. (Wang *et al.*, 2012) ont proposé de prédire l’évolution d’une distribution de probabilité sur les mots en utilisant une transformation linéaire pour modéliser les changements entre deux micro-blogs d’un utilisateur donné. Contrairement à ce type d’approches, le travail proposé ici ne suppose pas un temps discret, et est plus proche du travail de (Wang et McCallum, 2006). Ceux-ci utilisent une modification du modèle LDA (Blei *et al.*, 2003), où chaque thème est défini par une distribution de probabilité sur le temps ainsi que sur les mots. Dans notre travail, les groupes sont aussi définis en fonction du temps, mais contrairement à (Wang *et al.*, 2012), nous sommes intéressés par la représentation d’un mot à un temps  $t$  donné.

Finalement, nos travaux sont aussi liés au domaine de la détection de thèmes (Liu, 2009) qui a pour but de détecter des nouvelles informations et à les suivre. Les modèles utilisés pour cette tâche sont généralement basés sur un clustering incrémental. Par exemple, dans le cadre des micro-blogs, (Rosa *et al.*, 2011) regroupent les micro-blogs en utilisant l’algorithme des k-moyennes et LDA. Un exemple représentatif du regroupement incrémental est (Chen *et al.*, 2013) où à chaque pas de temps, un micro-blog peut être classifié dans un groupe pré-existant, ou un nouveau groupe peut être créé. Dans ce travail, nous évaluons notre modèle sur un corpus de micro-blogs, et nous le comparons au travail de (Rosa *et al.*, 2011).

### 3. Le modèle Word2Vec

Un des récents travaux à succès est le modèle *Word2Vec* (Mikolov *et al.*, 2013) qui propose une architecture de réseau de neurones simple permettant de prédire un terme dans une fenêtre contextuelle prédéfinie. Ce modèle est rapide à entraîner, et permet de calculer des représentations de mots sur de grandes collections. Il a été montré expérimentalement que cette représentation encode implicitement les relations syntaxiques et sémantiques entre les mots : par exemple, il existe une translation dans l’espace des représentations qui transforme la représentation d’un mot singulier (par exemple « ordinateur ») en sa représentation pluriel (« ordinateurs »).

Plusieurs modèles sont proposés dans (Mikolov *et al.*, 2013) ; dans ce travail, nous nous basons sur le modèle nommé *skip-gram* dont le fonctionnement est illustré Figure 1. Le modèle *skip-gram* a pour but, pour n’importe lequel des mots des documents, de prédire les mots apparaissant dans un voisinage proche de ceux-ci. Dans l’exemple

de la figure 1, le modèle skip-gram utilise la représentation du mot 1 pour prédire la probabilité d'occurrence des mots 2 ou 3.

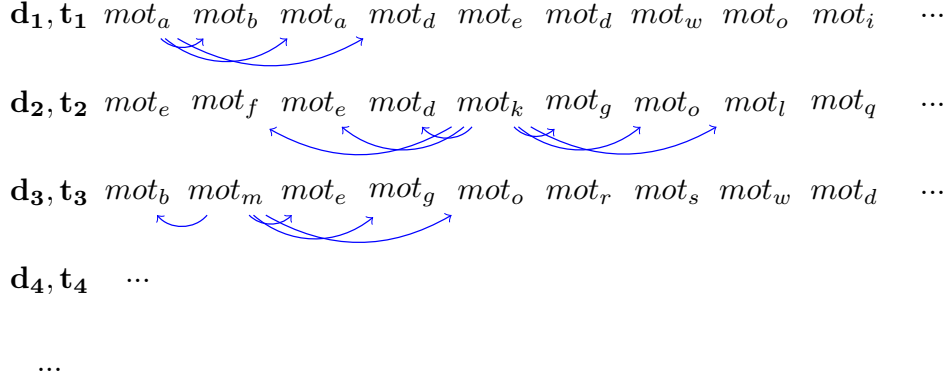


**Figure 1.** Documents observés dans une collection de documents. Chaque document correspond à une ligne, et les premiers mots du documents sont donnés pour les trois premiers : le document  $d_1$  commence par le mots  $a, b, a, d...$

La fonction objectif définie dans word2vec (Mikolov *et al.*, 2013) est la vraisemblance d'observer des mots sachant leur contexte :

$$\mathcal{L} = \prod_{(a,b) \in \mathcal{D}} p(b | a)^{\#a} \quad [1]$$

où  $\mathcal{D}$  est l'ensemble des observations, ou plus précisément l'ensemble des couples  $(a, b)$  tels que le mot  $a$  apparaisse près de  $b$  (à une distance inférieure à un seuil fixé à 5 comme dans (Mikolov *et al.*, 2013)), et  $\#a$  est le nombre de fois où  $a$  apparait dans la collection. Dans l'exemple de la figure 1, les couples  $(a, b)$ ,  $(a, a)$ ,  $(k, f)$  font partie de  $\mathcal{D}$ .

Le lien avec la représentation des mots est donné par la façon avec laquelle est calculée la probabilité  $p(b | a)$ , à savoir

$$p(b|a) = \frac{1}{K_a} \exp(x_a \cdot y_b) \quad [2]$$

où  $x_a$  et  $y_b$  sont les représentations des mots, et  $K_a$  une constante de normalisation. Notons qu'il y a deux représentations : une pour le contexte et l'autre pour la cible.

En pratique, une approximation de cette fonction objectif est utilisée, car calculer la normalisation  $K_a$  est trop coûteux en temps. Dans cet article, nous suivons la méthodologie de (Mikolov *et al.*, 2013) en utilisant un échantillonnage négatif (NEG)

$$\mathcal{C} = \sum_{(a,b) \in \mathcal{D}} (\log \sigma(x_a \cdot y_b) + k \mathbb{E} [\log \sigma(-x_a \cdot y'_b)]) \quad [3]$$

où  $\sigma(x) = 1/(1 + \exp(-x))$  est la fonction sigmoïde et l'espérance est estimée sur un échantillon de taille <sup>1</sup>  $k$ . Cette fonction objectif remplit le même rôle que le maximum de vraisemblance : elle rapproche les représentations  $x_a$  et  $y_b$  des mots qui co-occurrent, et éloigne les représentations  $x_a$  et  $y_{b'}$  des mots qui ne co-occurrent pas.

#### 4. Représentation Distribuée Temporelle des Mots

Une première méthode pour étendre le modèle (Mikolov *et al.*, 2013) serait d'utiliser un temps discret, et d'avoir une représentation de chaque mot pour chaque temps donné. Outre le problème du choix de la granularité de temps, cette solution est coûteuse en espace nécessaire pour stocker la représentation de tous les mots.

La solution que nous proposons permet d'éviter ces deux problèmes, en utilisant un temps continu et des groupes de mots (non exclusifs). La classe de modèles que nous proposons est basée sur l'intuition suivante : les mots sont représentés dans un espace thématique, comme pour dans les autres modèles distribués. La différence principale est que cette "position" n'est pas fixe mais dépend du temps.

Plus précisément, les mots ont des trajectoires dans cet espace thématique ; ces trajectoires sont définies par un ensemble de groupes  $\mathcal{C}$  appris comme décrit plus loin (section 4.3). Chaque groupe  $c \in \mathcal{C}$  est associé avec des paramètres liés au temps, qui définissent *avec quelle magnitude* la position d'un mot est modifiée à un temps  $t$  donné, ainsi qu'à des paramètres liés à l'espace thématique, qui définissent *comment* la position d'un mot est modifiée. Dans la suite, nous définissons deux modèles de transformation différents.

Formellement, la représentation  $x_a^{(t)}$  d'un mot  $a$  au temps  $t$  est donnée par :

$$x_a^{(t)} = x_a + \sum_{c \in \mathcal{C}} \rho_{ac} f_c(t) g_c(x_a) \quad [4]$$

Cette expression est composée de deux termes. Le premier, la position centrale  $x_a$  du mot  $a$ , définit le point de l'espace autour duquel la représentation du terme va fluctuer. Le second terme définit la transformation opérée sur cette représentation en fonction du temps – et plus particulièrement en fonction des groupes auxquels le mot appartient :  $\mathcal{C}$  est l'ensemble des groupes,  $\rho_{ac}$  est la probabilité avec laquelle le mot  $a$  appartient au groupe  $c$ ,  $f_c$  est une fonction qui définit l'influence du groupe  $c$  sur la représentation des mots à un temps  $t$  (probabilité du groupe d'être actif au temps  $t$ ), et finalement,  $g_c$  est une fonction qui spécifie comment transformer la représentation  $x_a$  d'un mot  $a$ . Les paramètres  $\rho$  sont appris (voir 4.3), et la représentation initiale  $x_a$  est donnée par word2vec.

1. En suivant l'article de Mikolov, nous avons choisi une valeur de  $k = 5$

Nous proposons plus loin deux modèles de transformation et utilisons par contre dans les deux cas une même fonction  $f_c$ , définie formellement par :

$$f_c(t) = \exp^{-|t_c-t|\tau} \quad [5]$$

où  $\tau$  est un paramètre définissant l'influence temporelle du groupe<sup>2</sup> et  $t_c$  est le temps associé au groupe  $c$ . Le paramètre  $\tau$  contrôle d'une certaine façon la granularité du temps considérée pour un groupe donné. Plus  $\tau$  est grand, plus l'influence temporelle est concentrée autour de  $t_c$ .

Dans cet article, nous fixons  $\tau$  à une valeur qui correspond aux événements principaux présents dans le corpus étudié, mais ce paramètre pourrait être appris. Outre l'apprentissage du paramètre  $\tau$ , d'autres modèles temporels pourraient être considérés, pour pouvoir par exemple capturer des événements cycliques ou brusques, en utilisant des fonctions périodiques.

Dans la suite, nous définissons deux types de transformations qui définissent l'influence des groupes dans l'espace des représentations.

#### 4.1. Modèle d'attraction

Le premier modèle que nous proposons est basé sur l'idée que la représentation d'un mot est "attirée" vers la représentation thématique  $u_c$  du groupe  $c$ . Formellement, la transformation est définie comme :

$$g_c(x_a) = u_c - x_a \quad [6]$$

où  $u_c$  est le vecteur associé au groupe  $c \in \mathcal{C}$ . En intégrant [6] dans [4], cela nous donne

$$x_a^{(t)} = x_a + \sum_{c \in \mathcal{C}} \rho_{ac} \exp^{-|t_c-t|\tau} (u_c - x_a) \quad [7]$$

Dans le cas où  $\rho_{ac}$  est égal à 1, et en ignorant l'influence des autres groupes, la représentation du mot  $a$  au temps  $t_c$  est donc exactement égale à  $u_c$  – et cette représentation retourne de façon continue à  $x_a$  quand la différence entre  $t$  et  $t_c$  augmente.

#### 4.2. Modèle de translation

Notre second modèle suppose que les mots appartenant à un groupe sont déplacés de manière similaire dans l'espace thématique. Formellement, la transformation est définie par :

$$g_c(x_a) = u_c \quad [8]$$

---

2. Nous n'apprenons pas  $\tau$  dans ce travail préliminaire

En intégrant [8] dans [4], cela nous donne

$$x_a^{(t)} = x_a + \sum_{c \in \mathcal{C}} \rho_{ac} \exp^{-|t_c - t| \tau} u_c \quad [9]$$

Ce modèle est moins adapté à la trajectoire des mots car celui-ci n’augmente que de manière indirecte la similarité (au sens du produit scalaire) entre deux différents mots, contrairement au modèle d’attraction. C’est toutefois un modèle plus simple qui pourrait être appris plus facilement.

### 4.3. Apprentissages des groupes

Afin d’apprendre la représentation des groupes (dans l’espace et le temps), ainsi que le degré d’appartenance des mots aux différents groupes  $\rho$ , nous nous basons sur la même fonction objectif que word2vec (section 3) que nous modifions afin de prendre en compte le temps :

$$\mathcal{L} = \prod_{(a,b,t) \in \mathcal{D}} p(b \mid a, t) \quad [10]$$

où l’ensemble d’apprentissage  $\mathcal{D}$  correspond à des triplets  $(a, b, t)$  pour lesquels le mot  $b$  apparaît à proximité de  $a$  au temps  $t$ . Par exemple, en prenant la figure 1, les triplets  $(a, b, t_1)$ ,  $(a, a, t_1)$ ,  $(k, f, t_2)$  et  $(k, e, t_2)$  appartiennent à  $\mathcal{D}$ .

Comme dans (Mikolov *et al.*, 2013), la probabilité  $p(b \mid a, t)$  dépend de la similarité entre deux mots dans l’espace vectoriel, à savoir

$$p(b \mid a, t) \propto \exp(y_a^{(t)} \cdot x_b^{(t)}) \quad [11]$$

où la représentation de deux mots est définie par le modèle d’attraction ou de translation.

Les deux formules précédentes montrent qu’il est possible d’apprendre tous les paramètres décrits plus haut – les paramètres des groupes ( $u_c$  et  $t_c$ ), le degré d’appartenance  $\rho_{ac}$  de chaque mot  $a$  au groupe  $c$  – en utilisant le principe de la maximisation de la vraisemblance, ou plus précisément le critère basé sur un échantillonnage négatif. Dans ce travail, nous pré-calculons la position centrale  $x_a$  et  $y_a$  de chaque mot en utilisant le logiciel word2vec<sup>3</sup> afin de réduire le nombre de paramètres à apprendre – mais il est tout à fait possible d’optimiser les deux jeux de paramètres en même temps.

## 5. Application : Clustering

Une fois les représentations des mots et des groupes apprises, notre modèle peut être utilisé pour regrouper les documents, ce qui est utile à des fins de visualisation et

3. <https://code.google.com/p/word2vec/>

d'évaluation. Dans cet article, nous utilisons les groupes pour évaluer notre modèle en terme de regroupement de documents (*clustering*).

En utilisant l'hypothèse du "Naïve Bayes", i.e. en supposant que l'occurrence d'un mot et le temps sont indépendants si le groupe est connu, et en supposant que chaque groupe a la même probabilité *a priori* :

$$p(c|d) = \frac{p(c)p(t_d|c) \prod_{a \in d} p(c|a)p(a)/p(c)}{p(d)} \propto f_c(t_d) \prod_{a \in d} \rho_{ac}$$

où  $d$  est un document associé à un temps  $t_d$ , et où le produit est défini sur l'ensemble des mots qui apparaissent dans le document  $d$ . Nous assignons ensuite le document  $d$  au groupe  $c_d$  qui a la plus grande probabilité  $p(c|d)$ .

### 5.1. Protocole expérimental

Nous avons utilisé une collection de 12 millions de micro-blogs (Yang et Leskovec, 2011), un sous-ensemble de micro-blogs de Twitter de juin 2009, que nous avons téléchargé en utilisant l'API de twitter<sup>4</sup>. Ces documents peuvent être écrits avec n'importe quelle langue, et traiter de n'importe quel thème. Nous avons tout d'abord nettoyé le jeu de données de la façon suivante. Nous avons enlevé les micro-blogs avec moins de 8 mots, ce qui a réduit l'ensemble à 10 millions de documents. Pour ceux restant, nous avons enlevé les nombres, la ponctuation, les hyperliens, les liens utilisateurs (@*user*) et étiquettes (les *hashtags* de la forme #*tag*).

Nous avons gardé l'association entre les micro-blogs et les étiquettes afin d'évaluer les groupes trouvés automatiquement par les différents algorithmes, en supposant qu'une étiquette correspond à un groupe donné.

Les groupes des micro-blogs ont été calculés en suivant la méthodologie décrite dans la section 5 pour notre modèle, ou en utilisant l'algorithme des k-moyennes. Des expériences préliminaires avec le modèle (Wang *et al.*, 2012) ont montré qu'il était difficile d'obtenir des résultats satisfaisants avec l'implémentation actuelle de cet algorithme<sup>5</sup>. Pour les k-moyennes, nous avons utilisé une représentation de chaque micro-blog dans l'espace de représentation thématique donné par word2vec, en sommant les vecteurs représentant chaque mot contenu dans le micro-blog.

Les groupes de références ont été constitués en faisant l'hypothèse qu'un groupe est défini par les micro-blogs contenant une étiquette donnée. Étant donné le grand nombre d'étiquettes dans ce corpus, nous avons sélectionné des sous-ensembles de taille réduite (environ 100 étiquettes), en suivant la méthodologie suivante. Tout

4. Les identifiants des tweets sont disponibles sur demande

5. Le corpus sur lequel il était évalué est bien plus petit ; nos premières tentatives pour adapter l'algorithme à une grande collection de tweets ont été infructueuses



d’abord, les étiquettes présentes dans moins de 500 micro-blogs ont été enlevées. Ensuite, nous avons généré les ensembles suivants :

**Random** Un ensemble choisi aléatoirement (probabilité uniforme) ;

**Top** L’ensemble des étiquettes les plus fréquentes ;

**H.C.** Un ensemble d’étiquettes choisies à la main parmi les plus fréquentes – en essayant de favoriser celles correspondant à un événement couvrant une courte période (quelques jours) ;

**G.I. 0.80, 0.85 et 0.90** Les étiquettes pour lesquelles l’index de Gini (Gini, 1912) est au dessus d’un certain seuil. L’index de Gini a été calculé en se basant sur la fréquence d’occurrence d’une étiquette pour chaque jour. Plus une étiquette apparaît de manière uniforme au cours des jours, plus son index de Gini est bas, et plus sa distribution est biaisée, plus son index de Gini est haut. Nous avons choisi de manière empirique les seuils de 0.8, 0.85 et 0.90 qui correspondent à des distributions très “inégalitaires” (quelques jours).

Les trois premiers ensembles (Random, Top, HC) ont plus de chance de favoriser l’algorithme des K-moyennes, car ils correspondent à des thèmes qui ont été traités tout au long de la période correspondant au mois de juin, contrairement aux derniers (GI 0.80, 0.85 et 0.90) qui correspondent plus à des thèmes s’étendant sur quelques jours, et qui *a priori* doivent être mieux détectés par notre modèle.

Pour mesurer la qualité des groupes trouvés, nous avons utilisé la mesure  $V$  (Rosenberg et Hirschberg, 2007), une mesure qui compare les groupes trouvés et ceux de référence en calculant la moyenne harmonique de deux mesures à valeurs dans  $[0, 1]$  :

1) L’homogénéité, qui est liée à l’entropie conditionnelle  $H(C|K)$  d’un groupe  $C$  étant donné les classes  $K$ , est d’autant plus grande que le groupe trouvé est “pur”, i.e. qu’il ne contient que des documents avec les mêmes étiquettes ;

2) La complétude, qui est liée à l’entropie conditionnelle  $H(K|C)$ , est d’autant plus grande que les documents associés à la même étiquette sont dans le même groupe trouvé.

Tout comme la précision et le rappel, augmenter l’homogénéité a tendance à faire diminuer la complétude, et vice-versa.

## 5.2. Apprentissage

Tous les modèles sont initialisés avec un nombre de groupes  $K$  fixé à 1000 afin que les modèles puissent détecter un nombre suffisant de thèmes. Les paramètres initiaux des groupes ont été initialisés de manière aléatoire, en utilisant une distribution uniforme pour les temps associés aux groupes ( $t_c$ ). Nous avons utilisé un espace de dimension 100 pour les modèles de représentations de mots. La représentation centrale

des mots  $x_a$  a été pré-calculée en utilisant word2vec<sup>6</sup> sur le même jeu de données, et n’a pas été modifiée par notre algorithme afin de réduire le temps de calcul et d’obtenir des solutions plus stables. Notons que nous n’avons utilisé qu’une seule représentation par mot, contrairement à (Mikolov *et al.*, 2013) où deux représentations sont utilisées pour chaque mot.

Pour apprendre le modèle, nous avons suivi la méthodologie de (Mikolov *et al.*, 2013), en utilisant un échantillon d’exemples “négatifs” (*Negative Sampling Strategy*) :

$$\sum_{(a,b,t) \in \mathcal{D}} \left( \underbrace{\log \sigma \left( x_a^{(t)} \cdot x_b^{(t)} \right)}_{u_p} + k \mathbb{E} \log \sigma \left( -x_a^{(t)} \cdot x_c^{(t)} \right) \right) \quad [12]$$

où  $u_p$  représente les exemples “positifs” (observés) et rapproche la représentation des mots  $a$  et  $b$  qui co-occurrent ;  $u_c$  représente les exemples “négatifs”, et éloigne la représentation de  $a$  et  $c$ . Dans notre cas, les mots vont se rapprocher (ou s’éloigner de manière similaire) en (1) de façon locale, en augmentant leur appartenance aux mêmes groupes (2) de façon globale, en déplaçant les temps associés aux groupes  $t_c$  vers les moments où les mots qui appartiennent à ces groupes co-occurrent.

Contrairement à (Mikolov *et al.*, 2013), notre but est de calculer la représentation des groupes et non des mots. Nous utilisons une descente de gradient, où nous mettons à jour de manière alternée les appartenances aux groupes ( $\rho$ ) et les centres des groupes ( $u_c$  et  $t_c$ ), en utilisant un pas de gradient de 3e-3 (pour  $u_c$  et  $\rho$ ) et 3e3 (pour  $t_c$ ) qui ont été déterminés expérimentalement. Nous avons utilisé une valeur de  $\tau$  de 5e-5, ce qui veut dire, étant donné que le temps est mesuré en secondes, que  $f_c$  a une valeur de respectivement 0.84 et 0.01 quand la différence entre  $t$  et  $t_c$  est respectivement une heure et un jour.

### 5.3. Résultats

Dans la table 1, nous donnons la mesure V pour les différents jeux d’étiquettes sélectionnés et modèles. Comme attendu, nos modèles se comportent mieux lorsque les étiquettes choisies correspondent (explicitement ou implicitement) à des événements qui couvrent peu de jours. La différence avec l’algorithme des k-moyennes est très importante pour ces quatre jeux d’étiquettes, ce qui signifie que les groupes trouvés, même s’ils ne sont que le sous-produit du modèle de trajectoire de mots, sont capables de capturer de manière satisfaisante les événements ainsi que les mots qui les décrivent.

Au niveau de nos modèles, le modèle d’attraction obtient des résultats qui sont meilleurs (0.03 à 0.05 points de différence), sauf dans le cas d’étiquettes choisies aléatoirement ou suivant leur fréquence. Cela correspond à l’hypothèse que nous avons

6. <http://word2vec.googlecode.com>

posée – la similarité des mots dans l’espace thématique est mieux prise en compte lorsque les mots convergent tous vers le centre du groupe ; ce qui était moins évident était que cela aurait une influence sur la qualité des groupes trouvés. Notons que cette différence s’inverse lorsque les jeux d’étiquettes ne sont plus ceux favorisant les modèles temporels.

| Étiquettes \ Modèle | Attraction | Translation | K-moyennes |
|---------------------|------------|-------------|------------|
| G.I. > 0.80         | 0.48       | 0.45        | 0.43       |
| G.I. > 0.85         | 0.52       | 0.48        | 0.44       |
| G.I. > 0.90         | 0.56       | 0.51        | 0.47       |
| H.C.                | 0.47       | 0.44        | 0.43       |
| Top 100             | 0.16       | 0.18        | 0.30       |
| Random 100          | 0.15       | 0.18        | 0.29       |

**Tableau 1.** *Mesure-V pour différents modèles et jeux d’étiquettes – le meilleur résultat pour chaque ensemble d’étiquettes est présenté avec un fond vert*

Afin de voir plus en détail le comportement des modèles, les tableaux 2 et 3 donnent respectivement les valeurs de l’homogénéité et de la complétude, et montrent l’algorithme des k-moyennes a une meilleure homogénéité (sur tous les jeux d’étiquettes) mais une complétude bien plus faible (surtout pour les jeux d’étiquettes correspondant à des événements courts). Nous pensons que cela est dû au fait que l’algorithme des k-moyennes n’utilise pas l’information temporelle, et peut potentiellement classer des micro-blogs de mêmes dates dans des groupes différents, alors que nos modèles, qui utilisent l’information temporelle, vont avoir tendance à mettre dans les même groupe des documents écrits à la même date.

| Étiquettes \ Modèle | Attraction | Translation | K-moyennes |
|---------------------|------------|-------------|------------|
| G.I. > 0.80         | 0.52       | 0.45        | 0.56       |
| G.I. > 0.85         | 0.58       | 0.51        | 0.59       |
| G.I. > 0.90         | 0.65       | 0.55        | 0.65       |
| H.C.                | 0.49       | 0.43        | 0.56       |
| Top 100             | 0.19       | 0.21        | 0.43       |
| Random 100          | 0.21       | 0.19        | 0.47       |

**Tableau 2.** *Homogénéité pour différents modèles et jeux d’étiquettes – le meilleur résultat pour chaque ensemble d’étiquettes est présenté avec un fond vert*

Nous avons également regardé, pour le modèle d’attraction, l’influence de la sélection du paramètre  $\tau$ . Le tableau 4 donne les mesures d’homogénéité pour le jeu d’évaluation G.I.> 0.90 et le modèle d’attraction. Un  $\tau$  plus petit implique que les groupes influencent la représentation sur moins de jours ; cela entraîne une homogénéité moindre. D’un autre côté la complétude augmente, mais sans compenser la perte en homogénéité. Dans nos expériences, un  $\tau$  trop petit entraîne une chute significative

| Étiquettes \ Modèle | Attraction | Translation | K-moyennes |
|---------------------|------------|-------------|------------|
| G.I. > 0.80         | 0.45       | 0.45        | 0.34       |
| G.I. > 0.85         | 0.47       | 0.47        | 0.34       |
| G.I. > 0.90         | 0.49       | 0.48        | 0.36       |
| H.C.                | 0.43       | 0.44        | 0.35       |
| Top 100             | 0.14       | 0.16        | 0.24       |
| Random 100          | 0.13       | 0           | 0.21       |

**Tableau 3.** Complétude pour différents modèles et jeux d'étiquettes – le meilleur résultat pour chaque ensemble d'étiquettes est présenté avec un fond vert

des performances. Il est donc important, dans les travaux futurs, d'apprendre automatiquement ce paramètre.

| mesure\ $\tau$    | 5e-5 | 1e-5 |
|-------------------|------|------|
| homogénéité       | 0.65 | 0.54 |
| <b>complétude</b> | 0.49 | 0.53 |

**Tableau 4.** Influence de différentes valeurs de  $\tau$  (le jeu d'évaluation G.I. > 0.90 et le modèle d'attraction)

| Date         | Nom                   | Mots                                                                                                                                                                      |
|--------------|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 26 juin 2009 | Mort de M. Jackson    | jackson, mj, michael, rt, thriller, pop, jackson's, tribute, king, death, brown, farrah, rip, men, died, mj's, dead                                                       |
| 29 juin 2009 | Hommages à M. Jackson | awards, was, bet, rt, just, chris, man, tribute, mj, joe, brown, jackson, drake, looks, sing, performance, stage, tiny, out, lil, died, show, fan, now, off, king         |
| 21 juin 2009 | Élections en Iran     | ppl, violence, panic, iran, protest, regime, tehran, reader, baby, iranian, seven, police, father's, protesters, symbol, neda, gov, killed, changing, dignity, revolution |

**Tableau 5.** Groupes trouvés par la méthode proposée en section 5. Les mots avec le  $\rho$  le plus haut sont donnés plus haut (après avoir enlevé les mots sans sémantique). Les noms des groupes ont été déterminés manuellement.

Finalement, nous donnons quelques exemples de groupes trouvés dans le tableau 5. Les groupes ont été sélectionnés en choisissant ceux qui étaient les plus importants (en terme de somme des  $\rho$  sur l'ensemble des mots). Ensuite, les mots dont l'appartenance au groupe était la plus forte ont été sélectionnés pour chaque groupe. Les deux premiers groupes sont tous deux liés à un aspect de la mort de Michael Jackson<sup>7</sup> - le

7. Il est mort le 25 juin 2009

premier contenant des micro-blogs datant du jour de sa mort, et les autres décrivant les “BET awards” qui se sont transformés en une commémoration de la mort de l’artiste.

## 6. Conclusion

Dans cet article, nous avons présenté un ensemble de modèles qui permettent d’apprendre des représentations de mots qui varient dans le temps en fonction de leur appartenance à des groupes (clusters) : chaque groupe définit une transformation de la représentation du mot dans l’espace qui est une déformation continue en fonction du temps. Nous avons ensuite détaillé deux modèles : dans le premier les mots sont attirés vers le centre du groupe, et dans le second la représentation des mots est modifiée par une translation. L’avantage de ce type d’approche est qu’il n’est plus nécessaire d’utiliser un temps discret (ce qui peut poser des problèmes si les événements ont des durées très différentes) et d’estimer la représentation d’un mot à chaque pas de temps (ce qui est coûteux en terme d’espace)

Les expériences que nous avons conduites sur un corpus de micro-blogs (issus de Twitter) ont montré que nos modèles détectaient mieux les thèmes, tels que définis par des étiquettes (*hashtags*), que l’algorithme des k-moyennes, à partir du moment où ces thèmes n’étaient pas constamment actifs dans le corpus.

Dans nos travaux futurs, nous apprendrons directement la fonction d’influence temporelle. Nous nous intéresserons également à l’utilisation de ces représentations temporelles pour d’autres tâches que le clustering, comme par exemple pour voir comment des concepts évoluent dans le temps. Il serait également intéressant d’utiliser ce type de modèles pour d’autres problèmes où une représentation distribuée d’entités est utilisée.

## Remerciements

REQUEST, projet investissement d’avenir, 2014-2017.

## 7. Bibliographie

- Beykikhoshk A., Arandjelović O., Venkatesh S., Phung D., « Hierarchical Dirichlet Process for Tracking Complex Topical Structure Evolution and Its Application to Autism Research Literature », *Advances in Knowledge Discovery and Data Mining*, Springer, 2015.
- Blei D. M., Ng A. Y., Jordan M. I., « Latent dirichlet allocation », *Journal of Machine of Machine Learning*, vol. 3, p. 993-1022, 2003.
- Chen Y., Amiri H., Li Z., Chua T., « Emerging topic detection for organizations from micro-blogs », *ACM SIGIR ’13*, 2013.
- Deerwester S., Dumais S. T., Furnas G. W., Landauer T. K., Harshman R., « Indexing by latent semantic analysis », *Journal of the American Society for Information Science*, vol. 41, n° 6, p. 391-407, September, 1990.

- Gini C., « Variabilità e mutabilità », 1912.
- Hall D., Jurafsky D., Manning C. D., « Studying the History of Ideas Using Topic Models », *EMNLP*, 2008.
- Liu N., « Topic Detection and Tracking », *Encyclopedia of Database Systems*, 2009.
- Mikolov T., Sutskever I., Chen K., Corrado G. S., Dean J., « Distributed Representations of Words and Phrases and their Compositionality. », *NIPS'14*, vol. cs.CL, p. 3111-3119, 2013.
- Rosa K. D., Shah R., Lin B., Gershman A., Frederking R., « Topical Clustering of Tweets », *SIGIR SWSM Workshop*, 2011.
- Rosenberg A., Hirschberg J., « V-measure : A conditional entropy-based external cluster evaluation measure », *EMNLP-CoNLL*, p. 410-420, 2007.
- Saha A., Sindhwani V., « Learning evolving and emerging topics in social media : a dynamic NMF approach with temporal regularization », *ACM WSDM'12*, 2012.
- Wang X., McCallum A., « Topics over time : a non-Markov continuous-time model of topical trends », *Proceedings of the 12th ACM SIGKDD*, 2006.
- Wang Y., Agichtein E., Benzi M., « TM-LDA : efficient online modeling of latent topic transitions in social media », *ACM KDD '12*, 2012.
- Yang J., Leskovec J., « Patterns of temporal variation in online media », *WSDM*, ACM, 2011.

---

# Tweet Contextualization Based on Wikipedia and Dbpedia

**Meriem Amina Zingla<sup>\*\*\*</sup> — Chiraz Latiri<sup>\*\*\*</sup> — Yahya Slimani<sup>\*</sup> — Catherine Berrut<sup>\*\*\*\*</sup> — Philippe Mulhem<sup>\*\*\*\*</sup>**

<sup>\*</sup> University of Carthage, INSAT, LISI research Laboratory, Tunis, Tunisia

<sup>\*\*</sup> University of Tunis El Manar, Faculty of Sciences of Tunis

<sup>\*\*\*</sup> University of Tunis El Manar, Faculty of Sciences of Tunis, LIPAH research

<sup>\*\*\*\*</sup> Grenoble Alpes University, LIG laboratory, MRIM group, Grenoble, France

---

**ABSTRACT.** Bound to 140 characters, tweets are short and not written maintaining formal grammar and proper spelling. These spelling variations increase the likelihood of vocabulary mismatch and make them difficult to understand without context. This paper falls under the tweet contextualization task that aims at providing, automatically, a summary that explains a given tweet, allowing a reader to understand it. We propose different tweet expansion approaches based on Wikipedia and Dbpedia as external knowledge sources. These proposed approaches are divided into two steps. The first step consists in generating the candidate terms for a given tweet, while the second one consists in ranking and selecting these candidate terms using a similarity measure. The effectiveness of our methods is proved through an experimental study conducted on the INEX 2014 collection.

**RÉSUMÉ.** La taille des tweets est limitée à un nombre maximum de caractères. Cette contrainte liée à la taille du message entraîne l'utilisation d'un vocabulaire particulier rendant le tweet difficile à comprendre. La tâche de contextualisation des tweets vise à fournir, automatiquement, un résumé qui explique un tweet donné, ce qui permet au lecteur de bien le comprendre. Nous proposons pour cela différentes méthodes basées sur deux énormes sources de connaissances à savoir, Wikipédia et Dbpedia. L'efficacité de notre méthode est prouvée par une étude expérimentale menée sur la collection d'INEX 2014.

**KEYWORDS:** Tweet contextualization, Association rules, INEX, Explicit Semantic Analysis, Query Expansion.

**MOTS-CLÉS :** Contextualisation des tweets, Expansion de requêtes, INEX Analyse Sémantique Explicite, Règles d'association

---

## 1. Introduction

Microblogging has emerged as one of the primary social media platforms for users to submit, in real-time, short messages, to report an idea or an opinion. Twitter is an online social networking service that enables users to tweet about any topic within the 140-character limit called *tweets*. However, this limit causes users to employ different strategies such as abbreviations and slangs in order to compress more information in minimum of characters. Tweets are, therefore, often misspelled or truncated and especially hard to understand.

To study this problem, INEX (Initiative for the Evaluation of XML retrieval) launched the tweet contextualization track for the first time in 2011. Thus, the tweet contextualization track proposed to answer questions of the form "What is this tweet about?" using a cleaned dump of Wikipedia, in order to allow the reader a better understanding of the tweet. The principle of this task is first to find documents that are related to the tweet (using a information retrieval system IRS), and then to generate an accurate summary of such documents (using an automatic summarization system ASS). INEX organizers provide the task participants with a baseline system that combines IRS and ASS. This baseline system takes as input a text query (tweet). This task allows participants to focus on the best tweet formulation for the IRS, since it has a direct impact on the summary quality.

Our goal here is to study tweet expansions as presented in Figure 1, and to evaluate the results using the INEX baseline system, on the INEX 2014 test collection. Several works have already proposed the use of the tweet expansion methods for the target task such as in (Morchid *et al.*, 2013), where the authors proposed to use Latent Dirichlet Analysis to expand the original tweets, and in (Zingla *et al.*, 2014), where the authors proposed to use association rules inter-terms, but these approaches do not include a term ranking step. The absence of this ranking step resulted in noisy queries containing unrelated terms to the original tweet. In this paper, we propose to expand the original tweets, using two external knowledge sources, namely, Wikipedia and Dbpedia. We opted to use Wikipedia because it is currently the largest knowledge repository on the Web. The use of this huge source is fruitful by allowing a massive knowledge representation of a tweet. We also use Dbpedia because it provides vast amounts of structured knowledge extracted from Wikipedia info boxes, hence, allowing to augment tweet representation with massive amounts of related information. Taking into account the weaknesses of the existing works cited before, we propose to enhance the tweet expansion process so that it will be composed of two steps : Tweet candidate terms generation, and candidate terms ranking and selection. While the first step extracts related information, i.e. terms, from Wikipedia and Dbpedia, the second step computes the semantic relatedness score between the original tweet and the candidate terms, using a new measure (ESAC) that relies on Explicit Semantic Analysis (ESA) and association rules.

The remaining of this paper is organized as follows: In section 2, our work is put in the context of related works, while section 3 gives a detailed description of



our tweet expansion methods. Section 4 presents our experimentations and results. Finally, section 5 is dedicated to the conclusion of this work and gives future works.

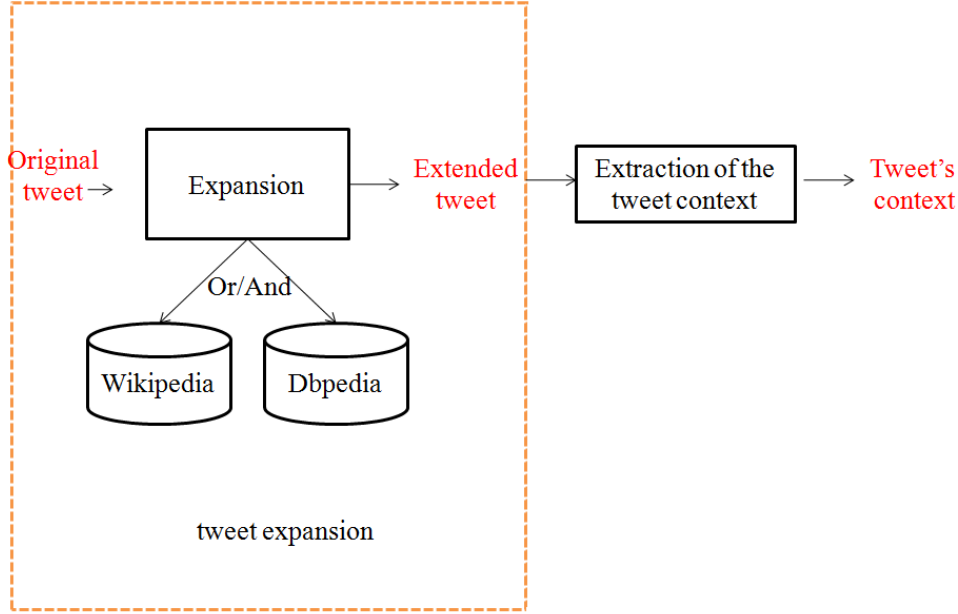


**Figure 1.** Explanatory schema of the proposed work

## 2. Related Work

In this section we review some related works, referring to query expansion and tweet contextualization.

### 2.1. Query expansion

Several works in the literature are proposed for the query expansion task, such as in (Song *et al.*, 2007) where the authors proposed a novel semantic query expansion technique that combines association rules with ontologies and Natural Language Processing techniques. This technique uses the explicit semantics as well as other linguistic properties of unstructured text corpus, it incorporates contextual properties of important terms discovered by association rules, and ontology entries are added to the query by disambiguating word senses. In (Latiri *et al.*, 2003), authors addressed query expansion by considering the term-document relation as fuzzy binary relations. Their approach to extract fuzzy association rules is based on the closure of an extended fuzzy Galois connection, using different semantics of term membership degrees. In (Shekarpour *et al.*, 2013), authors proposed an approach based on performing an initial retrieval of resources according to the original keyword query, the proposed process is

divided into three main steps. In the first step, all words closely related to the original keyword are extracted based on two types of features linguistic and semantic. In the second step, various introduced linguistic and semantic features are weighted using learning approaches. In the third step, they assign a relevance score to the set of the related words. Using this score they prune the related word set to achieve a balance between precision and recall. In these two previous works, relations weighting aspect are important. Authors in (Tan *et al.*, 2013) proposed a semantic approach that expands short queries by semantically related terms extracted from Wikipedia, they incorporate the expansion terms into the original query and adapt language models to evaluate the expanded queries.

The proposal of the paper is therefore, in a certain way, a continuation of these works, but in the case of tweet contextualization.

## **2.2. Query expansion for microblog retrieval**

Query expansion techniques are also used for microblog retrieval, authors in (Bandyopadhyay *et al.*, 2012), for example, used external corpora as a source for query expansion terms. Specifically, they used the Google Search API (GSA) to retrieve pages from the Web, and expanded the queries employing their titles. In (Lau *et al.*, 2011), authors proposed a twitter retrieval framework that focuses on topical features, combined with query expansion using Pseudo-Relevance Feedback (PRF) to improve microblogs retrieval results.

## **2.3. Tweet contextualization**

Despite the fact that the idea to contextualize tweets is quite recent, there are several works in this context. Recently, authors of (Ermakova and Mothe, 2012) proposed a method based on the local Wikipedia dump, they used the Term Frequency-Inverse Document Frequency TF-IDF cosine similarity measure enriched by smoothing from local context, named entity recognition and Part-Of-Speech weighting presented at INEX 2011. They modified this method by adding bigram similarity, anaphora resolution, hashtag processing and sentence reordering. The sentence ordering task was modeled as a sequential ordering problem, where vertices corresponded to sentences and sentence time stamps represented sequential constraints. They proposed a greedy algorithm to solve the sequential ordering problem based on chronological constraints. While in (Deveaud and Boudin, 2013a), authors used a method that allows to automatically contextualize tweets by using information coming from Wikipedia. They treated the problem of tweets contextualization as an automatic summarization task, where the text to resume is composed of Wikipedia articles that discuss the various pieces of information appearing in a tweet. One of the limitations of this approach is that the number of Wikipedia articles used to extract the candidate sentences is set manually. They explore the influence of various tweet-related articles retrieval methods as well as several features for sentence extraction. Whereas, in (Deveaud and Boudin, 2013b),

authors added a hashtag performance prediction component to the Wikipedia retrieval step. They used all available tweet features including web links which were not allowed by INEX's organizers.

In (Linhares, 2013), authors used an automatic summarizer named REG based on a greedy optimization algorithm to weigh the sentences. The summary is obtained by concatenating the relevant sentences weighed in the optimization step. In (Morchid *et al.*, 2013), authors used Latent Dirichlet Analysis (LDA) to obtain a representation of the tweet in a thematic space. This representation allows the finding of a set of latent topics covered by the tweet, this approach gives good results for the tweet contextualization task. Authors in (Zingla *et al.*, 2014) use the association rules between terms to extend the tweet, they project the terms of tweet on the rules' premises and add the conclusions to the original tweets.

Closely similar to our task, the authors in (Meij *et al.*, 2012), aim at adding semantics to microblog posts. They proposed a method that uses machine learning, and is based on a high-recall concept ranking and a high-precision concept selection step.

Finally, in (Torres-Moreno, 2014) authors developed three statistical summarizer systems the first one called Cortex summarizer, that uses several sentence selection metrics and an optimal decision module to score sentences from a document source, the second one called Artex summarizer, that uses a simple inner product among the topic-vector and the pseudo-word vector and the third one called REG summarizer which is a performant graph-based summarizer.

### **3. Tweet Expansion**

The tweet expansion aims at augmenting the thematic space of a given tweet by a massive amount of related terms. This is done to improve the IRS performance by giving more chance for a relevant document, which does not contain the original tweet terms, to be retrieved. In our work, we expand the original tweets in two steps (*cf.* Figure 2), namely:

- Step 1: Candidate terms generation.
- Step 2: Candidate terms ranking and selection.

The first step generates the candidate terms, for a given tweet, from Wikipedia and Dbpedia, the second one ranks these candidate terms according to their similarity to the tweet, and selects the best ones to be added. Three alternatives are proposed for the candidate terms generation step. Two are based on Wikipedia and one is based on Dbpedia.

#### **3.1. Step 1: Candidate Terms Generation**

Wikipedia is a huge knowledge source growing every day due to the contribution of people all around the world, it represents a very large, high quality, and valuable

Step 1: Candidate terms generation    Step 2: Candidate terms ranking and selection.

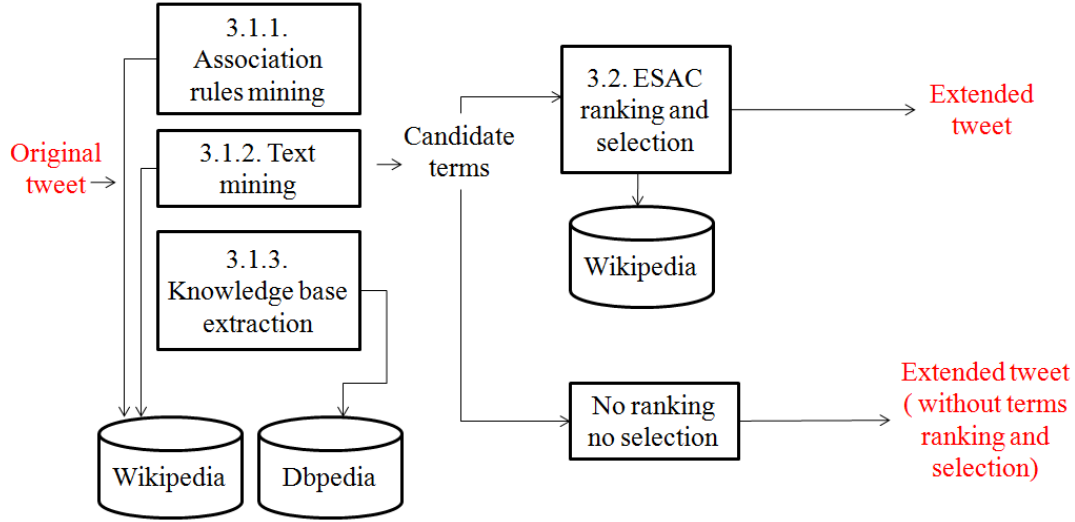


**Figure 2.** *The Proposed Methods for Tweet Expansion*

source in natural language, this is why we opted it to extend the tweets. We proposed to use two different methods to extract information related to a given tweet from this source. The first one is a statistical method based on association rules mining, the second one based on text mining. Authors, in (Tan *et al.*, 2013), have proved, in another frame, that the use of Wikipedia is interesting for the query expansion.

Dbpedia is a project aiming to represent Wikipedia content in RDF triples. It plays a central role in the Semantic Web, due to the large and growing number of resources linked to it.

We claim that the use of these knowledge sources will augment the tweet representation by a massive amount of related information. As DBpedia is structured and filtered, we expect less noisy expansions than with Wikipedia. However, using DBpedia may generate smaller expansions than Wikipedia. Using both sources is expected to select precise terms (from DBpedia) as well as broad terms (from Wikipedia). We propose to generate candidate terms for a given tweet using different methods based on these knowledge sources, Wikipedia and Dbpedia, namely: Association rules mining, text mining, and knowledge base extraction.

### 3.1.1. Association Rules Mining

An association rule binds two sets of terms, which respectively constitute its premise ( $T_1$ ) and conclusion ( $T_2$ ) parts. Thus, a rule approximates the probability of having  $T_2$  in a document, given that those of the  $T_1$  are already there. Compared to simple co-occurrences measures, association rules are then oriented, and we believe that this point is a must in our case, as we try to find new query terms from the initial

tweet.

The rule  $R$  is said to be based on the termset  $T$  equal to  $T_1 \cup T_2$ . The support of  $T$  is equal to the number of documents in  $C$  containing all the term of  $T$ . The support is formally defined as follows:

$$Supp(T) = |\{d | d \in C \wedge \forall t \in T : (d, t) \in I\}| \quad [1]$$

Where:

- $C$  is the whole set of documents which form the collection
- $d$  is a single document of the collection ( $d \in C$ )
- $\tau$  the whole set of distinct terms of the collection  $C$
- $T$  a set of terms of the collection ( $T \subseteq \tau$ )
- $t$  a single term of the collection ( $t \in \tau$ )
- $I \subseteq C \times T$  is a binary (incidence) relation. Each couple  $(d, t) \in I$  indicates that the document  $d \in C$  has the term  $t \in \tau$ .

The *confidence* of a rule  $R: T_1 \Rightarrow T_2$  is computed as:

$$Conf(R) = \frac{Supp(T)}{Supp(T_1)}. \quad [2]$$

An illustrative example of association rules is highlighted in Table 1.

**Table 1.** Association Rules examples taken from Wikipedia articles

| Premise     | Conclusion | Support | Confidence |
|-------------|------------|---------|------------|
| manufacture | car        | 356     | 0.8921     |
| manufacture | motor      | 301     | 0.9102     |
| campus      | university | 279     | 0.7431     |

An association rule  $R$  is said to be *valid* if its confidence value, *i.e.*,  $Conf(R)$ , is greater than or equal to a user-defined threshold denoted *minconf*. This confidence threshold is used to exclude non valid rules.

The process of generating candidate terms for a tweet is performed in the following steps:

- Selecting a sub-set of articles, from Wikipedia, similar to the tweet, using an algorithm based on the TF-IDF measure (Xia and Chai, 2011).
- Annotating the selected articles using TreeTagger. The choice of TreeTagger was based on the ability of this tool to recognize the nature (morpho-syntactic category) of a word in its context. TreeTagger uses the recursive construction of decision trees with a probability calculation to estimate the Part-Of-Speech of a word.
- Extracting specific terms, we extract terms that are nouns.

- Generating the association rules using an efficient algorithm called CHARM. We adapted the algorithm CHARM (Zaki and Hsiao, 2002), because it allows to generate non-redundant association rules (Yahia and Nguifo, 2004). As parameters, CHARM takes *minsupp* as the relative minimal support and *minconf* as the minimum confidence of the rules. While considering the *Zipf* distribution of the selected sub-set of Wikipedia articles, the minimal threshold of the support value is experimentally set in order to spread trivial terms which occur in the most of the documents, and are then related to too many terms.

- Obtaining the candidate terms for a given tweet, the candidate terms are the terms that appear in the conclusions of the irredundant association rules whose premise is included in the original tweet.

### 3.1.2. Text Mining

The second method to generate candidate terms for a given tweet from Wikipedia consists in exploring the Wikipedia articles related to the tweet, especially, the articles' first sentences that we call: definitions.

To achieve this, we use some heuristics, namely:

- Given a tweet, first, we search, in Wikipedia, all articles that correspond to the tweet's words. (Since the tweets are shorts, we consider all the words of tweet) This is done using WikipediaMiner<sup>1</sup>, which is a toolkit developed for tapping the rich semantics encoded within Wikipedia. This toolkit helps to integrate Wikipedia's knowledge into applications , by:

- Providing simplified, object-oriented access to Wikipedia's structure and content.
- Measuring how terms and concepts in Wikipedia are connected to each other.
- Detecting and disambiguating Wikipedia topics when they are mentioned in documents.
- We extract, from these articles, their corresponding definitions.
- We annotate these definitions using TreeTagger, then, we extract specific terms (nouns) from these annotated definitions. They are the candidate terms for the original tweet.

### 3.1.3. Knowledge Base Extraction

The Dbpedia Ontology currently contains about 4,233,000 instances. The table 2 below lists the number of instances for several classes with Class Instances

Dbpedia concepts are described by short and long abstracts in 13 different languages. These abstracts have been extracted from the English, German, French, Spanish, Italian, Portuguese, Polish, Swedish, Dutch, Japanese, Chinese, Russian, Finnish and Norwegian versions of Wikipedia

1. <http://wikipedia-miner.cms.waikato.ac.nz/>

**Table 2.** *Instances per class.*

|                   |           |
|-------------------|-----------|
| Resource(overall) | 4,233,000 |
| Place             | 735,000   |
| Person            | 1,450,000 |
| Work              | 411,000   |
| Species           | 251,000   |
| Organisation      | 241,000   |

We use information coming from Dbpedia to extend the original tweet. This method consists in extracting for each term in the original tweet, a set of related information from the Dbpedia ontology. These related information present the candidate terms for the original tweet. This is done using a SPARQL query: by *rdf : type*. SPARQL query is a semantic query language for databases, able to retrieve and manipulate data stored in Resource Description Framework (RDF) format.

### 3.2. Step 2 : Candidate Terms Ranking and Selection

This step consists in ranking the candidate terms according to their semantic relatedness to the given tweet, and selecting the best ones to be added. To achieve this, we propose a new semantic relatedness measure (*ESAC*) that combine the Wikipedia-based Explicit Semantic Analysis (*ESA*) measure and the association rules' confidence value.

Explicit Semantic Analysis (*ESA*) is a promising approach that calculates semantic relatedness proposed by Gabrilovich and Markovitch (Gabrilovich and Markovitch, 2007). It is a vectorial representation of text that uses Wikipedia as a knowledge base. Specifically, in *ESA*, a word is represented as a column vector in the TF-IDF matrix of the corpus and a text is represented as the centroid of the vectors representing its words. Semantic relatedness of two given texts can be obtained by calculating the correlation between two high dimensional vectors generated by *ESA*.

– The *ESAC* measure between a tweet *tw* and a term *w* is defined as:

$$ESAC_{tw,w} = \begin{cases} \alpha ESA_{tw,w} + (1 - \alpha) Conf_{R(tw,w)} & \text{if } \exists R(tw,w); \\ ESA_{tw,w}, & \text{otherwise.} \end{cases} \quad [3]$$

Where

-  $ESA_{tw,w}$  is the score of relatedness between the tweet *tw* and the candidate term *w* calculated as follow:

$$ESA_{tw,w} = \frac{tw.w}{\|tw\|^2 \|w\|^2}.$$

- $Conf_{R(tw,w)} = MAX \{Conf_{R_j(wt,w)}\}$  with  $MAX\{A\} = m$ ,  $m \in A$  is a maximal element of  $A$  if for all  $s \in A$ ,  $m \leq s$  implies  $m = s$ .
- $Conf_{R(wt,w)}$  is the confidence of the rule  $R$  that express the association between the candidate term  $w$  and a word in tweet  $wt$ .
- $\alpha$  is a weighting parameter  $\in [0, 1]$ .

We used the *ESA* implementation described in (Gabrilovich and Markovitch, 2007), we realized our runs with the best parameter value obtained by experiments. Once we calculate the semantic relatedness between the tweet and the candidate terms, we selecting the most related ones that have semantic relatedness score greater than a determined threshold and adding them to the original tweet.

## 4. Experimentations

Once, as described above, we have generated the extended tweets, we need to extract the tweet context. This context takes the form of an easy-to-read summary, composed of passages from a provided Wikipedia corpus.

To achieve this, as described in the Introduction, we use the system provided by the INEX 2014 Tweet Contextualization organizers<sup>2</sup> composed of : an Information Retrieval System (IRS) to find the most relevant Wikipedia articles, and an Automatic Summarizer System (ASS) to extract, from the relevant Wikipedia articles, the passages most representative of the tweet. This system was available to participants through a web interface or a Perl API. The system receives as input a query and returns a context. This latter consists of Part-Of-Speech (POS) sentences annotated with TreeTagger . This annotation process allows to assign a score for each sentence using TermWatch. The set of sentences, not exceeding 500 words (this limit is established by the organizers), defines the context of the tweet.

### 4.1. Data Collection

The tested INEX 2014 collections contains:

1) A corpus of 3 902 346 articles rebuilt in 2013 from a dump of the English Wikipedia of November 2012. All notes and bibliographic references that are difficult to handle are removed and only non-empty Wikipedia pages (pages having at least one section) are kept. Resulting documents are made of a title (title), an abstract (a) and sections (s). Each section has a sub-title (h). The abstract and sections are made of paragraphs (p) and each paragraph can have entities (t) that refer to Wikipedia pages. Each document is provided in XML format.

2. <http://qa.termwatch.es/data>



2) A collection of English tweets, composed of 240 tweets selected from the CLEF RepLab 2013. To focus on content analysis alone, urls are removed from the tweets.

#### 4.2. Runs

We conducted different runs (*cf.* Table 3), namely:

- (a) run-Wikipedia-textmining: Tweet expansion based on text mining without terms ranking and selecting step.
- (b) run-ESAC-Wikipedia-textmining: Tweet expansion based on text mining with terms ranking and selecting step.
- (c) run-Wikipedia-RA: Tweet expansion based on association rules mining without terms ranking and selecting step.
- (d) run-ESAC-Wikipedia-RA: Tweet expansion based on association rules mining with terms ranking and selecting step.
- (e) run-Dbpedia: Tweet expansion based on knowledge base (Dbpedia) without terms ranking and selecting step.
- (f) run-ESAC-Dbpedia: Tweet expansion based on knowledge base (Dbpedia) with terms ranking and selecting step.
- (g) run-ESAC-Wikipedia-Dbpedia: Combining run-ESAC-Wikipedia-textmining, run-ESAC-Wikipedia-RA and run-ESAC-Dbpedia.

**Table 3.** *The conducted runs.*

|                           | ESAC Ranking and selection |   | No ranking, no selection |
|---------------------------|----------------------------|---|--------------------------|
| Association rules mining  | g                          | d | c                        |
| Text mining               |                            | b | a                        |
| Knowledge base extraction |                            | f | e                        |

The parameters fixed for the experiments are:

$\alpha = 0.5$ ,  $minsupp = 15$ ,  $minconf = 0.7$ .

#### 4.3. Evaluation Metric

We have evaluated our runs according to the **Informativeness** metric, this latter is proposed by the INEX organizers, it aims at measuring how well the summary helps a user understand the tweets content. There are actually 47 tweets, from the 240 tweets, used for the INEX 2014 evaluation. For each tweet, each passage will be evaluated independently from the others, even in the same summary. The results are based on a thorough manual run on 1/5 of the 2014 topics using the baseline system. From this run two types of references were extracted, namely:

- a list of relevant sentences per topic.

– extraction of Noun Phrases from these sentences together with the corresponding Wikipedia entry.

The dissimilarity between a reference text and the proposed summary is given by:

$$Dis(T, S) = \sum_{t \in T} (P - 1) \times \left( 1 - \frac{\min(\log(P), \log(Q))}{\max(\log(P), \log(Q))} \right) \quad [4]$$

where :

- $T$ , a set of query terms present in reference summary.
- $S$ , a set of query terms present in a submitted summary.
- $f_T(t)$ , the frequency of term  $t$  in reference summary.
- $f_S(t)$ , the frequency of term  $t$  in a submitted summary.
- $P = \frac{f_T(t)}{f_T} + 1$ .
- $Q = \frac{f_S(t)}{f_S} + 1$ .

There are different distributions for the reference summaries, namely:

- Unigrams made of single lemmas (after removing stop-words).
- Bigrams made of pairs of consecutive lemmas (in the same sentence).
- Bigrams with 2-gaps also made of pairs of consecutive lemmas but allowing the insertion between them of a maximum of two lemmas (Also referred to as skip distribution).

#### 4.4. Results

proportion of the improvement / descent in the Bigrams with 2-gaps compared to run 361

##### 4.4.1. Within INEX 2014

Table 4 presents the official runs submitted by INEX 2014 participants from six countries (Canada, France, Germany, India, Russia, Tunisia).

We submitted the run: *run-Wikipedia-RA (361) (c)* to take part of the INEX 2014 competition, and we achieved the best informativeness results (the results are sorted by performance on Bigrams with 2-gaps and the lowest scores represent the best runs).

##### 4.4.2. After INEX 2014

After winning the INEX 2014 competition for the tweet contextualization task, we continued to improve our results.

Table 5 depicts our obtained results. As seen in the table, *run-ESAC-Wikipedia-Dbpedia (g)* has achieved the best informativeness results and has outperformed the

**Table 4.** *INEX Tweet Contextualization 2014 official informativeness results based on sentences.*

| Run     | Unigrams | Bigrams | Bigrams with 2-gaps | INEX system |
|---------|----------|---------|---------------------|-------------|
| 361 (c) | 0.7632   | 0.8689  | 0.8702              | YES         |
| 360     | 0.782    | 0.8925  | 0.8934              | YES         |
| 368     | 0.8112   | 0.9066  | 0.9082              | NO          |
| 369     | 0.814    | 0.9098  | 0.9114              | NO          |
| 359     | 0.8022   | 0.912   | 0.9127              | YES         |
| 370     | 0.8152   | 0.9137  | 0.9154              | NO          |
| 356     | 0.8415   | 0.9696  | 0.9702              | NO          |
| 357     | 0.8539   | 0.97    | 0.9712              | NO          |
| 364     | 0.8461   | 0.9697  | 0.9721              | -           |
| 358     | 0.8731   | 0.9832  | 0.9841              | NO          |
| 363     | 0.8682   | 0.9825  | 0.9847              | -           |
| 362     | 0.8686   | 0.9828  | 0.984               | -           |

**Table 5.** *The obtained informativeness results based on sentences.*

| Run | Unigrams | Bigrams | Bigrams with 2-gaps |
|-----|----------|---------|---------------------|
| g   | 0.7494   | 0.8520  | 0.8535              |
| b   | 0.7613   | 0.8629  | 0.863               |
| f   | 0.7610   | 0.8629  | 0.8638              |
| a   | 0.7665   | 0.8661  | 0.8668              |
| d   | 0.7612   | 0.8671  | 0.8695              |
| c   | 0.7632   | 0.8689  | 0.8702              |
| e   | 0.7940   | 0.8822  | 0.8831              |

other runs, this is due to the combination of the information coming from the two knowledge sources, Wikipedia and Dbpedia.

*run-ESAC-Wikipedia-textmining (b)*, *run-ESAC-Wikipedia-RA (d)*, *run-ESAC-Dbpedia (f)* outperform *run-Wikipedia-textmining (a)*, *run-Wikipedia-RA (c)*, *run-Dbpedia (e)* this is due to the term ranking step that reduced the noise in the extended tweets by eliminating the non related terms, and fine-grained the semantic representation of the tweets, indeed, the use of association rules that led to the enforcement of the relatedness score between the candidate terms and the tweet, ensured that the extended tweets contain adequate correlating terms with the initial ones and helped avoid inclusion of non-similar terms in them as much as possible, so the extended tweets were, to some extent, clean. We performed a bilateral paired Student t-test to evaluate the statistical significance of the differences of the averages between our best official run from INEX 2014 *run-Wikipedia-RA (361) (c)* from Table 4 and the best run from our proposals, *run-ESAC-Wikipedia-Dbpedia (g)* from table 5.

The differences, respectively for Unigrams, Bigrams or Bigrams with 2 gaps, are not significant according to the significance threshold of 0.05. However, we noticed that, for three topics (with tweet ids 257798105473380352, 262290292173045762 and 276815901897146368), the run- ESAC-Wikipedia-Dbpedia (g) largely underperforms the run-Wikipedia-RA (361) (c).

Table 6 presents the tweet with id 257798105473380352. Table 7 shows the expansions used for the runs run-Wikipedia-RA (361) (c) and run-ESAC-Wikipedia-Dbpedia (g) for the topic 257798105473380352. From this table, we find that the number of terms added for run-Wikipedia-RA (361) (c) is much smaller than for run-ESAC-Wikipedia-Dbpedia (g). In this large number of terms (exceeds 30) for the run-ESAC-Wikipedia-Dbpedia (g), noisy terms appear like "revenue", "english" and "country". The same observation is made on the two others topics mentioned above. We believe that this problem may be corrected in the future by limiting the number of added terms for the query expansion.

We propose to compare the runs run-Wikipedia-RA (361) (c) and run-ESAC-Wikipedia-Dbpedia (g), when our proposal run-ESAC-Wikipedia-Dbpedia (g) does not fail. We obtain the results presented in table 8. The respective differences between the run-Wikipedia-RA (361) (c) and our proposal run-ESAC-Wikipedia-Dbpedia (g) on these 44 topics are all statistically significant (noted † in table 8) according to bilateral paired Student t-tests with significance threshold of 0.05. This proves that, if we are able to detect in advance when our proposal fails, we can greatly improve the state of the art results.

**Table 6.** *The tweet with id 257798105473380352.*

|                                                                                                                                                                                                        |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>automotive Fiat S.p.A CNH-Fiat merger CNH rejects merger proposal from Fiat Industrial: US farm and industrial vehicle group CNH has rejected a merger proposal from its pa... 2012-10-15-13:00</p> |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

## 5. Conclusion and Perspectives

In this paper, we presented our works in the tweet contextualization field. We proposed different methods, based on Wikipedia and Dbpedia, to expand the tweets. Our proposed methods are divided into two steps, the first step generates the candidate terms and the second one ranks them and expands the original tweets by the most related ones. We conducted our experimentations on the INEX 2014 collection. The results we obtained through the different performed runs showed a significant improvement in the informativeness of the contexts, and have outperformed the winning run, which we submitted to the INEX 2014 tweet contextualization task. In our future work, we intend to extend our approach to take into account the tweets specificities (#,@,...). Furthermore, we mean to generalize our approach will

**Table 7.** The tweet (id= 257798105473380352) expansions by run-Wikipedia-RA (361) (c) and run-ESAC-Wikipedia-Dbpedia (g)

| Tweet id           | Expansion for run c                                                                              | Expansion for run g                                                                                                                                                                                                                                                                                                                 |
|--------------------|--------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 257798105473380352 | car vehicle government<br>acquisition model<br>market automaker auto<br>production america state | business car vehicle<br>manufacturing engine group<br>turin acquisition model<br>production market<br>auto fiat industry<br>design manufacture world<br>sector revenue factory<br>torino italy automotive<br>automobile manufacturer<br>sector revenue usa<br>company organisation<br>country english automaker<br>government world |

**Table 8.** Informativeness results based on sentences, on the 44 selected runs.

| Run | Unigrams | Bigrams  | Bigrams with 2-gaps |
|-----|----------|----------|---------------------|
| g   | 0.7364 † | 0.8439 † | 0.8454 †            |
| c   | 0.7800   | 0.8912   | 0.8923              |

contextualize normal (regular) queries by applying it on other data collections such as Cultural Heritage in CLEF collection (ChiC). This latter contains short queries that have no sufficient information to express their semantic.

**Acknowledgments.** This work is partially supported by the French-Tunisian project PHC-Utique RIMS-FD 14G 1404.

## 6. References

- Bandyopadhyay A., Ghosh K., Majumder P., Mitra M., “Query expansion for microblog retrieval”, *IJWS*, vol. 1, n<sup>o</sup> 4, p. 368-380, 2012.
- Deveaud R., Boudin F., “Contextualisation automatique de Tweets à partir de Wikipédia”, *CO-RIA 2013 - Conférence en Recherche d’Informations et Applications - 10th French Information Retrieval Conference, Neuchâtel, Suisse, April 3-5, 2013.*, p. 125-140, 2013a.
- Deveaud R., Boudin F., “Effective Tweet Contextualization with Hashtags Performance Prediction and Multi-Document Summarization”, *Working Notes for CLEF 2013 Conference, Valencia, Spain, September 23-26, 2013.*, 2013b.

- Ermakova L., Mothe J., “IRIT at INEX 2012: Tweet Contextualization”, *CLEF 2012 Evaluation Labs and Workshop, Online Working Notes, Rome, Italy, September 17-20, 2012*, 2012.
- Gabrilovich E., Markovitch S., “Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis”, *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007*, p. 1606-1611, 2007.
- Latiri C. C., Yahia S. B., jean-pierre chevallet, Jaoua A., “Query expansion using fuzzy association rules between terms”, *JIM’2003, France.*, 2003.
- Lau C. H., Li Y., Tjondronegoro D., “Microblog Retrieval Using Topical Features and Query Expansion”, *Proceedings of The Twentieth Text REtrieval Conference, TREC 2011, Gaithersburg, Maryland, November 15-18, 2011*, 2011.
- Linhares A. C., “An Automatic Greedy Summarization System at INEX 2013 Tweet Contextualization Track”, *Working Notes for CLEF 2013 Conference , Valencia, Spain, September 23-26, 2013.*, 2013.
- Meij E., Weerkamp W., de Rijke M., “Adding semantics to microblog posts”, *Proceedings of the Fifth International Conference on Web Search and Web Data Mining, WSDM 2012, Seattle, WA, USA, February 8-12, 2012*, p. 563-572, 2012.
- Milne D. N., Witten I. H., “An open-source toolkit for mining Wikipedia”, *Artif. Intell.*, vol. 194, p. 222-239, 2013.
- Morchid M., Dufour R., Linéars G., “LIA@inex2012 : Combinaison de thèmes latents pour la contextualisation de tweets”, *13e Conférence Francophone sur l’Extraction et la Gestion des Connaissances, Toulouse, France, 2013*.
- Shekarpour S., Höffner K., Lehmann J., Auer S., “Keyword Query Expansion on Linked Data Using Linguistic and Semantic Features”, *2013 IEEE Seventh International Conference on Semantic Computing, Irvine, CA, USA, September 16-18, 2013*, p. 191-197, 2013.
- Song M., Song I., Hu X., Allen R. B., “Integration of association rules and ontologies for semantic query expansion”, *Data Knowl. Eng.*, vol. 63, n° 1, p. 63-75, 2007.
- Tan K. L., Almasri M., Chevallet J., Mulhem P., Berrut C., “Multimedia Information Modeling and Retrieval (MRIM) /Laboratoire d’Informatique de Grenoble (LIG) at CHiC2013”, *Working Notes for CLEF 2013 Conference , Valencia, Spain, September 23-26, 2013.*, 2013.
- Torres-Moreno J., “Three Statistical Summarizers at CLEF-INEX 2013 Tweet Contextualization Track”, *Working Notes for CLEF 2014 Conference, Sheffield, UK, September 15-18, 2014.*, p. 565-573, 2014.
- Xia T., Chai Y., “An improvement to TF-IDF: Term Distribution based Term Weight Algorithm”, *JSW*, vol. 6, n° 3, p. 413-420, 2011.
- Yahia S. B., Nguifo E. M., “Approches d’extraction de règles d’association basées sur la correspondance de Galois”, *Ingénierie des Systèmes d’Information*, vol. 9, n° 3-4, p. 23-55, 2004.
- Zaki M., Hsiao C.-J., “An efficient algorithm for closed itemset mining”, *Second SIAM International Conference on Data Mining*, 2002.
- Zingla M. A., Ettaleb M., Latiri C. C., Slimani Y., “INEX2014: Tweet Contextualization Using Association Rules between Terms”, *Working Notes for CLEF 2014 Conference, Sheffield, UK, September 15-18, 2014.*, p. 574-584, 2014.



CORIA – CIFED 2016

Classification de documents  
CIFED

Président : *XX*





Papier absent

Extension automatique d'annotation et classification de documents en  
utilisant un modèle graphique probabiliste

Papier absent

# Papier absent

Extension automatique d'annotation et classification de documents en  
utilisant un modèle graphique probabiliste

Papier absent

# Papier absent



Papier absent

Classification et extraction des documents complexes à partir des images  
issues d'un périphérique mobile : application aux documents d'identité

Papier absent



Papier absent

Classification et extraction des documents complexes à partir des images  
issues d'un périphérique mobile : application aux documents d'identité

Papier absent

Papier absent



Papier absent

Papier absent

# Papier absent

Papier absent



## Papier absent





CORIA – CIFED 2016

Traitement de la langue  
CORIA

Président : *Jacques Savoy*



---

# Un modèle probabiliste pour la détection de l'incertitude dans le langage naturel

**Pierre-Antoine Jean\*** — **Sébastien Harispe \*** — **Sylvie Ranwez\*** — **Patrice Bellot\*\*** — **Jacky Montmain\***

\* *LGI2P, École des mines d'Alès, 69 rue Georges Besse F-30035 Nîmes cedex 1, {prenom.nom}@mines-ales.fr,*

\*\* *LSIS, Avenue Escadrille Normandie-Niemen F-13397 Marseille cedex 20, patrice.bellot@lsis.org,*

---

*RÉSUMÉ. La détection de l'incertitude dans le langage naturel est centrale pour le développement de nombreux modèles exploitant l'analyse de textes e.g. questions-réponses, raisonnement approché, enrichissement de bases de connaissances. Après une synthèse des différentes classifications de l'incertitude et des méthodes de détection correspondantes, cet article introduit une approche supervisée et générique de détection de l'incertitude. Celle-ci se base sur l'analyse statistique de différentes caractéristiques lexicales et syntaxiques afin de construire une représentation vectorielle d'une phrase analysable par des méthodes de classification éprouvées. L'évaluation que nous proposons tient compte des différentes dimensions de l'incertitude et de la nature des textes. Les résultats obtenus sur différents jeux de validation soulignent la performance globale de la méthode proposée et ouvrent de nombreuses perspectives.*

*ABSTRACT. Designing approaches able to automatically detect uncertain natural language expressions is central to design efficient models based on text analysis – for domains such as question-answering, approximate reasoning, knowledge-based population. This article proposes an overview of several contributions and classifications defining the concept of uncertainty expressions in natural language, and their related detection methods that have been proposed so far. A new supervised and generic approach is next introduced for this specific task; it is based on the statistical analysis of multiples lexical and syntactic features used to characterize sentences through vector-based representations that can be analyzed by proven classification methods. The global performance of our approach is demonstrated and discussed with regard to various dimensions of uncertainty and text specificities.*

*MOTS-CLÉS : Détection de l'incertitude, Classification binaire, Modèle supervisé.*

*KEYWORDS: Uncertainty detection, Binary classification, Supervised model.*

---

## 1. Introduction

Qu'elle soit d'ordre linguistique, numérique ou due à la subjectivité de certains jugements, l'incertitude est omniprésente dans toute situation langagière. En général levée par un *récepteur* humain qui réinterprète la phrase dans un contexte de co-énonciation (Fuchs, 2008), cette incertitude est beaucoup plus difficile à identifier de manière automatique dans des fragments de textes, qui peuvent de plus être sortis de leur contexte. Pourtant, ceux-ci peuvent être à la base d'un raisonnement approché ou, de façon plus globale, intégrés dans un processus décisionnel, pour ne citer que quelques-unes des applications possibles du traitement automatique des langues (TAL). La détection automatique de l'incertitude dans les textes a suscité un grand nombre de travaux ces dernières années et des événements majeurs comme la « *Conference on Natural Language Learning* » (CoNLL) en 2010 ont contribué au développement de méthodes dédiées. Leur intégration dans des applications d'analyse de sentiments, de recherche d'information, de questions-réponses ou encore d'extraction d'information à partir de textes a montré une réelle plus-value. Cependant, les formes diverses d'incertitude détectées ainsi que la forte dépendance de cette détection à la nature des textes analysés laissent largement ouvertes les perspectives de recherche dans ce domaine.

Caractériser l'incertitude renvoie rapidement à des dimensions diverses du texte. L'incertitude peut être interprétative par défaut ou excès de sens (présupposés, sous-entendus), ou bien provoquée par l'ambiguïté des termes (concurrence de sens, polysémie) (Fuchs, 2008). Plusieurs classifications de l'incertitude ont été proposées pour distinguer les différentes dimensions de l'incertitude et définir précisément celles qui peuvent être prises en compte dans un processus de détection automatique. (Jousselve *et al.*, 2003) présentent un état de l'art intéressant sur diverses classifications de l'incertitude. On y trouve notamment la classification de (Smets, 1997) proposée dans le domaine de la fusion de l'information, qui distingue l'incertitude comme une subdivision d'un concept plus général : l'imperfection de l'information. Si elle présente de façon claire les différentes formes d'incertitude, cette classification n'est cependant pas assez détaillée pour permettre une exploitation efficace dans la tâche de détection. Pour cela, on préférera la classification approfondie proposée par (Farkas *et al.*, 2010). Les auteurs y distinguent deux principales branches de l'incertitude : l'incertitude au niveau du discours et l'incertitude sémantique. **L'incertitude au niveau du discours** dénote dans la proposition du locuteur un manque d'information intentionnel ou non. Ainsi la proposition « Des personnes ont manifesté » appelle des compléments d'information : quelles personnes, combien étaient-elles ? (Ferson *et al.*, 2015). La subjectivité d'une proposition fait également partie de cette dimension de l'incertitude. Ainsi, l'incertitude au niveau du discours dépend principalement du contexte, du discours et de l'orateur ; en l'absence de connaissance sur ces différentes dimensions, l'incertitude persiste (Vincze, 2014). Par ailleurs, on appelle **incertitude sémantique** les propositions dont on ne peut pas déterminer la valeur de vérité étant donné l'état mental actuel du locuteur, soit le degré de confiance qu'il associe à sa proposition. Cette branche de l'incertitude se subdivise en deux catégories, d'une part l'incerti-

tude épistémique et d'autre part l'incertitude hypothétique. La principale différence entre ces deux catégories est que les propositions d'incertitude hypothétique peuvent être vraies, fausses ou incertaines *e.g. Il croit que la Terre est plate* (les connaissances actuelles du monde nous permettent d'infirmer cette proposition) tandis que les propositions d'incertitude épistémique sont définitivement incertaines *e.g. Il peut pleuvoir*, la factualité de la proposition ne peut être connue.

Cet intérêt pour l'incertitude est amplement justifié par le fait qu'elle est largement présente dans le langage naturel. (Light *et al.*, 2004) estiment que 11% des phrases dans les résumés des articles de MEDLINE sont incertaines. Toujours dans le domaine biomédical, le corpus de résumés BioScope (Szarvas *et al.*, 2008) contient 11871 phrases dont 2101 incertaines (17,5%), et si l'on s'intéresse à un domaine plus vaste, le corpus SFU (Konstantinova *et al.*, 2012) possède 17263 phrases tirées de documents dans divers domaines (films, livres, critiques) dont 23,7% incertaines.

Dans cet article, nous proposons une méthode de détection de l'incertitude basée sur une analyse statistique de différentes caractéristiques lexicales et syntaxiques. Cette méthode offre des résultats particulièrement intéressants lorsqu'elle est confrontée au processus de validation défini dans le cadre de la conférence CoNLL. Cette évaluation nous a permis différentes observations concernant, entre autres, l'influence de la nature des textes analysés et les dimensions de l'incertitude considérées. La section suivante analyse les différentes méthodes de détection d'incertitude citées dans la littérature, leurs caractéristiques et leurs performances. La section 3 détaille notre approche. Utilisant des techniques d'apprentissage automatique, cette approche matérialise les phrases comme des vecteurs de caractéristiques représentant des informations générales sur la phrase ou des informations basées sur les marqueurs lexicaux d'incertitude et leur contexte local. La section 4 présente les résultats de cette méthode confrontée aux jeux de tests de la conférence CoNLL 2010. Nous utilisons également des mesures éprouvées en classification de textes pour l'évaluer.

## 2. Les méthodes de détection de l'incertitude

De nombreux travaux, dans différents domaines, ont été consacrés à la détection des différentes formes d'incertitude et à leur prise en compte dans différentes applications de TAL, ce qui a permis d'en améliorer les performances. Par exemple, (Wu *et al.*, 2011) démontrent que la détection de l'incertitude permet d'améliorer la précision des informations extraites à partir de rapports radiologiques. Dans le domaine de l'analyse des sentiments, (Pang et Lee, 2004) ont montré que la détection de la subjectivité, considérée comme une forme d'incertitude au niveau du discours, aide à améliorer la classification de la polarité des phrases. En ce qui concerne les systèmes questions-réponses, (Ben Abacha, 2012) montre de manière empirique comment la détection de l'incertitude peut améliorer les performances du système MEANS.

L'incertitude s'exprime sous des formes diverses selon la nature des textes par l'emploi de verbes spéculatifs (suggérer, présumer), d'adjectifs et adverbess rap-

portant naturellement à l'incertitude (probablement, possible), d'auxiliaires modaux permettant d'exprimer une modalité (pouvoir, devoir) ou encore l'emploi de certains temps ou modes de conjugaison (subjonctif, conditionnel).

Différentes approches ont été suggérées dans le domaine de la détection automatique de l'incertitude. Ces approches se focalisent soit sur une détection binaire de la certitude d'une phrase, soit sur la détection de la portée des marqueurs d'incertitude au sein de la phrase. Un défi proposé lors de la conférence CoNLL 2010 a notamment permis de confronter différentes approches pour ces deux tâches. L'évaluation des méthodes était réalisée au travers de deux corpus : BioScope et WikiWeasel (Farkas *et al.*, 2010). BioScope est un corpus spécifique au domaine biomédical alors que WikiWeasel est un corpus généraliste constitué de paragraphes de Wikipedia (*cf.* tableau 1). La principale différence entre ces deux corpus au regard de la classification de (Szarvas *et al.*, 2012) est le type d'incertitude considéré. BioScope prend en compte uniquement l'incertitude sémantique tandis que WikiWeasel ajoute à celle-ci la prise en compte d'une partie de l'incertitude au niveau du discours, notamment au travers des mots *weasel* qui se rapportent à la notion de source dans le texte : *Qui dit ça ?* et à la part de subjectivité apportée par un contributeur de Wikipedia. L'identification automatique de ces mots *weasel* a été étudiée par (Ganter et Strube, 2009).

|            |                                                                                                                                                                      |
|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| BioScope   | <i>We <b>suggest</b> that these IL-10 producing effector T cells <b>may</b> contribute to clearing malaria infection without-inducing immune-mediated pathology.</i> |
| WikiWeasel | <i>He was <b>probably</b> born in Spain, but <b>some sources</b> say he was born in Quito.</i>                                                                       |

**Tableau 1.** Exemples de phrases issues du corpus BioScope et WikiWeasel. La phrase issue de BioScope révèle deux marqueurs d'incertitude épistémique et la phrase de WikiWeasel un marqueur d'incertitude épistémique et un autre d'incertitude au niveau du discours (manque d'information).

L'approche ayant obtenu les meilleurs résultats pour la tâche de détection binaire sur le corpus BioScope (une F-mesure de 86,4%) a été proposée par (Tang *et al.*, 2010). Leur méthode se base sur trois classifieurs disposés en deux couches. La première couche comprend un CRF (*Conditional Random Fields*) et un SVM (*Support Vector Machine*) se basant tous les deux sur un même ensemble de caractéristiques (mot, lemme, préfixe, suffixe, morphosyntaxe, syntagme) et un système d'étiquettes identiques (BIO). La seconde couche, quant à elle, est constituée d'un autre CRF et utilise des caractéristiques provenant des résultats de la première couche. Cette dernière couche réalise la détection finale des marqueurs et chaque phrase contenant un marqueur est annotée comme incertaine.

Pour le corpus WikiWeasel, (Georgescu, 2010) a proposé la meilleure approche avec une F-mesure de 60,2% en utilisant une classification par SVM basée sur une fonction *kernel RBF* (*Radial Basis Function*). Une méthode similaire a été mise en place dans (Cruz *et al.*, 2015) et a obtenu une F-mesure de 92,3% sur le corpus SFU



dont les annotations suivent celles proposées dans BioScope. On peut d’ores et déjà remarquer la différence de performance entre les meilleures méthodes en fonction de la nature du corpus (très spécialisé ou généraliste) et des dimensions de l’incertitude considérées. D’autres méthodes intéressantes de détection binaire ont été proposées et appliquées sur WikiWeasel. Par exemple, (Chen et Eugenio, 2010) ont présenté une méthode hybride en deux phases. La première réalise une recherche par motif de mots consécutifs, dont certains sont généralisés par leur *morphosyntaxe* à l’aide de Lucene<sup>1</sup>, pour récupérer des phrases candidates (potentiellement incertaines). La deuxième phase utilise ces phrases candidates comme entrées pour une classification par maximum d’entropie. Cette méthode a obtenu le troisième meilleur résultat sur 17 participants avec une F-mesure de 57,4%.

Les résultats obtenus à CoNLL entre les différents corpus nous dévoilent les limites des méthodes à être efficaces sur toutes les facettes de l’incertitude *e.g.* (Tang *et al.*, 2010) obtiennent 86,4% en F-mesure sur BioScope et 55% sur WikiWeasel correspondant à la meilleure moyenne (70.7%) de la conférence. Par conséquent, nous allons nous intéresser à la conception d’une méthode générique de détection de l’incertitude.

Dans la section suivante, nous présentons notre méthode basée sur une représentation vectorielle concise de la phrase – ce choix de représentation a été adopté afin d’éviter les biais de surapprentissage identifiés dans l’utilisation de représentations vectorielles de grandes tailles (Joachims, 2002) ; la taille réduite des vecteurs assure aussi une faible complexité de la tâche de classification, caractéristique souhaitée pour le traitement de gros volumes de données. La représentation vectorielle d’une phrase synthétise différentes statistiques propres à chaque caractéristique étudiée pour la détection d’incertitude (*e.g.* unigramme, bigramme). Plusieurs mesures fréquentistes sont ainsi proposées et évaluées pour le calcul de ces caractéristiques. Elles sont par la suite comparées aux mesures classiquement retrouvées dans la littérature associée à la classification de textes. Nous délégons ensuite la tâche de classification basée sur l’analyse des représentations vectorielles à un SVM (Sebastiani, 2002).

### 3. Un modèle probabiliste pour la détection de l’incertitude

#### 3.1. Vue d’ensemble du modèle

L’objectif est de distinguer si une phrase exprime de l’incertitude ou non – problématique de classification binaire. Pour cela, nous disposons d’un ensemble de phrases annotées  $S$  provenant des corpus BioScope, WikiWeasel ou SFU. De cet ensemble, il est possible d’extraire des informations sur les particularités lexicales et syntaxiques des phrases certaines et incertaines (*e.g.* la présence de marqueurs d’incertitude, les motifs morphosyntaxiques récurrents). Notre méthode propose de définir une représentation vectorielle sur un ensemble de caractéristiques d’une phrase. Chaque compo-

1. <https://lucene.apache.org>

sante du vecteur réalise une agrégation des poids affectés à une caractéristique locale dans la phrase (*e.g.* l'ensemble des unigrammes), en fonction d'une classe  $c$  à analyser (*e.g.* *est marqueur d'incertitude*). Cette représentation découle des méthodes de classification binaire de textes. En effet, dans le paradigme des méthodes d'apprentissage automatique, une des principales problématiques de la classification de textes est la manière de représenter un document. Celui-ci est généralement matérialisé comme un vecteur de poids associés à ses différentes caractéristiques pouvant être les mots d'un vocabulaire dans les approches les plus simples (Sebastiani, 2002). Ces poids ont pour objectif de sélectionner les caractéristiques les plus pertinentes d'une classe  $c$  afin de réduire l'espace des dimensions associé à l'ensemble des caractéristiques d'un corpus (Yang et Pedersen, 1997).

La sous-section suivante présente les différentes caractéristiques utilisées par notre méthode, et leur utilisation dans un modèle d'apprentissage supervisé. Les modalités d'évaluation et les résultats obtenus seront discutés dans la section 4.

### 3.2. Définition des caractéristiques locales et globales

Les fonctions caractéristiques sélectionnées et étudiées, bien que générales et se voulant indépendantes d'un domaine particulier, traduisent l'intuition que certains marqueurs lexicaux et sémantiques semblent importants pour la classification d'une phrase. Deux niveaux de granularité ont été considérés pour la définition de ces fonctions.

Le premier niveau s'intéresse aux spécificités globales d'une phrase traduisant potentiellement une expression d'incertitude *e.g.* nous considérons la taille d'une phrase car nous supposons que la longueur est un indice discriminant.

Le second niveau porte sur les motifs *n-grammes* qui composent une phrase. Nous entendons par motifs *n-grammes* les séquences de  $n$  éléments de même nature, par exemple, la forme lemmatisée des mots ou leur symbole morphosyntaxique (*PoS*). A chaque *n-gramme* est associé un poids exprimant le fait qu'il puisse traduire une expression d'incertitude. La composante de la projection de la phrase selon la caractéristique analysée tiendra compte de l'agrégation des poids associés aux *n-grammes* qui la composent. Chacune des caractéristiques est ainsi définie par un quadruplet (type, taille, contexte, agrégation) précisant le type de *n-gramme* analysé (lemme et motif morphosyntaxique), la taille des *n-grammes* ( $n$ ), le contexte, i.e. si le score des *n-grammes* se base sur la fréquence d'observations des *n-grammes* dans une phrase étiquetée incertaine ou comme marqueur explicite d'incertitude (ces derniers sont précisés dans le jeu de d'entraînement), et l'agrégation utilisée pour résumer les scores des différents *n-grammes* de la phrase pour cette caractéristique. Le tableau 2 résume les différentes caractéristiques basées sur l'analyse de *n-grammes*. La figure 1 détaille le calcul de deux caractéristiques. Une phrase est donc caractérisée par un vecteur de  $\mathbb{R}^6$  avant l'étape de classification – les cinq caractéristiques présentées dans le tableau 2 et la taille de la phrase ( $F_5$ ).

|       | Type  | Taille | Contexte                      | agrégation |
|-------|-------|--------|-------------------------------|------------|
| $F_1$ | Lemme | 1      | Marqueur d'incertitude        | somme      |
| $F_2$ | Lemme | 2      | Marqueur d'incertitude        | somme      |
| $F_3$ | Lemme | 1      | $\in$ à une phrase incertaine | somme      |
| $F_4$ | PoS   | 5      | $\in$ à une phrase incertaine | somme      |
| $F_6$ | Lemme | 1      | $\in$ à une phrase incertaine | max        |

**Tableau 2.** Description des caractéristiques locales utilisées.

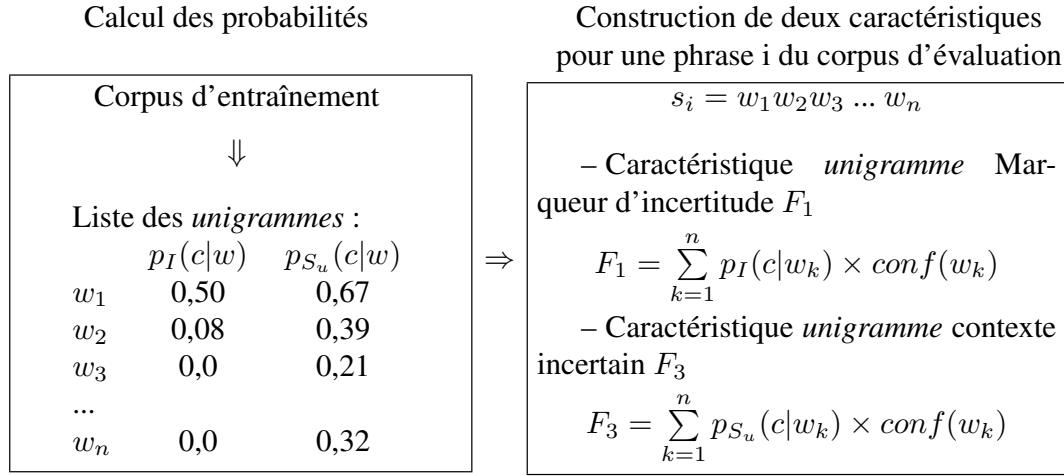


**Figure 1.** Calcul de deux caractéristiques d'une phrase  $s_i$  appartenant au corpus d'évaluation. La première phase (à gauche) permet de calculer les probabilités sur le corpus d'entraînement avec  $p_I(c|w)$  la probabilité conditionnelle qu'un lemme  $w$  soit marqueur d'incertitude et  $p_{S_u}(c|w)$  la probabilité conditionnelle qu'un lemme  $w$  soit présent dans une phrase incertaine. Ainsi, la signification de la classe  $c$  dépend de la probabilité conditionnelle considérée. La deuxième phase correspond à la construction de deux caractéristiques  $F_1$  et  $F_3$  définies dans le tableau 2. Le score de confiance  $\text{conf}$  est détaillé dans la sous-section 3.3.

La sous-section suivante présente les différentes mesures étudiées afin de calculer le score (poids) associé à chaque motif de *n-gramme*. Par commodité, nous illustrons désormais nos propos en considérant les observations en tant que marqueurs d'incertitude pour un lemme. Par conséquent, les exemples ne vont pas tenir compte du type *PoS*, du type lemme avec une taille supérieure à 1 et du contexte *appartient à une phrase incertaine* (cf. tableau 2). Nous partons également du postulat qu'un marqueur d'incertitude traduit une phrase incertaine.

### 3.3. Définition d'une mesure probabiliste

Les données d'entraînement définissent un ensemble de phrases incertaines  $S_u \subset S$  avec  $S$  l'ensemble des phrases. Ces données nous permettent d'obtenir pour chaque lemme  $w$  son nombre d'occurrences dans le corpus, noté  $\#_S(w)$ , son nombre d'occurrences dans les phrases incertaines,  $\#_{S_u}(w)$  avec  $\#_{S_u}(w) \leq \#_S(w)$  ainsi, que son nombre d'occurrences en tant que marqueur d'incertitude  $\#_{I_{S_u}}(w)$ , avec  $I_{S_u}$  l'ensemble des marqueurs d'incertitude du corpus et  $\#_{I_{S_u}}(w) \leq \#_{S_u}(w)$ . Connaissant le lemme  $w$ , nous pouvons alors définir la probabilité conditionnelle qu'il soit marqueur d'incertitude *i.e.* qu'il appartienne à la classe  $c$  (cf. équation 1). La signification de la classe  $c$  et la définition de cette probabilité dépendent du contexte de la caractéristique considérée (marqueur d'incertitude ou *appartient à une phrase incertaine* – cf. tableau 2).

$$p_I(c|w) = \#_{I_{S_u}}(w) / \#_S(w) \quad [1]$$

Cependant, l'analyse de cette probabilité dans le but de distinguer les marqueurs d'incertitude n'est pas suffisante. Du fait de la taille limitée des corpus d'entraînement, il est en effet fréquent d'obtenir des probabilités très élevées pour certains termes, malgré leur présence limitée dans le corpus d'entraînement. Prenons le cas extrême d'un lemme  $w$  qui n'apparaît qu'une seule fois dans le corpus et ce, de façon fortuite, dans un contexte incertain, sa probabilité d'appartenir à une phrase incertaine serait alors :  $p_I(c|w) = 1$ .

Afin de pallier cette limite, nous définissons un score de *confiance* associé à cette probabilité qui évalue la pertinence de considérer le motif analysé (ici le lemme  $w$ ) comme marqueur d'incertitude. Dans la modélisation de ce score de confiance, nous cherchons à considérer à la fois le nombre d'occurrences  $\#_S(w)$  et la probabilité  $p(c)$  qu'un lemme, observé dans l'ensemble des mots du corpus  $W$  et tiré aléatoirement, soit marqueur d'incertitude (cf. équation 2). Par conséquent, si un lemme obtient une forte probabilité d'être marqueur d'incertitude, la confiance dans ce score sera d'autant plus élevée que ce lemme est représentatif du corpus et que la probabilité  $p(c)$  est faible.

$$p(c) = \frac{\sum_{w \in W} \#_{I_{S_u}}(w)}{\sum_{w \in W} \#_S(w)} \quad [2]$$

Pour la modélisation de ce score de confiance, nous avons étudié deux mesures, possédant une sémantique propre, utilisant comme paramètre  $\#_S(w)$  et  $p(c)$  ainsi qu'une mesure témoin utilisant uniquement  $\#_S(w)$ . Le premier score de confiance étudié repose sur une loi de distribution binomiale cumulée utilisant :  $p(c)$ , la probabilité de tirer un marqueur d'incertitude dans  $W$ , le nombre d'occurrences  $\#_{I_{S_u}}(w)$  du mot  $w$  observé en tant que marqueur d'incertitude et le nombre d'occurrences  $\#_S(w)$  du mot  $w$  dans le corpus complet. Cette loi est définie par la probabilité de fonction de masse suivante, avec  $n = \#_S(w)$ ,  $k = \#_{I_{S_u}}(w)$  et  $p = p(c)$  :

$$p_b(X \geq k) = \sum_{i=k}^n \binom{n}{i} p^i (1-p)^{n-i} \quad [3]$$

Ainsi, la confiance associée à la probabilité conditionnelle  $p_I(c|w)$  est fonction de la probabilité d'effectuer un nombre d'observations identifiées comme marqueurs d'incertitude supérieur ou égal à  $\#_{I_{S_u}}(w)$  (loi cumulative) en effectuant  $\#_S(w)$  tirages aléatoires. Par conséquent, plus la valeur associée à la loi binomiale cumulative est élevée et moins  $p_I(c|w)$  traduit une incertitude. Le score de confiance est alors modélisé par  $1 - p_b(X \geq k)$ .

La seconde modélisation de la confiance que nous avons étudiée suppose intuitivement que plus la probabilité  $p(c)$  est grande, plus le nombre d'occurrences d'un motif doit être conséquent pour associer un score de confiance élevé à la probabilité qu'il soit marqueur d'incertitude. Cette représentation de la confiance peut être modélisée par une fonction sigmoïde de  $\#_S(w)$  dont le paramètre  $p(c)$  caractérise la courbure. Plus  $p(c)$  est grand et plus la pente de la courbe est lissée. (cf. figure 2).

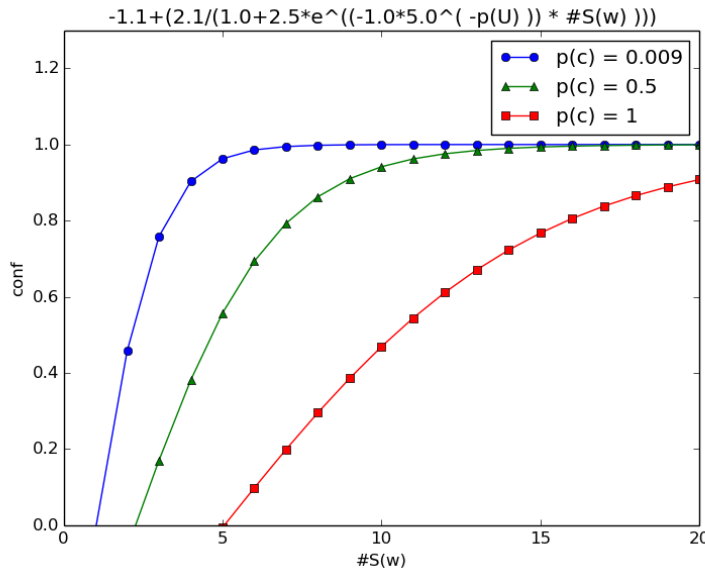


**Figure 2.** Modélisation de la confiance en fonction du paramètre  $p(c)$ . La courbe avec les ronds a un  $p(c) = 0.009$  correspondant à la probabilité pour un mot d'être marqueur d'incertitude dans le corpus BioScope. La courbe avec les triangles a un  $p(c) = 0.5$  et la courbe avec les carrés un  $p(c) = 1$ .

Finalement, la dernière modélisation de la confiance tient uniquement compte de  $\#_S(w)$ . Cette mesure, nous permet d'observer l'impact de la probabilité  $p(c)$  dans les modélisations précédentes. Elle signifie que plus un lemme est fréquent, plus la confiance qui lui sera accordée sera importante (cf. équation 4).

$$conf(w) = 1 - \frac{1}{\#_S(w)} \quad [4]$$

La fonction  $F_1$  utilisée pour calculer une des dimensions de la représentation vectorielle d'une phrase  $s$ , qui caractérise les unigrammes marqueurs d'incertitude est fonction de : la probabilité que l'unigramme traduise une forme d'incertitude, modulée par  $conf(w)$  la confiance associée à la probabilité de l'unigramme  $w$  (cf. équation 5). Cette formulation est généralisable à l'ensemble des caractéristiques.

$$F_1(s) = \sum_{k=1}^n p_I(c|w_k) \times conf(w_k) \quad [5]$$

### 3.4. Sélection automatique des caractéristiques optimales

Les vecteurs représentant les phrases sont par la suite utilisés comme entrée d'un modèle d'apprentissage automatique SVM<sup>2</sup>. Les natures très différentes des corpus BioScope, WikiWeasel et SFU font qu'ils n'ont pas le même ensemble de caractéristiques optimales dans le SVM (cf. tableau 3). Ainsi, une stratégie de sélection automatique des caractéristiques optimales à partir d'un corpus d'entraînement a été appliquée en suivant les travaux de (Chen et Lin, 2006). Ces travaux mettent en avant l'utilisation d'une forêt aléatoire. Ainsi, nous avons mis en place une procédure de sélection récursive où les caractéristiques en dessous d'un certain pourcentage d'importance sont supprimées jusqu'à obtenir les caractéristiques les plus pertinentes.

| Caractéristiques                                                 | BioScope | WikiWeasel | SFU |
|------------------------------------------------------------------|----------|------------|-----|
| $F_1$ - Unigramme, marqueurs d'incertitude                       | x        | x          | x   |
| $F_2$ - Bigramme, marqueurs d'incertitude                        |          | x          |     |
| $F_3$ - Unigramme, dans les phrases incertaines                  | x        | x          | x   |
| $F_4$ - Motifs <i>PoS</i> taille 5, dans les phrases incertaines |          | x          |     |
| $F_5$ - $ s $ la taille de la phrase $s$                         |          | x          |     |
| $F_6$ - $max$ (Unigramme marqueurs d'incertitude)                | x        | x          | x   |
| $F_7$ - Trigramme, marqueurs d'incertitude                       |          |            |     |

**Tableau 3.** Les caractéristiques optimales pour BioScope, WikiWeasel et SFU obtenues par l'application d'une forêt aléatoire. La caractéristique des trigrammes n'est jamais pertinente quel que soit le jeu de données.

Les travaux de (Øvrelid *et al.*, 2010) suggèrent que les caractéristiques syntaxiques ne sont pas nécessaires dans la tâche de détection de l'incertitude. Cependant, on

2. Ce modèle se base sur une fonction *kernel* RBF (Gaspar *et al.*, 2012) et est optimisé au niveau des paramètres  $C$  et  $\gamma$  selon l'étude réalisée par (Georgescul, 2010)

remarque que pour WikiWeasel la caractéristique des motifs *PoS* contenus dans les phrases incertaines est discriminante.

#### 4. Résultats et discussion

Cette section présente les résultats obtenus en utilisant la probabilité conditionnelle  $p_I(c|w)$  couplée avec les différentes définitions de la confiance présentées dans la section précédente : la loi binomiale cumulée, la fonction sigmoïde et la confiance témoin ( $1 - 1/\#_S(w)$ ). Les résultats sont ensuite comparés en modifiant la probabilité  $p_I(c|w)$  par des mesures éprouvées en théorie de l'information.

##### 4.1. Résultats de l'approche probabiliste

Le tableau 4 précise les résultats obtenus en utilisant la probabilité d'être marqueur d'incertitude couplée aux différents scores de confiance définis précédemment. Chaque entrée donne la précision, le rappel et la F-mesure obtenus pour chaque expérience.

|                 | BioScope                  | WikiWeasel                | SFU corpus                |
|-----------------|---------------------------|---------------------------|---------------------------|
| Loi binomiale   | 77,9 / 82,9 / <b>80,3</b> | 66,7 / 25 / 36,3          | 87,8 / 95,8 / 91,6        |
| Sigmoïde        | 75,8 / 82,1 / 78,8        | 73,8 / 43,6 / 54,8        | 88,2 / 96,6 / <b>92,2</b> |
| $1 - 1/\#_S(w)$ | 75,8 / 81,6 / 78,6        | 64,9 / 48,8 / <b>55,7</b> | 88,2 / 96,4 / 92,1        |
| Sans confiance  | 76,3 / 81,2 / 78,7        | 72,1 / 11,8 / 20,2        | 88,1 / 95,9 / 91,9        |

**Tableau 4.** Résultats de la méthode en utilisant différentes confiances associées à la probabilité  $p_I(c|w)$  sur les corpus BioScope, WikiWeasel et SFU. Les caractéristiques utilisées entre les différentes confiances sont fixées selon le jeu de données.

Ces résultats amènent plusieurs interprétations. Dans un premier temps, on remarque que les différents scores de confiance n'impactent pas ou peu les résultats sur les jeux de données BioScope et SFU, considérant tous les deux uniquement l'incertitude sémantique. Cependant, la confiance améliore les scores sur WikiWeasel. Une analyse fine de ces corpus nous permet d'observer que la disparité des marqueurs d'incertitude pour WikiWeasel est bien plus grande que pour BioScope, elle-même plus grande que celle de SFU. Cette forte disparité, représentée par le nombre, la nature et la distribution des marqueurs d'incertitude à l'échelle du corpus ajoute un bruit important dans ces données. Pour diminuer ce bruit, nous avons appliqué un filtre de pré-classification sur le nombre d'occurrences  $\#_{S_u}(w)$  des motifs *n-grammes* dans les phrases incertaines (cf. équation 6).

$$\#_{S_u}(w) = \begin{cases} \#_{S_u}(w) & \text{si } \#_{I_{S_u}}(w) \geq 1 \\ 0 & \text{sinon.} \end{cases} \quad [6]$$

Ce filtre impacte les caractéristiques considérant les lemmes et les motifs morpho-syntaxiques dans le contexte des phrases incertaines. Les résultats en appliquant ce filtre sont indiqués dans le tableau 5.

Dans un second temps, on s'aperçoit que les résultats pour la loi binomiale sur WikiWeasel sont moins bons qu'avec les autres scores de confiance. Ces résultats s'expliquent par la faible valeur de la probabilité de succès  $p(c)$  (e.g. 0,03 pour WikiWeasel) qui a pour effet d'augmenter le score de confiance de la loi binomiale.

Enfin, les résultats sur les différents corpus d'évaluation soulignent l'importance de la définition de l'incertitude que l'on souhaite détecter et de la nature des textes.

|                 | BioScope                  | WikiWeasel                | SFU corpus                |
|-----------------|---------------------------|---------------------------|---------------------------|
| Loi binomiale   | 76,2 / 82,9 / 79,4        | 61,3 / 61,5 / 61,4        | 88,1 / 96,7 / 92,2        |
| Sigmoïde        | 76,3 / 82,9 / <b>79,5</b> | 69,7 / 55,3 / 61,7        | 88,3 / 96,7 / <b>92,3</b> |
| $1 - 1/\#_S(w)$ | 76,3 / 82,9 / <b>79,5</b> | 68,9 / 57,7 / <b>62,8</b> | 88,3 / 96,7 / <b>92,3</b> |
| Sans confiance  | 76,2 / 82,9 / 79,4        | 64,7 / 54,5 / 59,2        | 88,3 / 96,7 / <b>92,3</b> |

**Tableau 5.** Résultats de la méthode en utilisant un filtre sur le nombre d'occurrences des lemmes présents dans le contexte des phrases incertaines. Les caractéristiques utilisées entre les différentes confiances sont fixées selon le jeu de données.

Les résultats du tableau 5 démontrent l'efficacité du filtre sur les résultats de WikiWeasel en éliminant la majeure partie du bruit issu des motifs présents dans les phrases incertaines. Finalement, ce filtre permet de compenser la faiblesse de la méthode lorsqu'elle est appliquée sur des jeux de données dont la disparité des marqueurs est forte.

Ces résultats améliorent ceux de l'approche de (Georgescu, 2010) sur le corpus WikiWeasel, qui avait obtenu la première place de la tâche 1 lors de CoNLL 2010 sur ce même corpus avec une F-mesure de 60,2%. De plus, nous obtenons la meilleure moyenne en terme de F-mesure, 71,2%, sur les jeux BioScope et WikiWeasel par rapport à la meilleure moyenne de la conférence, 70,7% par (Tang *et al.*, 2010). Au niveau du corpus de SFU, non-utilisé dans CoNLL 2010, nous avons des résultats similaires à ceux de (Cruz *et al.*, 2015).

## 4.2. Comparaison avec d'autres mesures

L'utilisation de la probabilité conditionnelle  $p_I(c|w)$  a été confrontée à des mesures couramment utilisées dans le domaine de la classification de textes. Ces métriques considèrent un lemme  $w$  et sa relation avec une classe  $c$ . L'ensemble des valeurs obtenues à ces différents tests sont données dans le tableau 6.

*Pointwise mutual information*, PMI, mesure l'association d'un lemme  $w$  avec la classe  $c$  (cf. équation 7). Cette mesure est proche de la définition de notre probabilité



$p_I(c|w)$ . Elle pondère simplement cette probabilité par  $p(c)$ . Cependant, cette probabilité  $p(c)$  est très faible lorsqu'on considère la classe *est marqueur d'incertitude* et aura pour conséquence de bruite la valeur de la probabilité.

$$pmi(w, c) = \log\left(\frac{p(c, w)}{p(c).p(w)}\right) = \log\left(\frac{p(c|w)}{p(c)}\right) \quad [7]$$

*Odds Ratio* mesure le degré de dépendance entre un lemme  $w$  et la classe  $c$  (cf. équation 8). Appliqué à nos données, le *Odds Ratio* favorise les motifs avec un faible écart entre  $\#_{I_{Su}}(w)$  et  $\#_S(w)$ .

$$orr(w, c) = \log\left(\frac{p(w|c).(1 - p(w|\bar{c}))}{p(w|\bar{c}).(1 - p(w|c))}\right) \quad [8]$$

*Categorical Proportional Difference*, CPD, est un ratio qui considère pour un lemme  $w$  le nombre de documents appartenant aux classes  $c$  et  $\bar{c}$  qui le contiennent. L'équation 9 définit CPD avec  $dw_c$  le nombre de documents de la classe  $c$  contenant  $w$ ,  $dw_{\bar{c}}$  le nombre de documents de la classe  $\bar{c}$  contenant  $w$ . Dans notre problématique de détection binaire de l'incertitude au niveau de la phrase, cette mesure a été adaptée,  $dw_c$  représente le nombre d'occurrences du lemme  $w$  en tant que marqueur d'incertitude.

$$cpd(w, c) = \frac{dw_c - dw_{\bar{c}}}{dw} \quad [9]$$

*Weighted Log Likelihood Ratio* mesure la dissimilarité de la distribution du lemme  $w$  en fonction des classes  $c$  et  $\bar{c}$  (cf. équation 10).

$$wllr(w, c) = p(w|c).log\left(\frac{p(w|c)}{p(w|\bar{c})}\right) \quad [10]$$

Nous avons couplé ces différentes métriques avec les mesures de confiance définies dans la sous-section 3.3. Ce couplage s'apparente à l'adaptation de modèles classiquement retrouvés pour la classification de textes. Par exemple, (Hamdan, 2015) définit le poids final d'un terme par la formule  $w_i = localWeight \times globalWeight \times normalization$  avec *localWeight* une mesure fréquentiste du terme dans le document (e.g.  $\log(termFrequency + 1)$ ), *globalWeight* une métrique appliquée aux termes à l'échelle du corpus (présentée en début de section) et *normalization* permet d'ajuster les poids en fonction de la taille du document. Les résultats sont présentés dans le tableau 6. Une analyse a également été menée en amont sur ces mesures ainsi que sur les mesures suivantes : *Chi Square*, *Natural Entropy* et *Kullback-Leibler Divergence*. Cette étude a porté sur l'analyse du comportement de chaque mesure par rapport à la contrainte principale fixée pour notre modèle. Nous l'avons vu, cette contrainte repose sur la prise en compte, lorsque la probabilité conditionnelle est fixe, du nombre d'observations  $\#_S(w)$  pour le calcul du score, tel que pour deux mots  $w_1$

| Métrique   | Confiance       | BioScope     | WikiWeasel | SFU          |
|------------|-----------------|--------------|------------|--------------|
| PMI        | $\log(\#_S(w))$ | 75,6%        | 33,8%      | 88,3%        |
|            | $1 - 1/\#_S(w)$ | 77,3%        | 40,6%      | 91,1%        |
|            | Loi binomiale   | 76,6%        | 52,3%      | 91,5%        |
|            | Sigmoïde        | 77,1%        | 37,7%      | 91%          |
|            | Sans confiance  | 76,4%        | 35,1%      | 90,6%        |
| Odds Ratio | $\log(\#_S(w))$ | 78,1%        | 45,5%      | 91,1%        |
|            | $1 - 1/\#_S(w)$ | <b>79,3%</b> | 52%        | 92,1%        |
|            | Loi binomiale   | <b>79,3%</b> | <b>55%</b> | <b>92,2%</b> |
|            | Sigmoïde        | <b>79,3%</b> | 51,5%      | 92,1%        |
|            | Sans confiance  | 79,2%        | 51,3%      | 92,1%        |
| CPM        | $\log(\#_S(w))$ | 70,8%        | 45,2%      | 78,6%        |
|            | $1 - 1/\#_S(w)$ | 70,4%        | 49,9%      | 78%          |
|            | Loi binomiale   | 69,7%        | 48,1%      | 80,1%        |
|            | Sigmoïde        | 70,5%        | 48,6%      | 78,1%        |
|            | Sans confiance  | 69,6%        | 48%        | 73,3%        |
| Wllr       | $\log(\#_S(w))$ | 53,7%        | 16,5%      | 69,8%        |
|            | $1 - 1/\#_S(w)$ | 55,1%        | 11%        | 66,3%        |
|            | Loi binomiale   | 55,5%        | 45%        | 67,1%        |
|            | Sigmoïde        | 55,1%        | 11,6%      | 65,8%        |
|            | Sans confiance  | 55,1%        | 18,9%      | 65,7%        |

**Tableau 6.** *F-mesure des confiances associées à différentes métriques globales étudiées sur les corpus BioScope, WikiWeasel et SFU. Les caractéristiques utilisées entre les différentes confiances sont fixées selon le jeu de données. Le filtre sur le nombre d'occurrences des lemmes présents dans les phrases incertaines est appliqué.*

et  $w_2$  avec  $\#_S(w_1) > \#_S(w_2)$  et  $p_I(c|w_1) = p_I(c|w_2)$  le score de la mesure soit supérieur pour  $w_1$ .

Dans la problématique de détection de l'incertitude, la probabilité conditionnelle  $p_I(c|w)$  obtient de meilleurs résultats comparée aux autres métriques globales. De plus, cette probabilité couplée avec la confiance  $1 - 1/\#_S(w)$  est la plus performante en moyenne sur les jeux de données.

## 5. Conclusion et Perspectives

Dans cet article, nous avons proposé une méthode d'apprentissage automatique pour la détection binaire de l'incertitude dans le langage naturel. Cette méthode se base sur une représentation vectorielle concise ( $\mathbb{R}^6$ ) de la phrase construite à partir des différentes probabilités conditionnelles de motifs *n-grammes* pondérées par un score

de confiance fréquentiste. L'approche obtient des résultats intéressants au regard de toutes les dimensions de l'incertitude.

Plusieurs pistes d'amélioration de la méthode sont envisagées. Ces pistes concernent notamment le calcul des poids des motifs *n-grammes*. En effet, un mécanisme de propagation basé sur l'analyse des collocations permettrait une pondération contextuelle plus précise (Lavalley *et al.*, 2010); ceci dans le but d'éviter des erreurs de classification dues au poids d'un lemme trop discriminant. Une autre piste d'amélioration serait d'ajouter une caractéristique contextuelle au niveau de la phrase *i.e.* indiquer par une valeur booléenne si la phrase précédente est détectée comme incertaine. On considère dans ce cas l'hypothèse qu'une phrase aura plus de chance d'être incertaine si les phrases précédentes sont incertaines. Une autre amélioration serait d'étendre la nature des motifs utilisés dans les caractéristiques. Actuellement, seulement deux types sont utilisés, les lemmes et les motifs morphosyntaxiques. Nous pourrions par exemple expérimenter les étiquettes d'un arbre des dépendances en tant qu'unité de base d'un motif ou élaborer un motif hybride de plusieurs types (Chen et Eugenio, 2010). Enfin, nous envisageons d'étendre les comparaisons effectuées dans cette étude en prenant notamment en compte des travaux récents dans le domaine des réseaux neuronaux démontrant des performances intéressantes dans la tâche de classification de la polarité des phrases (Tai *et al.*, 2015).

## 6. Bibliographie

- Ben Abacha A., « Recherche de réponses précises à des questions médicales : le système de questions-réponses MEANS », *PhD, Univ Paris Sud-Paris XI*, p. 162, 2012.
- Chen L., Eugenio B. D., « A Lucene and Maximum-Entropy Model based hedge detection system », *Fourteenth Conference on Computational Natural Language Learning*, p. 114-119, 2010.
- Chen Y. W., Lin C. J., « Combining SVMs with various feature selection strategies », *Feature extraction*, p. 315-324, 2006.
- Cruz N., Taboada M., Mitkov R., « A machine learning approach to negation and speculation detection », *Association for Information Science and Technology*, 2015.
- Farkas R., Vincze V., Móra G., Csirik J., Szarvas G., « The CoNLL-2010 shared task : learning to detect hedges and their scope in natural language text », *Fourteenth Conference on Computational Natural Language Learning*, p. 1-12, 2010.
- Ferson S., O'Rawe J., Antonenko A., Siegrist J., Mickley J., Luhmann C. C., Sentz K., Finkel A. M., « Natural language of uncertainty : numeric hedge words », *International Journal of Approximate Reasoning*, vol. 57, p. 19-39, 2015.
- Fuchs C., « L'incertitude interprétative dans l'activité de langage », *revue de l'IUF*, vol. 5, p. 41-57, 2008.
- Ganter V., Strube M., « Finding hedges by chasing weasels : Hedge detection using Wikipedia tags and shallow linguistic features », *ACL-IJCNLP*, 2009.
- Gaspar P., Carbonell J., Oliveira J. L., « On the parameter optimization of Support Vector Machines for binary classification », *J Integr Bioinform*, vol. 9, n° 3, p. 201, 2012.

- Georgescu M., « A Hedgehop over a Max-Margin Framework Using Hedge Cues », *Fourteenth Conference on Computational Natural Language Learning*, p. 26-31, 2010.
- Hamdan H., Sentiment Analysis in Social Media, Ph.d thesis, Université d'Aix-Marseille, 2015.
- Joachims T., « Learning to classify text using support vector machines : Methods, theory and algorithms », *Kluwer Academic Publishers*, vol. , p. 205, 2002.
- Jousselme A. L., Maupin P., Bosse E., « Uncertainty in a situation analysis perspective », *Sixth International Conference of Information Fusion*, p. 1207-1214, 2003.
- Konstantinova N., de Sousa S. C., Díaz N. P. C., López M. J. M., Taboada M., Mitkov R., « A review corpus annotated for negation, speculation and their scope », *LRE*, p. 3190-3195, 2012.
- Lavalley R., Clavel C., Bellot P., « Extraction probabiliste de chaînes de mots relatives à une opinion », *Traitement Automatique des Langues*, vol. 51, p. 101-130, 2010.
- Light M., Qiu X. Y., Srinivasan P., « The language of bioscience : Facts, speculations, and statements in between », *BioLink 2004 workshop on linking biological literature, ontologies and databases : tools for users*, p. 17-24, 2004.
- Pang B., Lee L., « A sentimental education : Sentiment analysis using subjectivity summarization based on minimum cuts », *The 42nd annual meeting on Association for Computational Linguistics*, vol. Association for Computational Linguistics, p. 271, 2004.
- Sebastiani F., « Machine learning in automated text categorization », *ACM computing surveys*, vol. 34, n° 1, p. 1-47, 2002.
- Smets P., « Imperfect information : Imprecision and uncertainty », *Uncertainty Management in Information Systems* p. 225-254, 1997.
- Szarvas G., Vincze V., Farkas R., Csirik J., « The BioScope corpus : annotation for negation, uncertainty and their scope in biomedical texts », *Workshop on Current Trends in Biomedical Natural Language Processing*, p. 38-45, 2008.
- Szarvas G., Vincze V., Farkas R., Móra G., Gurevych I., « Cross-genre and cross-domain detection of semantic uncertainty », *Computational Linguistics*, vol. 38, n° 2, p. 335-367, 2012.
- Tai K. S., Socher R., Manning C. D., « Improved semantic representations from tree-structured long short-term memory networks », *eprint arXiv :1503.00075*, 2015.
- Tang B., Wang X., Wang X., Yuan B., Fan S., « A Cascade Method for Detecting Hedges and their Scope in Natural Language Text », *Fourteenth Conference on Computational Natural Language Learning*, p. 13-17, 2010.
- Vincze V., « Uncertainty Detection in Natural Language Texts », *PhD, University of Szeged*, p. 141, 2014.
- Wu A. S., Do B. H., Kim J., Rubin D. L., « Evaluation of negation and Uncertainty detection and its impact on precision and recall in search », *Journal of Digital Imaging*, vol. 24, n° 2, p. 234-242, 2011.
- Yang Y., Pedersen J. O., « A comparative study on feature selection in text categorization », *ICML*, vol. 97, p. 412-420, 1997.
- Øvrelid L., Veldal E., Oepen S., « Syntactic scope resolution in uncertainty analysis », *23rd International Conference on Computational Linguistics*, vol. 10, p. 1379-1387, 2010.

---

# Structured Indexing Model for Cross-Language Information Retrieval

Chedi Bechikh Ali\* — Hatem Haddad\*\*

\* ISG, Université de Tunis

LISI laboratory, INSAT, Carthage University

\*\* Mevlana University, Konya, Turkey

{chedi.bechikh, haddad.hatem}@gmail.com

---

**ABSTRACT.** In recent digital library systems or World Wide Web environment, parallel corpora are used by many applications (Natural Language Processing, machine translation, terminology extraction, etc.). This paper presents a new cross-language information retrieval model based on the language modeling. The model avoids query and/or document translation or the use of external resources. It proposes a structured indexing schema of multilingual documents by combining a keywords model and a keyphrases model. Applied on parallel collections, a query, in one language, can retrieve documents in the same language as well as documents on other languages. Promising results are reported on the MuchMore parallel collection (German language and English language).

**RÉSUMÉ.** Dans les systèmes récents de bibliothèques numériques ou dans le contexte du Web, les corpus parallèles sont utilisés par de nombreuses applications (traitement du langage naturel, la traduction automatique, extraction de terminologie, etc.). Cet article présente un nouveau modèle de recherche d'information inter-langue basé sur le modèle de langue. Le modèle évite la traduction des requêtes et/ou des documents ainsi que l'utilisation des ressources externes. Il propose un schéma d'indexation structurée des documents multilingues en combinant un modèle de mots-clés et un modèle de phrase-clés. Appliquée sur une collection parallèle, une requête dans une langue, peut récupérer des documents dans la même langue ainsi que des documents dans d'autres langues. Appliqué à la collection parallèle MuchMore (en langue allemande et en langue anglaise), le modèle a montré des résultats prometteurs.

**KEYWORDS:** Cross language information retrieval, parallel collection, multilingual index, keyphrases, natural language processing.

**MOTS-CLÉS :** Recherche d'information inter-langue, collection parallèle, index multilingue, phrase-clés, traitement de la langue naturelle.

---

## 1. Introduction

Cross-language information retrieval (CLIR) allows an information seeker to apply a request in one language and to find information in a different language. Accordingly, multilingual information research models must overcome and eliminate language barriers by allowing an Information Retrieval System (IRS) to retrieve relevant documents expressed in languages other than the query language. For example, considering the German query<sup>1</sup>: “*Arthroskopische Behandlung bei Kreuzbandverletzungen*”<sup>2</sup>, the following English text<sup>3</sup> should be judged relevant by the CLIR system: “... *In all knee ligament procedures, arthroscopy is obligatory for diagnosing and conducting meniscus surgery ...*”.

Most of the techniques proposed to solve the problem of Cross-Language Retrieval center around a common idea: they attempt to translate the query from the user’s language to the language of the documents (Peters *et al.*, 2012). In most cases, the translation is done using a dictionary or a machine translation system. Any given word may have multiple possible translations, so significant effort has been devoted to disambiguating the resulting translations.

CLIR is facing the correct terminology equivalent (translation) selection challenge from one language to another. Indeed, the terms of the various languages almost never cover the same semantic field and the sense drift is unavoidable in a translation. To minimize the sense drift and the effort on the translation we propose to avoid the query or the document translation step.

Further, the CLIR systems are facing the challenge to be effective and to rank the relevant retrieved documents. To achieve this, CLIR systems must use an expressive documents (as well as queries) representation. Earlier works showed that the use of simple words as keywords is not always accurate enough to represent the documents contents due to the words ambiguity (Arampatzis *et al.*, 1998). Thus, we propose a documents representation and queries representation using keyphrases. By keyphrase, we refer to a list of phrases of two or more words such as many academic journals include in the beginning of an article (Bechikh-Ali *et al.*, 2015). Keyphrases can be extracted statistically, linguistically, or by combining the two approaches (Haddad and Bechikh-Ali, 2014). We note that the keyphrases extracted statistically to represent the document content may contain noise that may affect negatively the IRS performances. Considering the following example: “*In all knee ligament procedures, arthroscopy is obligatory for diagnosing and conducting meniscus surgery ...*”. The statistical methods extract keyphrases such as “in all”, “is obligatory”, “for diagnosing”, etc. These keyphrases can decrease the IRS performance. On the other hand, linguistic methods extract keyphrases such as: “knee ligament”, “meniscus surgery”, etc. We note that these keyphrases are more likely to represent the content or the topic of the sen-

---

1. German Query number 1 from the MuchMore collection (?)

2. Arthroscopic treatment of cruciate ligament injuries

3. English document DerChirurg/70681093 from the MuchMore collection

tence than those extracted by statistical methods (Bechikh-Ali *et al.*, 2015). For this reason, in addition to keywords, we propose to represent documents and queries by keyphrases.

The remainder of this paper is organized as follows. The related works are reviewed and discussed in Section 2. In Section 3, we introduce our proposed CLIR model based on a structured multilingual indexing. The experimental setup is detailed in Section 4. Evaluation results and discussion are given in Section 5 followed by a conclusion and future improvements to end the paper.

## 2. Related Work

Four types of approaches for CLIR are identified (Oard and Diekema, 1998) and can be divided into two categories (Kishida, 2005):

- Translation: in this category the following approaches can be used: query translation (QT) (translate the query representation to match the document representations), document translation (DT) (translate the document representations to match the query representation) and interlingual techniques (IT) (translate the document and the query representations into a third language or semantic space). IT approach uses a language independent representation for both queries and documents of a given parallel document collection (Chew and Abdelali, 2007). QT and DT use dictionary-based translation (i.e. machine-readable, ontologies, dictionaries) or machine translation (Zhou *et al.*, 2012). In this context, CLIR is an integration of words translation into word-based retrieval models (Hieber and Riezler, 2015).

- No translation: in this category, the cognate matching is used between languages having a close linguistic relationship (English, French, Spanish, ...) where unchanged words (such as proper nouns or technical terminology) can be expected to match successfully (Oard and Diekema, 1998; Gey, 2005).

QT has become the popular matching approach for CLIR due to the fact that it is less computationally costly to process the translation of a query than the translation of a large documents set. On the other hand, it presents two main disadvantages. The first disadvantage is the word translation ambiguity. The reason is that queries are often short and provide little context for disambiguation leading for many translation possibilities (Oard and Diekema, 1998). For example, "air flight" will be translated to French as "vol air". The word "vol" can refer to either "robbery" or to "flight". Furthermore, the dictionary-based translation approaches are limited by the availability of phrases and compound words for translation.

Various methods have been proposed to solve the problems of disambiguation, often relying on the document collection or integrating a translation step directly into the research model (Kraaij *et al.*, 2003). Other methods rely on external resources, such as Wikipedia (Nguyen *et al.*, 2008) or the Web (Hu *et al.*, 2008). Gao *et al.* in (Gao *et al.*, 2006) proposed syntactic translation models to address the homonymy problems. Translations candidates proposed by these models are then reclassified us-

ing the learned model to minimize translation error. Because of modeling translation and modeling retrieval separately, (Ture *et al.*, 2011) proposed to include a weighted translation into the query structure. Hiemstra *et al.* (Hiemstra and de Jong, 1999a) applied QT using one translation per source language query word. QT using all possible translations per source language query word. They concluded that using all possible translations for searching leads to better average precision retrieval performance than using one translation. In the same way, (Nguyen *et al.*, 2008) mapped queries to Wikipedia concepts and used corresponding translations of these concepts in the target language to create the final query.

The second disadvantage of the QT approach is, considering a N-lingual environment, the number of required systems to achieve CLIR reaches  $\frac{N(N-1)}{2}$  (Costa-jussà M, 2014). For this reason, most research in CLIR has not attempted to use many languages at a time. The reason is, that for each new language pair, a new translation algorithm must be used.

When using QT approach (and also in the case for DT strategy), once queries (and also the case for documents) are translated, a monolingual information retrieval system is then used. The quality of the IR step is strongly correlated to the quality of the translation step (Hieber and Riezler, 2015).

Other researchers have attempted to apply the language model (LM) to CLIR tasks. One of the advantages of the LM approach to CLIR tasks is to enable researchers to put translation probability linked to words pairs (a source word and a target word) in the ranking function (Kraaij *et al.*, 2003; Hiemstra and de Jong, 1999b). Kraaij *et al.* (Kraaij *et al.*, 2003) compared several models for CLIR on the CLEF-2000, 2001 and 2002 test collections. They showed that CLIR models based on mapping a query language model onto a document language model, or mapping a document language model onto a query language model, significantly outperformed CLIR models that do not use translation probabilities. They used probabilistic translation models trained on parallel texts mined from the Web. Hiemstra and de Jong (Hiemstra and de Jong, 1999b) combined the LM and translation probability from statistical dictionaries based on parallel corpora mined from the Web, instead of using one translation optimized for readability and well-formedness. They propose to improve recall by using many alternative translations. They evaluated the models on the French and Italian corpus of CLEF 2000. A performance enhancement, between 68% and 79% with respect to the monolingual baseline, was achieved.

Our main research question is to overcome the translation approaches limitations without using external resources. In parallel collections context, we propose a multilingual information model to represent the documents content (as well as queries content) in a multilingual context. It allows users to query a multilingual collection using one language and retrieve documents in a different language. The proposed structured indexing schema is combining a keywords model and a keyphrases model. For this reason, a mixture LM is used to allow the matching between keywords and between keyphrases. Our model could be useful in a wide variety of circumstances where a user is not sufficiently fluent in a document collection language to express a query in



that language. But on the other hand, the user is able to make use of the documents that are retrieved by an IRS; especially technical documents. Our Experimental evaluation is done on the MuchMore parallel collection including two languages: English and German. We compare the proposed model performances with QT performances and DT performances on the same collection presented in (Volk *et al.*, 2003).

### 3. Proposed CLIR Model

Our proposed CLIR model is based on the LM introduced in (Ponte and Croft, 1998). This choice is motivated by the model good performances while using few parameters. The basic idea behind this model is first to estimate a LM for each document. Then, given a query, the documents are ranked according to the likelihood that the query has been generated by the document as evidence (Zhai, 2008). Thus, for a query  $Q$ , the probability of a document  $D$  is:

$$P(Q|D) = \prod_{q_i \in Q} P(q_i|D) \quad \text{with } P(q_i|D) = \frac{tf(q_i)}{l_d} \quad [1]$$

where  $q_i$  is a query term,  $tf(q_i)$  is the term frequency of  $q_i$  in the document  $D$  and  $l_d$  is the document length.

To avoid the zero probability when a query word does not appear in a document, we use the Dirichlet smoothing (Zhai and Lafferty, 2001). It determines the collection LM for the word  $q_i$ :  $P(q_i|C)$ , where  $C$  is the document collection:

$$P_{Dir}(q_i|D) = \frac{P(q_i|D) + \mu P(q_i|C)}{|D_j| + \mu} \quad [2]$$

The value of  $\mu$ , determined empirically, is set to maximize a retrieval metric (mean average precision (MAP) for example) for a set of queries and a collection of documents.

#### 3.1. Parallel Text Collection

A parallel collection is a text paired with its translation into one or many languages (Resnik and Smith, 2003). It is composed of a set of document pairs in different languages that are mutual translations.

Let  $C$  be a parallel collection and  $L$  the set of languages used in  $C$  where  $n$  is the number of languages used in  $C$ . Thus,  $L = \{L_1, L_2, \dots, L_n\}$  where  $|L| = n$  with  $n \geq 2$ . The collection  $C$  is then composed by  $n$  sub-collections:  $C = \{C_{L_1}, C_{L_2}, \dots, C_{L_n}\}$  where  $L_l \in L$  and  $1 \leq l \leq n$ . In a parallel collection, each document is translated to *all* languages in  $L$ . The number of documents  $k$  in any sub-collection is the same:  $\forall$  a sub-collection  $C_{L_l} \in C$ ,  $|C_{L_l}| = k$  where  $k \geq 1$ .

### 3.2. Multilingual Document Indexing

A multilingual document in the collection  $C$  is the merging of all  $k$  paired documents. Given a multilingual document  $D_j \in C$ ,  $D_j = \{D_{j,L_1}, D_{j,L_2}, \dots, D_{j,L_n}\}$  where  $D_{j,L_l}$  is the sub-document of  $D_j$  corresponding to the language  $L_l$  and  $\|D_j\| = n$ . The document index is a logical view where a document in a collection is represented through a set of keywords. Specifically, given a text collection, a vocabulary  $V$  is created and used by the IRS to represent the documents content. The parallel collection vocabulary  $V$  is composed of two sub-vocabularies: the keywords vocabulary  $V_T$  and the keyphrases vocabulary  $V_K$ :  $V = V_T \cup V_K$ .

#### 3.2.1. Multilingual Document Indexing with Keywords

The keywords vocabulary  $V_T$  consists of  $n$  sub-vocabularies corresponding to each specific language. Thus, given any language  $L_l \in L$ , a sub-vocabulary  $V_{TL_l}$  is created and used to index any document  $D_{j,L_l}$ :  $V_T = \bigcup_{i=1}^n V_{TL_i}$ . Given  $I_{Tj}$  the keywords index of a multilingual document  $D_j \in C$ ,  $I_{Tj}$  is composed of  $n$  sub-indexes:  $I_{Tj} = \bigcup_{i=1}^n I_{Tj,L_i}$ . On the other hand, the size of a language vocabulary is different from language to another. Hence, the sizes of sub-indexes are different.

#### 3.2.2. Multilingual Document Indexing with Keyphrases

The keyphrases vocabulary  $V_K$  is composed by  $n$  sub-vocabularies corresponding to each specific language. Thus, given any language  $L_l \in L$ , a sub-vocabulary  $V_{KL_l}$  is created and used to index any document  $D_{j,L_l}$ :  $V_K = \bigcup_{i=1}^n V_{KL_i}$ . Given  $I_{Kj}$  the keyphrases index of a multilingual document  $D_j \in C$ ,  $I_{Kj}$  is composed by  $n$  sub-indexes:  $I_{Kj} = \bigcup_{i=1}^n I_{Kj,L_i}$ . As in the case of keywords, the size of a language keyphrases vocabulary is different from one language to another. Hence, the sizes of keyphrases sub-indexes are different.

### 3.3. Structured Multilingual Documents Indexing and Matching

Our proposed multilingual document model can be estimated using two models: keywords model ( $M_T$ ) and keyphrases model ( $M_K$ ). The document  $D_j$  is then represented by a structured index  $I_{D_j}$ . This document index is a sequence of  $w$  keywords and  $m$  keyphrases:

$$I_{D_j} = [[[t_{j,1}, \dots, t_{j,w}], [K_{j,1}, \dots, K_{j,m}]]_{L_1}, \dots, [[t_{j,1}, \dots, t_{j,w}], [K_{j,1}, \dots, K_{j,m}]]_{L_n}]$$

with  $t_{j,i} \in V_T$  and  $K_{j,i} \in V_K$ .

A query  $Q$  index is a structured index combining a sequence of  $r$  keywords and  $s$  keyphrases:

$$Q_q = [[t_{q,1}, t_{q,2}, \dots, t_{q,r}], [K_{q,1}, K_{q,2}, \dots, K_{q,s}]]_{L_l} \text{ with } t_{q,i} \in V_T \text{ and } K_{q,i} \in V_K$$

and the query is in the language  $L_l$ .

Although the proposed model is based on the assumption that the Keyphrases are important to capture the dependencies between words and thus find the relevant documents. It is also based on the assumption of independence between the words, effective to treat keywords. To consider the matching between keywords and the matching between keyphrases, we assume that the document model can be estimated using two models: the keyword model  $M_T$  and the keyphrases model  $M_K$ .

When combining several elements in a model, it is important to assign the appropriate weight to each part of the model. Indeed, we consider the matching between the keywords and the matching between the keyphrases as the probability that a query is generated for a given document and can be expressed as follows:

$$P(Q|D) = \alpha \prod_{i=1}^n P_{Dir}(t_{q,i}|D_T) + (1 - \alpha) \prod_{j=1}^m P_{Dir}(K_{q,j}|D_K) \quad [3]$$

$P_{Dir}(t_{q,i}|D)$  and  $P_{Dir}(K_{q,j}|D)$  are the corresponding probabilities for keywords and keyphrases.  $\alpha$  is a weighting parameter that controls the importance of the two different models,  $\alpha \in [0, 1]$ .

A query in the source language is matching the sub-index in the same language. After the matching process, sub-documents in different languages than the query language are returned as results. In the CLIR context, a query  $Q$  is expressed using only one language source  $L_s$ . The query index is then represented by  $I_Q$ . A CLIR system returns a set of documents  $C_Q \in C$  relevant to  $Q$  using a matching function  $(D_j, Q)$  where  $D_j \in C_Q$ . The matching function is used to calculate the score of the document  $D_j$  for the given query  $Q$ .

## 4. Experimental Setup

In this section, we described the parallel collection and the experimental setup we used.

### 4.1. MuchMore Parallel Collection

To evaluate our CLIR model, we used MuchMore parallel collection (?). The collection contains English documents and their direct German translations. These documents have identical contents but in two languages. Documents are from 41 medical journals, each constituting a homogeneous medical sub-domain (e.g. Neurology, Radiology, etc.). Each document consists of a title, an abstract and a set of keywords. For the CLIR propose, the collection contains 25 topics. Each topic is represented by two identical queries: one English query and one German query. Each query is provided with a set of relevant documents varying from 7 relevant documents to 104 relevant

**Table 1.** *Parallel Collection Statistics (MuchMore Collection)*

|                          | German Dataset | English Dataset |
|--------------------------|----------------|-----------------|
| #Total Documents         | 7808           | 7823            |
| # Number of terms        | 547696         | 639788          |
| # Number of unique terms | 80502          | 25304           |
| #Terms by document       | 70             | 81              |
| #Query                   | 25             | 25              |
| #Queries terms           | 117            | 132             |
| #Terms by query          | 4.68           | 5.28            |

documents. The number of English relevant documents for English is 500 while it's 959 for the German relevant documents. The reason is that two different teams participated in judging the relevance (Volk *et al.*, 2002). The collection statistics are given in Table 1.

#### 4.2. Keyphrases Extraction

Our approach to extract keyphrases is based on two phases:

1) We conducted a linguistic analysis with a tagger to generate a tagged collection. Each word is associated to a tag corresponding to the syntactic category of the word, for example: noun, adjective, preposition, etc. To tag the parallel collection, we used TreeTagger<sup>4</sup>. It is a tool for annotating text with part-of-speech and lemma information. The TreeTagger has been used to tag German and English languages.

2) Then, we used the tagged collection to extract a set of keyphrases by the identification of syntactic patterns (Bechikh-Ali *et al.*, 2015). Given  $S$  the collection lexicon, a pattern is a syntactic rule of the form:

$$X := Y_1 Y_2 Y_k \dots Y_{k+1} Y_n \text{ where } Y_i \in S \text{ and } X \text{ is a keyphrase.}$$

For examples, "*arthroscopic treatment*" is extracted using the syntactical rule "*Adjective Noun*". The keyphrase "*ligament injury*" is extracted using the syntactical rule "*Noun Noun*".

Table 2 presents statistics about extracted keyphrases from documents and queries in both English and German languages.

4. <http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/>

**Table 2.** *Keyphrases Extraction Statistics (in two languages)*

|                          | German Dataset | English Dataset |
|--------------------------|----------------|-----------------|
| #Keyphrases in documents | 162125         | 225571          |
| #Keyphrases by document  | 16.15          | 28.83           |
| #Keyphrases in queries   | 27             | 48              |
| #Keyphrases by query     | 1.08           | 1.92            |

### 4.3. Terrier IR System and Evaluation Measures

To implement our CLIR model, we used the Terrier IR system (Ounis *et al.*, 2005). The preprocessing includes lemmatisation with Treetagger and removing stop words. It implements various retrieval models. To investigate the effectiveness of the multilingual indexing, we implemented three different runs:

- $Run_1$  is implementing the  $M_T$  model indexing keywords only.
- $Run_2$  is implementing the  $M_T$  and the  $M_K$  models combining keywords and keyphrases but in the same documents/queries indexes. In this run, the index structure is not used.
- $Run_3$  implements our CLIR model introduced in section 3 where we use separate indexes one for keywords and another for keyphrases.

$Run_1$  and  $Run_2$  are evaluated using three models: Tf.Idf (Salton *et al.*, 1975), BM25 (Robertson and Walker, 1994) and LM (Ponte and Croft, 1998).  $Run_3$  is evaluated using a LM based on the matching function described in sub-section 3.3. The model BM25F (Robertson *et al.*, 2004) was used in this run instead of BM25 because it is able to evaluate a structured index considering the weighting and the matching of sub-indexes separately.

Results are evaluated using the the mean average precision (MAP) and precision measures at low recall considering only the k-top documents returned by the system: precision at 5 documents (P@5) and precision at 10 documents (P@10).

## 5. Evaluation Results and Discussion

Our CLIR system can perform multilingual indexing as well as monolingual indexing. To investigate the effectiveness of our model, we use the monolingual result as baseline. Table 3 shows the English and the German monolingual best results performances with the optimal parameter: the model Tf.Idf and BM25 use pivoted document length normalisation with a slope parameter  $b$  set by default to 0.20 for Tf.Idf and 0.75 for BM25. The LM use the parameter  $\mu$  and it is set to 4500. Tf.Idf model provided the best results for the English monolingual run. For the German monolingual run, BM25 model provided the best results.

**Table 3.** *English Monolingual and German Monolingual Results using 25 queries*

| Measures | English Monolingual |        |        | German Monolingual |               |        |
|----------|---------------------|--------|--------|--------------------|---------------|--------|
|          | Tf.Idf              | BM25   | LM     | Tf.Idf             | BM25          | LM     |
| P@5      | <b>0.5840</b>       | 0.5760 | 0.5360 | 0.5840             | <b>0.5920</b> | 0.5840 |
| P@10     | <b>0.4920</b>       | 0.4920 | 0.4360 | 0.5080             | <b>0.5080</b> | 0.4800 |
| MAP      | <b>0.3515</b>       | 0.3511 | 0.3204 | 0.2489             | <b>0.2530</b> | 0.2482 |

**Table 4.** *Querying English Collection using 25 German Queries*

| Measures | Run <sub>1</sub> |               |               | Run <sub>2</sub> |        |        | Run <sub>3</sub>           |                         |
|----------|------------------|---------------|---------------|------------------|--------|--------|----------------------------|-------------------------|
|          | Tf.Idf           | BM25          | LM            | Tf.Idf           | BM25   | LM     | BM25F<br>( $\alpha=0.99$ ) | LM<br>( $\alpha=0.98$ ) |
| P@5      | 0.5600           | <b>0.5680</b> | 0.5440        | 0.5600           | 0.5680 | 0.5440 | 0.4560                     | 0.4960                  |
| P@10     | <b>0.4920</b>    | 0.4880        | 0.4640        | 0.4640           | 0.4600 | 0.4480 | 0.4080                     | 0.4120                  |
| MAP      | 0.2096           | 0.2132        | <b>0.2139</b> | 0.1995           | 0.2033 | 0.2069 | 0.1784                     | 0.1845                  |

### 5.1. German-English CLIR

Performance results of querying the English collection with German queries are presented in Table 4. For the Run<sub>3</sub>, we tuned the parameter  $\alpha$  by the use of a grid search. We selected the  $\alpha$  values that maximizes the MAP. Run<sub>1</sub> yields the best results compared to Run<sub>2</sub> and Run<sub>3</sub>. One reason for this is the small queries size. Indeed, the average extracted keyphrases number per query is 1.08. In addition, four queries include 0 keyphrases. Comparing Run<sub>2</sub> and Run<sub>3</sub> results, we can observe that using a unified index is performing better than using the structured index. Comparing Run<sub>3</sub> BM25F results and Run<sub>3</sub> LM results, we observe that using the structured index in the case of LM model is performing better than the case of BM25F model.

### 5.2. English-German CLIR

Table 5 shows the performances of querying the German collection using English queries. The  $\alpha$  values allowing the best MAP performances are used after tuning  $\alpha$  by the use of a grid search. The CLIR results performances outperform the monolingual results performances. Indeed, querying a German collection with an English query gives better results than using a German query.

## Querying German Collection using 25 English Queries

**Table 5.** *Querying German Collection using 25 English Queries*

| Measures | Run <sub>1</sub> |        |        | Run <sub>2</sub> |               |        | Run <sub>3</sub>           |                        |
|----------|------------------|--------|--------|------------------|---------------|--------|----------------------------|------------------------|
|          | Tf.Idf           | BM25   | LM     | Tf.Idf           | BM25          | LM     | BM25F<br>( $\alpha=0.99$ ) | LM<br>( $\alpha=0.5$ ) |
| P@5      | 0.7200           | 0.7120 | 0.6320 | <b>0.7440</b>    | 0.7280        | 0.6720 | 0.6240                     | 0.5760                 |
| P@10     | 0.6080           | 0.6080 | 0.5480 | 0.6080           | <b>0.6160</b> | 0.5880 | 0.5480                     | 0.5440                 |
| MAP      | 0.3126           | 0.3147 | 0.2899 | 0.3204           | <b>0.3232</b> | 0.3109 | 0.2854                     | 0.2735                 |

Combining keywords with keyphrases performs better than using only keywords. Indeed, the best results are given by *Run<sub>2</sub>* for all evaluation measures. The BM25 achieves better performances than Tf.Idf and LM models. Comparing BM25 results of *Run<sub>2</sub>* with the German monolingual results, we observe an improvement of MAP and P@10 measure with respectively 27.74% and 21.25%.

### 5.3. Discussion

To analyse the monolingual results, we processed a deep investigation on the Muchmore collection. We observed that not all the documents are translated. For example, the English document number 1449 and the German document number 1434 have no translation.

In (Volk *et al.*, 2003), QT approach and IT (semantic information combination from different resources) approach results applied to the MuchMore collection are presented. Authors limited their evaluation to the use of German queries to retrieve English documents. Comparing these results to our model results presented in Table 4, our proposed model outperformed both approaches. Indeed, our model performances exceed the QT approach performances for *Run<sub>1</sub>*, *Run<sub>2</sub>* and *Run<sub>3</sub>* by respectively 55%, 50% and 33.6% for MAP and 67%, 59% and 41.1% for P@10. *Run<sub>1</sub>*, *Run<sub>2</sub>* and *Run<sub>3</sub>*. MAP performances exceed by respectively 20.6%, 16.6% and 4% the IT approach MAP performances and by 80.9% and 70.6% the IT approach P@10 performances. Nonetheless, we can not draw detailed conclusions because of the collection characteristics. Indeed, given the short queries size and the medical topic of the collection, few keyphrases are extracted from queries and some technical keyphrases are not extracted because they are not correctly identified by the POS tagger. That is, more experiments are needed to conclude about the efficiency of our model.

## 6. Conclusion

In this paper, we proposed a CLIR model avoiding translation and external resources. Our model uses a structured index to represent multilingual documents and queries. We evaluated our model on the MuchMore parallel collection (English language and German language). In the context of a parallel collection, using an English query to retrieve German documents is performing better than using a German query. We compared the proposed model performances with QT approach performances and DT approach performances. We conclude that our proposed model is performing better than both approaches. In the future, we plan to investigate more in-deep weighting schema of our model and apply our model to a larger parallel collection involving more than two languages with longer queries.

## 7. References

- Aramatzis A., Tsoris T., Koster C. H. A., van der Weide T. P., "Phrase-Based Information Retrieval", *Inf. Process. Manage.*, vol. 34, n° 6, p. 693-707, 1998.
- Bechikh-Ali C., Wang R., Haddad H., "A Two-Level Keyphrase Extraction Approach", *The 16th International Conference on Intelligent Text Processing and Computational Linguistics*, Cairo, Egypt, p. 390-401, 2015.
- Chew P. A., Abdelali A., "Benefits of the 'Massively Parallel Rosetta Stone': Cross-Language Information Retrieval with over 30 Languages", *the 45th Annual Meeting of the Association for Computational Linguistics*, Prague, Czech Republic, p. 872-879, 2007.
- Costa-jussà M. Banchs R. E., "Is there Hope for Interlingua methods? A CLIR comparison experiment between Interlingua and Query Translation", *Research in Computing Science*, vol. 74, p. 81-87, 2014.
- Gao J., Nie J., Zhou M., "Statistical query translation models for cross-language information retrieval", *ACM Trans. Asian Lang. Inf. Process.*, vol. 5, n° 4, p. 323-359, 2006.
- Gey F. C., "How Similar are Chinese and Japanese for Cross-Language Information Retrieval?", *Proceedings of the Fifth NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering and Cross-Lingual Information Access*, NTCIR-5, National Center of Sciences, Tokyo, Japan, December 6-9, 2005, 2005.
- Haddad H., Bechikh-Ali C., "Performance of Turkish Information Retrieval: Evaluating the Impact of Linguistic Parameters and Compound Nouns", *The 15th International Conference on Intelligent Text Processing and Computational Linguistics*, Kathmandu, Nepal, p. 381-391, 2014.
- Hieber F., Riezler S., "Bag-of-Words Forced Decoding for Cross-Lingual Information Retrieval", *The Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Colorado, USA, p. 1172-1182, 2015.
- Hiemstra D., de Jong F., "Cross-language information retrieval in Twenty-One: Using one, some or all possible translations?.", *In the 14th Twente Workshop on Language Technology*, Twente, Netherlands, p. 19-25, 1999a.



- Hiemstra D., de Jong F., "Disambiguation Strategies for Cross-Language Information Retrieval", *The Third European Conference on Research and Advanced Technology for Digital Libraries, London, UK, ECDL '99*, p. 274-293, 1999b.
- Hu R., Chen W., Bai P., Lu Y., Chen Z., Yang Q., "Web Query Translation via Web Log Mining", *The 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Singapore*, p. 749-750, 2008.
- Kishida K., "Technical issues of cross-language information retrieval: a review", *Inf. Process. Manage.*, vol. 41, n° 3, p. 433-455, 2005.
- Kraaij W., Nie J., Simard M., "Embedding Web-Based Statistical Translation Models in Cross-Language Information Retrieval", *Computational Linguistics*, vol. 29, n° 3, p. 381-419, 2003.
- Nguyen D., Overwijk A., Hauff C., Trieschnigg D., Hiemstra D., de Jong F., "WikiTranslate: Query Translation for Cross-Lingual Information Retrieval Using Only Wikipedia", *The 9th Workshop of the Cross-Language Evaluation Forum, Evaluating Systems for Multilingual and Multimodal Information Access, Aarhus, Denmark*, p. 58-65, 2008.
- Oard D. W., Diekema A. R., "Cross-language information retrieval", *Annual review of information science and technology*, vol. 33, p. 223-256, 1998.
- Ounis I., Amati G., Plachouras V., He B., Macdonald C., Johnson D., "Terrier Information Retrieval Platform", *The 27th European Conference on Information Retrieval Research, Santiago de Compostela, Spain*, p. 517-519, 2005.
- Peters C., Braschler M., Clough P. D., *Multilingual Information Retrieval - From Research To Practice*, Springer, 2012.
- Ponte J. M., Croft W. B., "A Language Modeling Approach to Information Retrieval", *The 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Melbourne, Australia*, p. 275-281, 1998.
- Resnik P., Smith N. A., "The Web as a Parallel Corpus", *Computational Linguistics*, vol. 29, n° 3, p. 349-380, 2003.
- Robertson S. E., Walker S., "Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval", *The 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Dublin, Ireland*, p. 232-241, 1994.
- Robertson S. E., Zaragoza H., Taylor M. J., "Simple BM25 extension to multiple weighted fields", *The thirteenth ACM international conference on Information and knowledge management, Washington, DC, USA*, p. 42-49, 2004.
- Salton G., Yang C. S., Yu C. T., "A theory of term importance in automatic text analysis", *Journal of the American Society for Information Science*, vol. 26, n° 1, p. 33-44, 1975.
- Ture F., Elsayed T., Lin J., "No Free Lunch: Brute Force vs. Locality-sensitive Hashing for Cross-lingual Pairwise Similarity", *The 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, Beijing, China*, p. 943-952, 2011.
- Volk M., Ripplinger B., Vintar S., Buitelaar P., Raileanu D., Sacaleanu B., "Semantic annotation for concept-based cross-language medical information retrieval", *I. J. Medical Informatics*, vol. 67, n° 1-3, p. 97-112, 2002.
- Volk M., Vintar S., Buitelaar P., "Ontologies in Cross-Language Information Retrieval", *2nd Conference on Professional Knowledge Management, Luzern, Switzerland*, p. 43-50, 2003.

- Zhai C., “Statistical Language Models for Information Retrieval: A Critical Review”, *Foundations and Trends in Information Retrieval*, vol. 2, n<sup>o</sup> 3, p. 137-213, 2008.
- Zhai C., Lafferty J. D., “A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval”, *The 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Louisiana, USA*, p. 334-342, 2001.
- Zhou D., Truran M., Brailsford T. J., Wade V., Ashman H., “Translation techniques in cross-language information retrieval”, *ACM Comput. Surv.*, vol. 45, n<sup>o</sup> 1, p. 1, 2012.

---

# Génération d'une ontologie dans le domaine des ressources humaines

Rémy Kessler\* — Guy Lapalme\* — Eric Tondo\*\*

\* RALI - Département d'informatique et de recherche opérationnelle  
Université de Montréal  
C.P. 6128, Succ Centre-Ville, Montréal, Québec, Canada H3C 3J7  
{kessler,lapalme}@iro.umontreal.ca

\*\* LittleBigJob  
204 rue du St-Sacrement, Montréal, Québec, Canada H2Y 1W8  
eric.tondo@littlebigjob.com

---

**RÉSUMÉ.** Notre époque est de plus en plus influencée par la prééminence des données intelligentes et du web sémantique. Les processus de recrutement n'en sont pas toujours facilités en particulier en matière de recherche de profils et de talents. La plupart des systèmes d'appariement entre une offre d'emploi et un profil s'appuient sur une ou plusieurs ressources linguistiques, mais se heurtent à la difficulté de développer et à entretenir des ressources spécifiques à chaque domaine. Nous proposons une approche de construction semi-automatique d'une ontologie s'appuyant sur des millions de profils issus de plusieurs réseaux sociaux et sur des dizaines de milliers d'offres d'emploi collectées sur internet. Une évaluation manuelle a été réalisée par un expert du domaine et a montré des résultats de très bonne qualité.

**ABSTRACT.** The current era is increasingly influenced by the prominence of smart data and the semantic web. This does not make recruitment processes easier, especially when seeking candidates and skills. Most systems matching job offers to profiles build on one or more language resources. Developing and maintaining specific resources in each field is difficult. From millions of user profiles gathered on multiple social networks and tens of thousands of job offers collected on the internet, we describe an approach to generating a semi-automatic ontology. A manual evaluation of the results has shown excellent results.

**MOTS-CLÉS :** Ontologie, réseaux sociaux, ressources humaines, application industrielle.

**KEYWORDS:** Ontology, social networks, human resources, industrial application.

---

## 1. Introduction

Durant la dernière décennie, nous avons assisté à un développement rapide du web et des réseaux sociaux qui a modifié la dynamique de la recherche d'emploi. Les informations professionnelles publiées par les utilisateurs dans leurs profils (formations, antécédents de travail, résumé de carrière, liens sociaux, etc. ) peuvent être exploitées par les recruteurs pour identifier de nouveaux candidats ou pour obtenir des informations complémentaires à leur propos.

D'après une étude de RegionsJob<sup>1</sup> (2011), « 43% des recruteurs avouent recourir à des recherches de type nom/prénom sur les candidats qui postulent chez eux et 8% des recruteurs interrogés déclarent avoir écarté un candidat à cause de traces jugées négatives trouvées en ligne. » La plupart de ces recherches étant effectuées de façon rapide et manuelle, les éléments recueillis sur un individu à un premier niveau de recherche sont peu structurés, disparates, incomplets, redondants, parfois obsolètes et peuvent être biaisés, voire trompeurs (ex. : à cause des homonymes).

Dans le cadre du projet Butterfly Predictive Project<sup>2</sup>, nous développons une plateforme pour améliorer de manière importante la mise en correspondance entre des candidats et des offres d'emploi en étudiant comment les développements récents et à venir du web peuvent aider résoudre ces problématiques. Nous faisons l'hypothèse que l'acquisition et l'exploitation de cette trace laissée par les individus, diffuse, plus ou moins accessible et peu structurée sont une voie prometteuse pour l'expression du positionnement professionnel des candidats, pour la caractérisation des emplois et pour leur mise en correspondance. En complément du processus habituel de sélection d'un sous-ensemble des candidats actifs, c'est-à-dire ayant posé leur candidature, nous proposons d'identifier des candidats passifs, qui ne sont pas en recherche d'emploi, mais qui pourraient être intéressés par de nouvelles opportunités, en parcourant les différents médias sociaux afin de récolter les profils les plus en adéquation avec une offre d'emploi. Une fois cette collecte terminée, une proposition d'opportunité serait transmise à ces candidats passifs qui pourraient alors décider si la proposition les intéresse suffisamment pour être intégrés à la liste de candidats potentiels pour le poste. Même si de grands efforts ont été déployés ces dernières années afin de développer des ressources linguistiques afin d'aider les systèmes d'appariement de candidatures et d'offres d'emploi, ces derniers se heurtent à la difficulté de créer des ontologies/taxonomies spécifiques à chaque domaine et particulièrement à leur entretien et mise à jour.

En effet, les métiers d'aujourd'hui ne sont pas forcément les mêmes que ceux d'hier, et ne seront peut-être pas identiques aux métiers de demain. L'évolution de notre société entraîne l'apparition de nouveaux métiers et de nouvelles compétences ainsi que la disparition d'autres. Même si chaque profil est différent dans le détail, il

---

1. [http://entreprise.regionsjob.com/enquetes/reseaux\\_sociaux/resultats\\_enquete\\_2.pdf](http://entreprise.regionsjob.com/enquetes/reseaux_sociaux/resultats_enquete_2.pdf)

2. <http://rali.iro.umontreal.ca/rali/?q=fr/butterfly-predictive-project>

semble cependant envisageable qu'un regroupement d'informations issues des mêmes métiers fasse émerger des relations communes permettant de construire un ensemble structuré des termes et concepts représentant chaque domaine. La génération de ces ressources (tout en restant automatisée) permettra de créer une représentation de la connaissance de chaque domaine qui pourra être exploitée par la suite dans le cadre de l'appariement (p. ex. évaluer si un candidat a toutes les compétences pour un métier), de la génération de texte (p. ex. suggérer à un candidat comment mettre en avant ses compétences les plus en adéquation pour un poste).

## 2. Travaux connexes

De nos jours, des milliers de candidats mettent en ligne leur profil, et les entreprises ou les établissements publient une quantité importante de postes recherchés. Analyser automatiquement cette quantité d'informations est une tâche difficile à accomplir. De grands efforts ont cependant été déployés ces dernières années afin de constituer des ressources linguistiques afin d'améliorer les systèmes d'appariement.

La première étude a été proposée par (Lau et Sure, 2002) : ils décrivent une méthodologie afin de développer une ontologie (en se basant sur une étude de cas de la société Swiss Life) centrée sur le domaine des technologies de l'information (TI). Ils précisent que celle-ci a été construite manuellement même si des approches semi-automatiques avaient été tentées, mais les résultats ne permettaient pas d'obtenir une représentation claire et structurée des compétences.

Les premiers travaux dans l'appariement de candidature et d'offres d'emploi à l'aide d'une ontologie ont été proposés par (Colucci *et al.*, 2003 ; Colucci *et al.*, 2007). Leur système, IMPAKT<sup>3</sup> (Colucci *et al.*, 2013) combine une approche sémantique avec une ontologie et repose sur des méthodes de formalisation des raisonnements et des connaissances. Le système propose un appariement en effectuant des correspondances partielles ou complètes des compétences entre les offres d'emplois et les candidatures. La spécificité des informations contenues dans les documents d'une candidature et d'une offre d'emploi a conduit au développement d'approches sémantiques pour construire des ressources linguistiques.

(Desmontils *et al.*, 2002 ; Trichet *et al.*, 2004) proposent une méthode d'indexation sémantique. La méthode consiste à exploiter les caractéristiques dispositionnelles du document afin d'identifier chacune des parties et l'indexer en conséquence. Ils concluent les travaux en proposant une instanciation d'ontologie en partant des données récoltées et en s'appuyant sur des ressources externes (base ROME<sup>4</sup> et CI-GREF<sup>5</sup>). Même si l'approche semble intéressante, l'absence de résultat ne permet pas d'évaluer l'apport de cette indexation particulière. (Mochol et Simperl, 2006) décrivent l'importance d'une ontologie commune (HR ontology) ainsi qu'un guide pour

---

3. Information Management and Processing with the Aid of Knowledge-based Technologies

4. Répertoire Organisationnel des Métiers et Emplois

5. Club Informatique des Grandes Entreprises Françaises

mettre en place ce type d'application tandis que (Trichet *et al.*, 2004) décrivent différentes approches permettant la gestion des compétences à l'aide d'ontologies dans le cadre du e-recrutement.

Dans le cadre du projet Prolix, (Trog *et al.*, 2008) décrivent une ontologie de ressources humaines basée sur le cas de British Telecom. Ils proposent une architecture en plusieurs niveaux en fonction des compétences, des interactions et du contexte. Toujours dans l'idée de créer des ressources linguistiques (Gómez-Pérez *et al.*, 2007) proposent une annotation sémantique des documents (offre d'emploi et CV) afin de construire différentes ontologies. Développées en anglais et disponibles en ligne<sup>6</sup>, ces ontologies décrivent des compétences et formations spécifiques au domaine des TI tandis que d'autres sont plus généralistes comme les classes de métiers, permis de conduire ou encore les secteurs d'activités.

(Roche et Kodratoff, 2006) décrivent une étude d'extraction de terminologie spécifique sur un corpus de CV. Leur approche permet d'extraire un certain nombre de collocations contenues dans les CV sur la base de patrons (tels que Nom-Nom, Adjectif-Nom, Nom-préposition-Nom, etc.) et de les classer en fonction de leur pertinence en vue de la construction d'une ontologie spécialisée. Partant du constat que les médias sociaux deviennent une source incontournable pour les recruteurs dans leur recherche de candidats, (Tétreault *et al.*, 2011) décrivent la maquette d'une plateforme de recrutement intégrant les technologies du web sémantique ainsi que l'élaboration d'une ontologie à l'aide d'OWL dédiée au domaine TI. L'approche décrit les avantages d'une application de ce type ainsi que différents scénarios de recrutement.

Plus récemment, (le Vrang *et al.*, 2014) présente l'ontologie ESCO<sup>7</sup>, un projet européen multilingue permettant de classer compétences, métiers et certifications afin de créer une harmonisation européenne en matière de recrutement. Cependant, même si le modèle est d'excellente qualité, il n'en est actuellement qu'à la version 0.1, ce qui limite les domaines et métiers couverts. La version actuelle regroupe 4 761 métiers et 5 096 compétences en 24 langues différentes pour environ 250 000 termes différents (21 000 termes en anglais et 18 000 en français). Chaque métier ou compétence est défini dans chaque langue par un *label préféré* ainsi que par un ou plusieurs *labels alternatifs* pouvant être un synonyme, une féminisation, une forme abrégée ou une variation orthographique. Chaque métier est en outre relié aux compétences nécessaires à sa pratique et de la même façon chaque compétence est reliée aux différents métiers.

Ces travaux montrent que différentes approches ont été envisagées, que ce soit avec des méthodes statistiques, sémantiques ou encore à l'aide de ressources linguistiques complémentaires. Les réseaux sociaux et plus généralement l'information récoltée sur internet sur les candidats peuvent être une source pertinente afin de faire émerger des compétences et des connaissances communes pour construire un ensemble structuré de termes et de concepts.

6. <http://mayor2.dia.fi.upm.es/oeg-upm/index.php/en/ontologies/99-hrontology>

7. European Skills Competences and Occupations <https://ec.europa.eu/esco/>

### 3. Données et connaissances

Le projet BPP s'appuie sur une modélisation en termes de 44 secteurs d'activités (univers), chacun regroupant des familles de métiers assez proches et un ou plusieurs univers connexes. Les univers connexes au secteur considéré sont ceux pour lesquels une transition est envisageable pour un profil envisageant un changement de secteur. Par exemple, un profil issu de l'univers *Conseil et service informatiques, édition de logiciels* pourra plus facilement envisager une transition dans les domaines *Télécoms, Hébergement, Internet* ou *Ingénierie - R&D* que dans celui de *la cosmétique*. Nous avons combiné ces connaissances avec l'ontologie issue du modèle ESCO, ainsi qu'un dictionnaire des métiers issus de la Classification nationale des professions<sup>8</sup> (CNP) et du Répertoire opérationnel des métiers et des emplois (ROME)<sup>9</sup>. La CNP est la référence reconnue des professions au Canada tandis que le ROME est la référence en France. La CNP répartit plus de 40 000 appellations d'emplois (anglais/français) en 500 profils de groupes professionnels tandis que le ROME comprend 10 000 appellations différentes réparties en 531 groupes.

Par ailleurs, plus de 10 millions de profils issus de plusieurs réseaux sociaux (*LinkedIn, Viadeo, Indeed* et d'autres) ont été récoltés à l'aide d'un processus de collecte automatique de sites internet. Ces données issues de profils publics professionnels ont été préalablement anonymisées puis agrégées dans un format uniforme. L'origine géographique étant le Canada ou la France, les profils sont soit en français soit en anglais ou encore bilingues. Chaque profil regroupe différentes informations sur le parcours du candidat tels que ses diplômes et formations ainsi que la date d'obtention de chacun d'eux. De la même façon, une section du profil contient ses expériences. Chacune contenant plusieurs éléments tels que les dates de début et de fin, le nom de la société employeur (avec éventuellement une URL vers sa page internet), la fonction occupée par le candidat au cours de cette expérience ainsi que le lieu et un éventuel descriptif de sa mission au sein de cette société. Un résumé des expériences sous forme de texte est par ailleurs présent ainsi qu'une courte description sous la forme d'une phrase d'accroche permettant au candidat de se décrire en quelques mots. D'autres informations sont récoltées ou éventuellement calculées telles que l'expérience totale du candidat, ses langues maîtrisées, ses loisirs, le nombre de relations ou encore les compétences acquises au cours de son parcours professionnel. Chaque profil regroupe une cinquantaine de champs, mais il existe cependant un nombre important de profils ne contenant pas ou peu d'informations comme le montre le tableau 1 qui présente quelques statistiques descriptives de cette collection. Dans le cadre de cette application, nous nous concentrons sur les compétences issues de ces profils, environ 2,3 millions.

En complément de ces données, 100 000 offres d'emplois ont été collectées sur internet, 60 000 en anglais et 40 000 en français. Ces offres d'emplois couvrent un grand nombre de métiers et sont issues, elles aussi, du Canada ou de la France. Chaque offre

8. <http://www5.hrsdc.gc.ca/noc/>

9. <http://www.pole-emploi.fr/candidat/le-code-rome-et-les-fiches-metiers-@/article.jspz?id=60702>

|                                            | Canada    | France    |
|--------------------------------------------|-----------|-----------|
| Nombre de profils                          | 2 658 467 | 7 484 311 |
| Moyenne du nombre d'expériences            | 3.2       | 2.3       |
| Moyenne du nombre de formations            | 1.39      | 1.06      |
| Moyenne du nombre de compétences           | 0.17      | 0.12      |
| <i>statistiques textuelles des profils</i> |           |           |
| Profil vide                                | 9.61%     | 16.34%    |
| Moins de 100 caractères                    | 26.95%    | 40.53%    |
| Moins de 300 caractères                    | 16.18%    | 12.70%    |
| Moins de 500 caractères                    | 6.43%     | 4.66%     |
| Plus de 500 caractères                     | 40.83%    | 25.77%    |

**Tableau 1.** *statistiques de la collection de profils de réseaux sociaux*

d'emploi contient un titre, une description contenant le détail de l'offre d'emploi, la date de mise en ligne, le lieu de la mission proposée ainsi que le nom de la compagnie qui recrute.

Les premières observations ont montré que, bien qu'extrêmement bruitées, ces données peuvent constituer une source d'informations intéressantes afin de constituer une ontologie de façon semi-automatique.

#### 4. Méthodologie

La plupart des systèmes d'appariement entre une offre d'emploi et un profil présentés dans la section 2 s'appuient sur une ou plusieurs ressources linguistiques, de type ontologie ou taxonomie, mais se heurtent cependant à la difficulté de développer des ressources spécifiques à chaque domaine, ainsi qu'à leur entretien. Afin de pallier ce problème, nous souhaitons développer un système permettant de générer une ontologie de façon semi-automatique en s'appuyant sur les données collectées sur internet et présentées dans la section précédente. Nous faisons l'hypothèse que l'exploitation de ces informations est une voie prometteuse pour constituer un référentiel commun, une représentation de chaque domaine avec des liens logiques.

Dans un premier temps, nous avons effectué une comparaison des compétences issues des réseaux sociaux et les compétences contenues dans l'ontologie ESCO. Les résultats ont montré de nombreuses correspondances exactes ou partielles (plus d'un million) et variées. Par ailleurs les compétences les plus fréquentes dans les données issues des réseaux sociaux se retrouvent bien dans l'ontologie ESCO. Cette approche a cependant montré certaines limites. Même si le modèle ESCO est d'excellente qualité, les domaines et métiers couverts par l'ontologie restent limités (par exemple, on ne retrouve pas *business analyst* ou *account manager*, pour les métiers, ni *marketing strategy*, *financial modeling*, *food cost management* pour les compétences).



Par ailleurs, il est impossible de discerner dans les profils des candidats les compétences issues d'une expérience plutôt que d'une autre : par exemple, le profil d'un chef de projet sénior en technologies de l'information cumulera les compétences en tant que développeur issues de ses premières expériences avec celles par la suite de chef de projet. Il est donc difficile de se limiter à cette source comme identification des compétences requises pour un emploi.

Afin de pallier ces problèmes, nous avons décidé d'utiliser comme source de documents la collection d'offres d'emplois. En effet, celles-ci définissent les expériences et qualifications recherchées pour un métier particulier. Comme nous disposons d'un nombre important d'offres d'emplois (100 000), nous croyons être en mesure de faire émerger les compétences et connaissances communes pour construire un ensemble structuré des termes et concepts représentant chaque métier. L'ensemble des offres d'emplois étant collecté sur internet, le modèle s'enrichira à terme de façon semi-automatique avec l'apparition et disparition de métiers dans les offres d'emplois.

## 5. Vue d'ensemble du système AGOHRA

La figure 1 présente une vue d'ensemble du système AGOHRA<sup>10</sup>. Au cours d'une première étape ①, on effectue une normalisation des offres d'emploi. Le module suivant ② utilise les titres des offres d'emplois afin de détecter les métiers avant d'y associer l'univers concerné. L'étape suivante ③ consiste à utiliser l'ensemble du vocabulaire récolté afin de faire émerger les compétences. Le dernier module ④ transforme ensuite les informations ordonnées et structurées en une ontologie au format RDF.

### 5.1. Normalisation des offres d'emploi

Au cours d'une première étape, nous effectuons une séparation des offres en français et en anglais ainsi qu'un filtrage des quelques offres dans les autres langues (environ 150 offres sont en espagnol, japonais, italien, etc.). L'observation de la collection montrant un grand nombre de doublons (offres similaires avec des dates différentes, offres similaires pour des lieux différents, etc.); nous effectuons un dédoublement des offres d'emplois afin d'éviter d'augmenter artificiellement les fréquences des termes. Environ 6000 offres d'emplois sont ainsi écartées de la collection. En parallèle avec cette étape, une normalisation des titres des offres d'emplois est effectuée afin de détecter les métiers en utilisant différents patrons afin d'améliorer la qualité du processus. On effectue ainsi une suppression des accents, des caractères particuliers tels que « - », « / », « ( ) » ou encore des expressions couramment utilisées dans les titres d'offres d'emplois tels que *full-time*, *temporary*, ou encore *entry level*. Différents processus linguistiques sont utilisés afin de réduire le bruit dans le modèle : les expressions courantes (par exemple, *c'est-à-dire*, *chacun de*, etc.), les chiffres et

10. Automatic Generation of an Ontology for Human Resource Applications

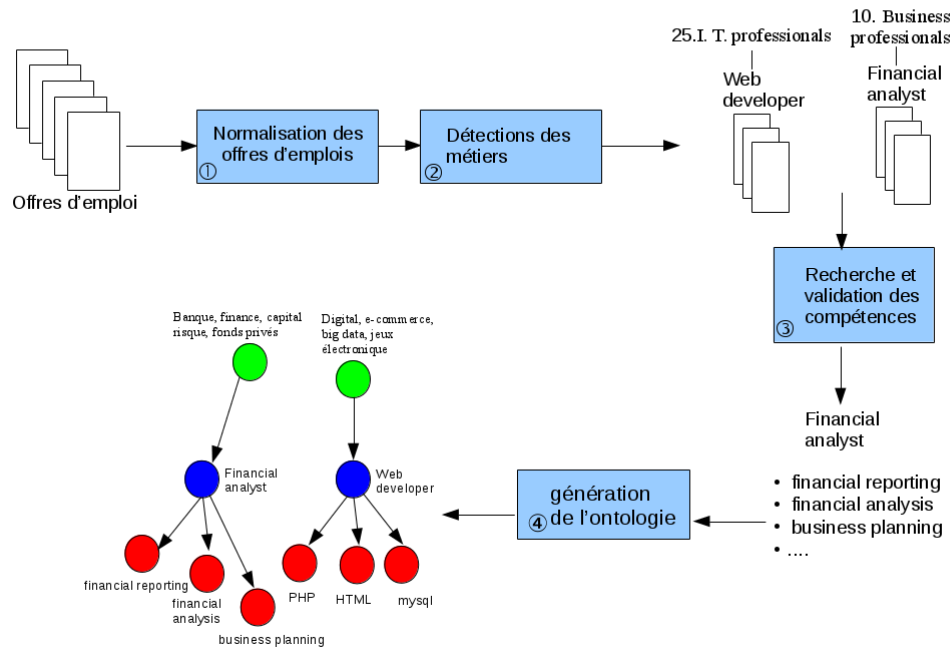


**Figure 1.** *Vue d'ensemble du système AGOHRA.*

nombres (numériques et/ou textuelles), les symboles spéciaux ainsi que les termes contenus dans un anti-dictionnaire spécifique et adapté à notre problème.

## 5.2. Détection des métiers

Nous utilisons au cours de cette étape les titres des offres d'emploi normalisées (voir 5.1) afin de construire une liste de métiers. Diverses règles sont ainsi appliquées afin d'extraire les métiers contenus dans ces titres. À l'aide d'un seuil de fréquence minimum déterminé empiriquement, nous ne conservons que les métiers avec une fréquence importante. Une première version du système permettait de générer les résultats pour l'ensemble des métiers, mais nous avons décidé de privilégier les métiers ciblés par notre partenaire industriel afin de filtrer les métiers avec peu de qualifications requises, comme *gardienne de chat*, *nounous*, etc. Nous filtrons ainsi une grande quantité d'offres d'emploi pour n'en conserver qu'environ 9 000 (6 000 en anglais et 3 000 en français).

### 5.3. Recherche et validation des compétences

L'objectif de cette étape (③) est d'utiliser le vocabulaire issu des offres d'emplois afin de faire émerger les compétences associées à chacun des métiers. Nous effectuons pour cela une agrégation du vocabulaire en le regroupant par métier. L'extraction d'information dans une offre d'emploi n'est pas une tâche triviale comme le souligne (Loth *et al.*, 2010). (Kessler *et al.*, 2008) montre qu'en raison d'une grande variété dans les paramètres (texte libre, tailles différentes, découpage incertain, délimiteurs variés), le découpage d'offres d'emploi en blocs d'information est une tâche délicate, mais que celles-ci suivent cependant un ordre conventionnel dans leur apparition. Afin de diminuer la taille du vocabulaire considéré, nous recherchons différents motifs séparateurs et fréquents dans une offre d'emploi, tels qu'*exigences*, *qualifications*, *responsabilités*, etc. Ces motifs, bien que pas toujours présents, permettent ainsi de réduire considérablement le vocabulaire en ne prenant en compte que la partie de l'annonce suivant le motif.

L'observation des compétences de la collection de profils de réseaux sociaux décrite en section 3 montre que plus de 80% des compétences se présentent sous la forme de n-grammes de mots (par exemple *financial modeling*, *php development*) répartis comme suit : 24% d'unigrammes, 42% de bigrammes et 20% de trigrammes, 8% de 4-grammes, 4% de 5-grammes et 1% de vide. Compte tenu de ces observations, nous avons décidé de transformer l'ensemble du vocabulaire issu des offres d'emplois sous forme d'unigrammes, de 2-grammes et 3-grammes et de les ordonner selon un score  $S_{job}$ , inspiré du *TF-IDF* :

$$S_{job}(u, m) = tf(u) \cdot \log \frac{D_m}{df(u)}$$

avec  $u$  l'unité lexicale considérée (unigramme, 2-grammes ou 3-grammes),  $tf(u)$  la fréquence de  $u$  dans la collection,  $df(u)$  le nombre de documents où l'unité lexicale  $u$  apparaît et  $D_m$  le nombre de documents associés au métier  $m$ .

Afin de filtrer certains termes et expressions courants dans les offres d'emplois et qui viennent parfois brouter les résultats (par exemple *employment equity*, *experience working*), nous effectuons par la suite une comparaison de la liste ordonnée de n-grammes obtenus avec un arbre préfixe contenant environ 25 000 compétences issues des profils de réseaux sociaux. Cette structure permet d'effectuer des comparaisons rapides entre l'arbre de compétences et les n-grammes candidats tout en conservant les variations d'écriture pour une même compétence (p. ex. *en charge de projet*, *en charge de projets*, *conduite de projets*, etc.).

### 5.4. Génération de l'ontologie

Une fois l'ensemble des comparaisons effectuées, nous générons une structure de données hiérarchique associant à chaque univers (*uni:n* dans la figure 2) par la relation *bpp:comprises* un ensemble de métiers *occ:i* nommés *occupations* par

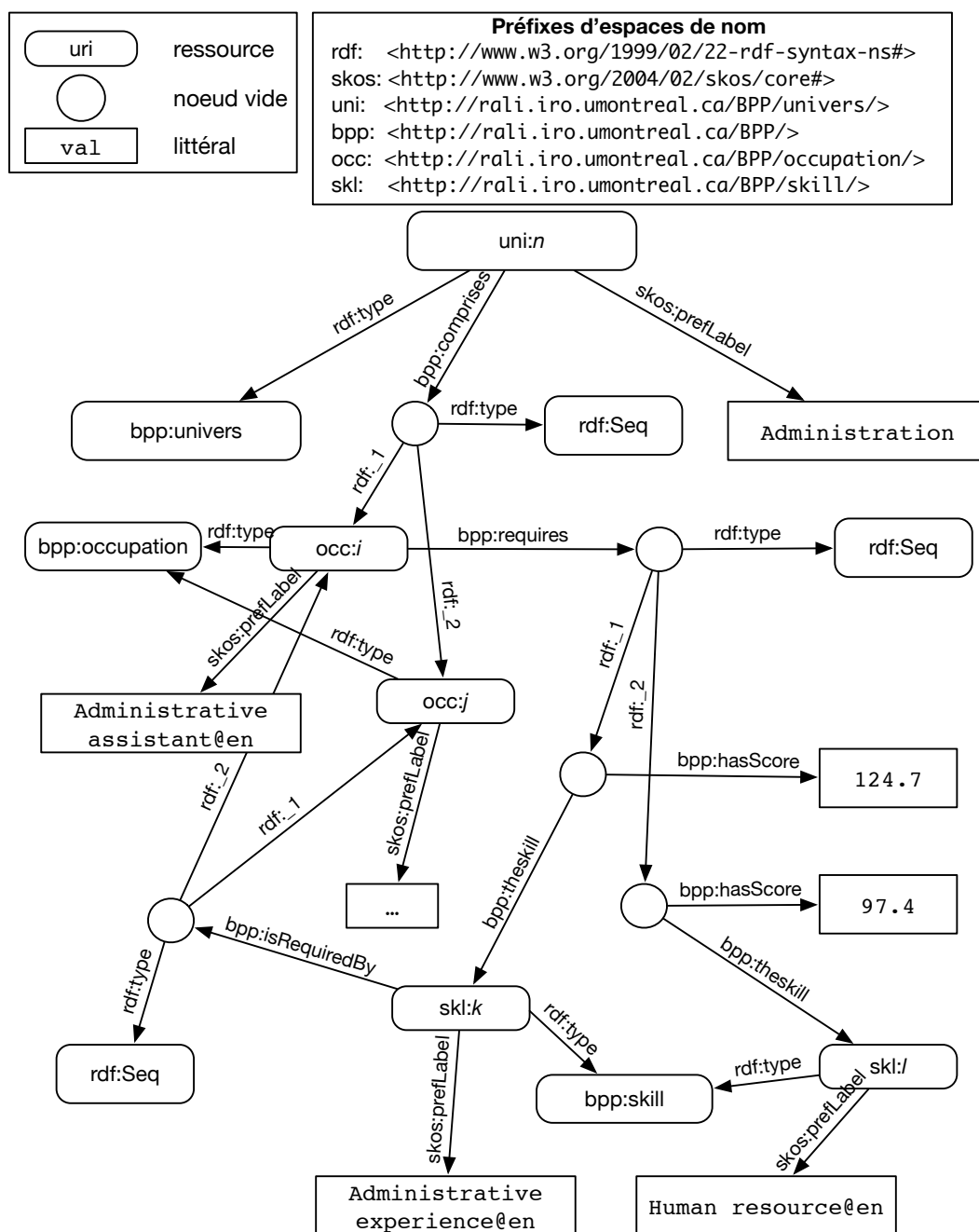


**Figure 2.** Organisation du graphe RDF de l'ontologie générée.

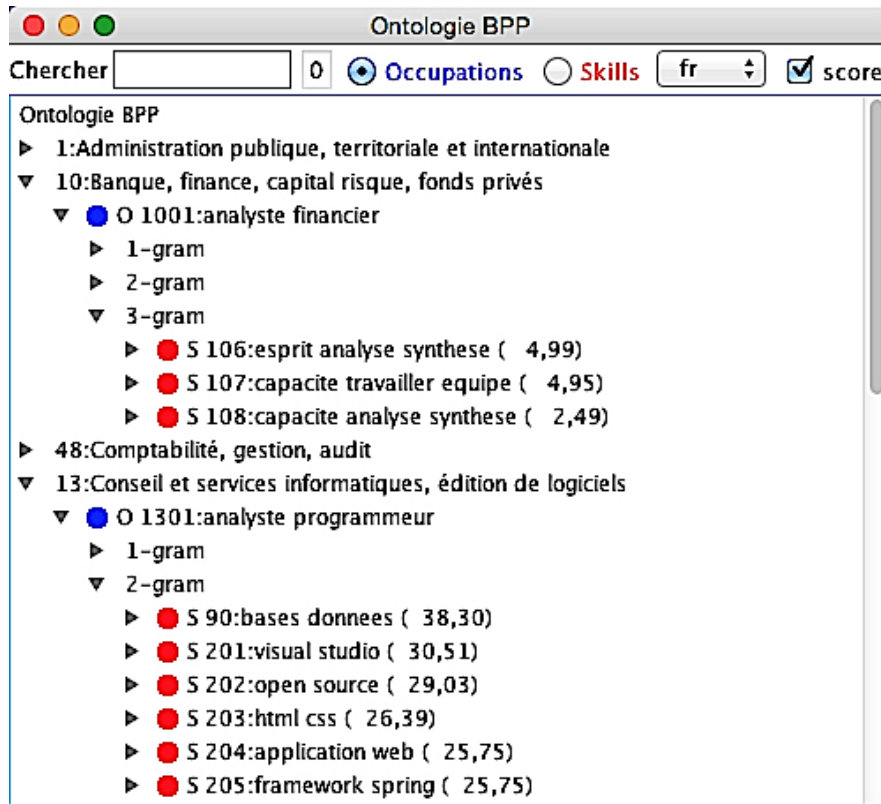


**Figure 3.** Browser affichant un extrait de l'ontologie. Chaque concept présent dans l'ontologie est déterminé à l'aide d'un identifiant numérique, les compétences sont précédées de la lettre S (Skills) les métiers par O (Occupations), suivi du numéro associé à l'univers (par exemple O1301 analyste programmeur appartient à l'univers 13 Conseil et services informatiques, édition de logiciels).

souci de compatibilité avec la nomenclature ESCO. Chaque occupation est liée par la relation `bpp:requires` à des listes de compétences (*skills* dans ESCO). Chaque compétence est liée à une étiquette par la relation `skos:prefLabel` sous forme de n-grammes et associée à un score `bpp:hasScore` qui est la valeur de  $S_{job}$ . Chaque métier est ainsi rattaché à son domaine ainsi qu'à une liste des compétences associées. De plus, un lien direct `bpp:isRequiredBy` est ajouté entre chaque compétence et le métier qui l'exige. Le réseau RDF pour l'anglais contient 25 569 triplets pour 57 métiers et 1608 compétences et celui pour le français comprend 4 183 triplets pour 11 métiers et 330 compétences. Ils sont interrogeables avec SPARQL dans nos applications. Pour en faciliter l'exploration visuelle, nous avons aussi développé un browser spécialisé qui affiche la structure RDF en faisant ressortir les différents niveaux : univers, métiers et compétences. La figure 3 présente une capture d'écran contenant un extrait de l'ontologie finale avec les métiers évalués en section 6.

| financial analyst |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| unigramme         | accounting, finance, financial, cma, budget, analysis, <del>end</del> , review, reporting, projects, support, <del>including</del> , process, <del>key</del> , business, <del>com</del> , <del>canada</del> , information, <del>knowledge</del> , <del>maintain</del> , develop, <del>qualifications</del> , responsibilities, <del>requirements</del> , <del>environment</del> , team, work, <del>strong</del> , <del>working</del> , <del>experience</del>                                                                                                                                                                                                                                                                                                                                                                                                        |
| bigrammes         | financial analyst, financial analysis, financial reporting, financial modeling, journal entries, professional accounting, variance analysis, accounting finance, financial planning, finance accounting, balance sheet, management reporting, financial reports, financial statements, finance business, <del>special projects</del> , financial systems, management reports, <del>internal-external</del> , analytical skills, <del>ability work</del> , financial services, strong analytical, <del>experience working</del> , strong organizational, business processes, decision making, communication skills, <del>skills strong</del> , fast paced                                                                                                                                                                                                            |
| trigrammes        | financial planning analysis, analytical problem solving, problem solving skills, advanced excel skills, communication interpersonal skills, verbal written communication, ability work independently, organizational time management, strong analytical skills, fast paced environment, ability multi task, oral written communication, exceptional interpersonal skills, financial analysis reporting, verbal communication skills, time management skills, strong communication skills, excel pivot tables, written oral communication, strong interpersonal skills, written verbal communication, ability work pressure, knowledge microsoft office, microsoft office suite, key performance indicators, work team environment, strong organizational skills, excellent interpersonal skills, excellent communication skills, interpersonal communication skills |

| analyste financier |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|--------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| unigramme          | finances, financement, analyse, <del>social</del> , finance, <del>public</del> , documents, <del>direction</del> , process, rigueur, audit, <del>experience</del> , office, procedures, <del>realisation</del> , business, <del>connaissance</del> , support, <del>formation</del> , organisation, gestion, service, <del>premiere</del> , <del>international</del> , commerce, reporting, responsable, developpement, <del>services</del> , <del>clients</del>                                                                                                                                                         |
| bigrammes          | vision strategique, tableaux bord, strategique gestion, strategie commerciale, sens organisation, sens analyse, <del>ressources humaines</del> , processus budgetaire, politiques publiques, optimisation processus, modelisation financiere, <del>mise place</del> , middle office, gestion risque, gestion projets, gestion priorites, gestion groupe, excel vba, etats financiers, <del>direction generale</del> , developpement local, developpement economique, creation entreprise, controle gestion, business plan, bases donnees, analyses financieres, analyse risques, analyse financiere, analyse economique |
| trigrammes         | esprit analyse synthese, capacite travailler equipe, capacite analyse synthese                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |

**Tableau 2.** Liste de compétences sous forme de n-grammes pour les métiers “financial analyst” et “analyste financier” obtenues à partir de respectivement 135 et 44 offres.

| programmer analyst |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| unigramme          | programmer, analyst, programming, net, application, test, applications, developing, support, technical, systems, design, development, tools, develop, data, <del>knowledge</del> , <del>following</del> , solutions, environment, maintain, requirements, professional, <del>including</del> , work, business, qualifications, team, <del>experience</del> , <del>time</del>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| bigrammes          | sql server, application development, web applications, programming languages, systems development, information technology, web services, unit testing, working knowledge, business requirements, working experience, crystal reports, visual studio, software development, <del>latin-america</del> , design development, technology solutions, version control, development team, database design, production support, web development, operating systems, best practices, wealth management, business systems, <del>general knowledge</del> , <del>experience working</del> , <del>internal-external</del> , communication skills                                                                                                                                                                                                                                                  |
| trigrammes         | object oriented programming, microsoft sql server, problem solving skills, sql reporting services, java enterprise edition, corporate social responsibility, team foundation server, sql server reporting, ability work independently, microsoft dynamics crm, <del>job description development</del> , <del>visual source safe</del> , enterprise content management, excellent communication skills, service oriented architecture, enterprise application integration, excellent organizational skills, customer service skills, able work independently, written verbal communication, analytical problem solving, software development lifecycle, build strong relationships, user acceptance testing, problem solving ability, strong analytical skills, work minimal supervision, written oral communication, strong interpersonal skills, communication interpersonal skills |

| analyste programmeur |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| unigramme            | applications, java, <del>environnement</del> , net, informatique, javascript, ingenierie, developpement, programmation, <del>end</del> , conception, mysql, <del>connaissance</del> , windows, maintenance, solutions, <del>experience</del> , <del>competences</del> , <del>information</del> , <del>techniques</del> , <del>direction</del> , <del>formation</del> , <del>accompagnement</del> , <del>service</del> , analyse, anglais, autonome, responsable, <del>services</del> , <del>clients</del>                                                                                  |
| bigrammes            | bases donnees, visual studio, open source, html css, application web, framework spring, apache tomcat, esprit equipe, <del>mise place</del> , <del>enseignement superieur</del> , systeme information, base donnees, sql server, <del>direction generale</del> , rpg iii, analyse besoins, suivi production, zend studio, administration reseau, <del>propriete intellectuelle</del> , <del>parler-ecrit</del> , developpement logiciel, nouvelles technologies, maitrise anglais, esprit synthese, dynamics crm, sens analyse, capacite analyse, <del>supply chain</del> , tableaux bord, |
| trigrammes           | microsoft sql server, esprit analyse synthese, capacite travailler equipe,                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |

**Tableau 3.** Liste de compétences sous forme de n-grammes pour les métiers “programmer analyst” et “analyste programmeur” obtenues à partir de respectivement 112 et 69 offres.

|                      | anglais |      |      | français |      |      |
|----------------------|---------|------|------|----------|------|------|
|                      | uni.    | bi.  | tri. | uni.     | bi.  | tri. |
| N-grammes total      | 60      | 60   | 60   | 60       | 60   | 6    |
| N-grammes pertinents | 42      | 51   | 58   | 34       | 51   | 6    |
| Précision            | 0.7     | 0.85 | 0.96 | 0.56     | 0.85 | 1    |

**Tableau 4.** *synthèse de l'évaluation des résultats.*

## 6. Résultats

Nous présentons ici les compétences en ordre décroissant de  $S_{job}$  (voir section 5.3) pour les métiers *analyste financier* (Tableau 2) et *analyste programmeur* (Tableau 3) en français et en anglais. Cette portion de l'ontologie a été évaluée par un expert du domaine. Nous avons barré les n-grammes qui ont été considérés comme non pertinents par celui-ci. Le tableau 4 présente une synthèse de cette évaluation en termes de *précision*. Ne disposant pas de liste de l'ensemble des compétences pour un métier, nous n'avons pas été en mesure de calculer le *rappel*.

Même si l'échantillon évalué est de taille relativement petite, les résultats obtenus sont de très bonne qualité (0.79 sur l'ensemble de l'évaluation). La qualité des listes en anglais (0.83) est meilleure que celles en français (0.75). Nous attribuons cette différence au plus petit nombre d'offres d'emploi pour les métiers considérés en langue française (135 et 112 offres en anglais contre 44 et 69 offres en français). Ce nombre restreint d'offres d'emploi en français explique aussi le faible nombre de trigrammes obtenus pour les métiers d'*analyste financier* et d'*analyste programmeur*.

Le tableau 4 montre des résultats de meilleure qualité pour les bigrammes et trigrammes que pour les unigrammes (respectivement 0.63 pour les unigrammes, 0.85 pour les bigrammes et 0.97 pour les trigrammes), quelle que soit la langue considérée.

Nous observons cependant un certain nombre de compétences transversales (tels que *verbal/written communication skills*, *capacité à travailler en équipe*, etc. ) qui peuvent être considérées pertinentes quelque soit le métier considéré. Nous envisageons des traitements particuliers afin d'être capable de les distinguer des compétences plus techniques, communément appelées *hard skills*<sup>11</sup>. Nous constatons par ailleurs une redondance de certaines compétences en fonction de l'usage du singulier ou du pluriel (par exemple : *analyse financière* et *analyses financières*, *bases donnees* et *base donnees*). De la même façon, on observe une fréquence importante de certaines compétences (par exemple *excel*, *microsoft office*, etc. ) qui se retrouvent dans un grand nombre de métiers ainsi que certain termes/expressions courants dans les offres d'emplois et qui viennent parfois bruyter les résultats (par exemple *employment equity*, *human ressources*, *strong experience required*, etc. ).

11. Les hard skills sont nos compétences formellement démontrables, nées d'un apprentissage technique, souvent d'ordre académique, et dont la preuve est apportée par l'obtention de notes, diplômes, certificats.



## 7. Conclusion et travaux futurs

Nous avons présenté dans cet article les travaux réalisés sur la génération automatique de ressources linguistiques pour les besoins de l'e-recrutement. À partir de 10 millions de profils issus de plusieurs réseaux sociaux, ainsi que 100 000 offres d'emplois récoltées sur internet, nous avons fait émerger des compétences et des connaissances communes pour construire un ensemble structuré des termes et concepts représentant chaque métier. Ces offres contenant le profil minimal recherché pour un métier ont été utilisées afin de détecter les compétences associées. Ces offres d'emploi étant collectées sur internet, le modèle pourra ainsi s'enrichir de façon semi-automatique avec l'apparition et la disparition de métiers. Nous avons présenté par la suite les premiers résultats obtenus ainsi qu'une évaluation manuelle réalisée par un expert du domaine. L'analyse détaillée a montré des résultats de très bonnes qualités (précision moyenne de 0,79) particulièrement en anglais. Nous attribuons cette différence à la quantité plus faible d'offres d'emploi pour les métiers considérés en langue française.

Nous envisageons des traitements plus fins afin de distinguer les compétences transversales des compétences spécifiques à chaque métier. Nous souhaitons utiliser cette ontologie afin d'évaluer si un candidat dispose toutes les compétences pour un métier ou encore au travers d'un outil de génération de texte afin de suggérer à un candidat comment mettre en avant ses compétences ou encore suggérer des mots-clés pour les recruteurs qui effectuent des recherches dans des bases de profils de candidats. Nous prévoyons par ailleurs de continuer à augmenter la taille de la collection d'offres d'emplois afin de couvrir un nombre de métiers plus important pour améliorer la qualité des résultats.

## Remerciements

Les auteurs tiennent à remercier le Conseil de Recherches en Sciences Naturelles et en Génie du Canada (CRSNG), qui a partiellement financé ces travaux dans le cadre d'une subvention de recherche et développement coopérative.

## 8. Bibliographie

- Colucci S., Di Noia T., Di Sciascio E., Donini F. M., Mongiello M., Mottola M., « A formal approach to ontology-based semantic match of skills descriptions », *J. UCS*, vol. 9, n° 12, p. 1437-1454, 2003.
- Colucci S., Di Noia T., Di Sciascio E., Donini F. M., Ragone A., Trizio M., « A Semantic-based Search Engine for Professional Knowledge », *Proc. 7th Int. Conf. on Knowledge Management and Knowledge Technologies (I-KNOW 2007)*, (Sep 2007), p. 472-475, 2007.
- Colucci S., Tinelli E., Giannini S., Di Sciascio E., Donini F. M., « Knowledge Compilation for Core Competence Extraction in Organizations », *Business Information Systems*, Springer, p. 163-174, 2013.

- Desmontils E., Jacquin C., Morin E., « Indexation sémantique de documents sur le Web : application aux ressources humaines », *Journées de l'AS-CNRS Web Sémantique*, 2002.
- Gómez-Pérez A., Ramírez J., Villazón-Terrazas B., « An ontology for modelling human resources management based on standards », *Knowledge-Based Intelligent Information and Engineering Systems*, Springer, p. 534-541, 2007.
- Kessler R., Béchet N., Roche M., El-Bèze M., Torres-Moreno J. M., « Automatic profiling system for ranking candidates answers in human resources », *On the Move to Meaningful Internet Systems : OTM 2008 Workshops*, Springer, p. 625-634, 2008.
- Lau T., Sure Y., « Introducing ontology-based skills management at a large insurance company », *Proceedings of the Modellierung 2002p*. 123-134, 2002.
- le Vrang M., Papantoniou A., Pauwels E., Fannes P., Vandensteen D., De Smedt J., « ESCO : Boosting Job Matching in Europe with Semantic Interoperability », *Computer*, vol. 47, n° 10, p. 57-64, Oct, 2014.
- Loth R., Battistelli D., Chaumartin F.-R., De Mazancourt H., Minel J.-L., Vinckx A., « Linguistic information extraction for job ads (SIRE project) », *Adaptivity, Personalization and Fusion of Heterogeneous Information*, p. 222-224, 2010.
- Mochol M., Simperl E. P. B., « Practical guidelines for building semantic erecruitment applications », *International Conference on Knowledge Management, Special Track : Advanced Semantic Technologies (AST'06)*, Citeseer, 2006.
- Roche M., Kodratoff Y., « Pruning terminology extracted from a specialized corpus for CV ontology acquisition », *On the Move to Meaningful Internet Systems 2006 : OTM 2006 Workshops*, Springer, p. 1107-1116, 2006.
- Trichet F., Bourse M., Leclerc M., Morin E., *Human Resource Management and Semantic Web Technologies*, springer edn, ICTTA, Berlin, 2004.
- Trog D., Christiaens S., Zhao G., de Laaf J., « Toward a Community Vision Driven Topical Ontology in Human Resource Management », *On the Move to Meaningful Internet Systems : OTM 2008 Workshops*, Springer, p. 615-624, 2008.
- Tétreault M., Dufresne A., Gagnon M., « Development of an Ontology-Based E-Recruitment Application that Integrates Social Web », *Electronic business interoperability : Concepts, opportunities and challengesp*. 363-395, 2011.



CORIA – CIFED 2016

Apprentissage et métriques

Président : *Éric Gaussier*



---

# Réseaux de Neurones pour la Représentation de Contextes Continus des Mots

Killian Janod<sup>\*,\*\*</sup> — Mohamed Morchid<sup>\*</sup> — Richard Dufour<sup>\*</sup> — Georges Lianrès<sup>\*</sup>

<sup>\*</sup> Université d'Avignon, LIA (France), <sup>\*\*</sup> ORKIS - Aix en Provence (France)

---

**RÉSUMÉ.** Les méthodes d'apprentissage profond s'appuient de plus en plus sur des représentations vectorielles continues des mots. Ces méthodes, déjà appliquées avec succès dans de nombreuses tâches de traitement automatique du langage naturel écrit et oral, sont capables de représenter des mots ainsi que les relations les liant. De manière générale, ces méthodes utilisent des représentations par "sac-de-mots" et traitent donc tous les mots d'un contexte de façon égale. Cet article propose une méthode originale qui s'appuie sur les modèles de contextes continus en intégrant la position relative des mots dans un contexte. Les résultats montrent que l'information portée par les contextes continus permet un gain jusqu'à 7 % sur le test qualitatif "de relation sémantique" et permet d'obtenir des résultats pertinents pour une application concrète (identification de thèmes de dialogues dans le cadre du projet DECODA), alors que des performances faibles sont obtenues sans l'information contextuelle.

**ABSTRACT.** Deep learning methods use more and more word embedding representations. Those methods, which have been already applied with success on various tasks of written and spoken natural language processing, are able to represent words and the relations between them. Usually in those methods context windows are represented as bag-of-words, i.e. every word in the context is treated equally. This paper proposes an original method inspired from the Continuous Context Models by integrating words relative positions. The results observed confirm that the information given by continuous context models allow us to gain more than 7 % on the Word Relationship test and achieve relevant results on a real application ( theme identification with the DECODA corpus) that couldn't be done without this original information .

**MOTS-CLÉS :** sac-de-mots, Contextes Continus, réseau de neurones artificiel, Word2vec

**KEYWORDS:** Word embeddings, Continuous context, Neural network, Word2vec

---

## 1. Introduction

Le choix d'une bonne représentation des mots est souvent indispensable pour mener à bien des tâches de traitement automatique du langage naturel écrit ou oral. La représentation par "sac-de-mots" est la plus couramment utilisée, les documents étant représentés par les vecteurs de fréquence des mots qui les composent. Cependant, cette représentation a pour défaut de ne capturer que très peu d'informations sur la structure des documents et leur relation entre eux. Pour cette raison, les approches à base de  $n$ -grammes ont été introduites. Cette représentation associe à chaque mot son historique proche pour modéliser des occurrences plus complexes. Dans (Sahlgren, 2008), l'auteur propose de capturer les relations sémantiques entre les mots en utilisant "l'Hypothèse de Distribution" qui dicte que "*les mots apparaissant dans le même contexte ont la même signification*". Les représentations découlant de cette hypothèse projettent les mots dans l'espace de tous les contextes présents dans les données d'apprentissage. Les projections dans ces espaces sont le plus souvent creuses et de très grande dimension. Ces projections souffrent de ce qui est appelé "Fléau de la dimension". D'autres représentations avec un plus haut niveau d'abstraction, comme l'Allocation Latente de Dirichlet (*Latent Dirichlet Allocation* (LDA) (Blei *et al.*, 2003)) cherchent à modéliser les thèmes latents d'un corpus de documents.

Plus récemment, des méthodes s'appuyant sur des réseaux de neurones artificiels (Bengio *et al.*, 2003) se sont imposées. Ces méthodes sont capables de représenter un mot par un vecteur plein de taille contrôlée. Ce vecteur correspond à la position du mot dans un espace multi-dimensionnel. Ce type de technique est appelé *Word embedding*. Ces méthodes étaient d'abord employées pour des Modèles de Langue Neuronaux (Bengio et Heigold, 2014 ; Collobert *et al.*, 2011) et ensuite appliquées dans de nombreuses tâches de traitement du langage. (Do *et al.*, 2014 ; Vaswani *et al.*, 2013 ; Mesnil *et al.*, 2015). Parmi ces approches, la méthode *Word2vec* (Mikolov *et al.*, 2013a) apparaît aujourd'hui à l'état-de-l'art en matière de représentation distribuée des mots. *Word2vec* propose deux architectures s'appuyant sur les réseaux de neurones artificiels et conçues pour traiter une grande quantité de textes ainsi qu'inférer une structure linéaire qui modélise des relations sémantiques et syntaxiques liant les mots. Cette modélisation a prouvé son efficacité dans plusieurs tâches en traitement du langage naturel (TALN) oral et écrit. Les réseaux de neurones artificiels *Word2vec* construisent, pour chaque mot, une fenêtre de contexte. À l'intérieur de cette fenêtre, tous les mots sont traités de façon égale.

Ce papier introduit une pondération log-linéaire des mots s'appuyant sur les contextes continus de ces mots (Bigot *et al.*, 2013) intégrant une information structurelle des mots. Cette méthode permet de réduire l'importance de la taille maximum de la fenêtre du contexte. Cette approche originale a été appliquée sur deux tâches du TALN : le test "*Semantic-Syntactic Word Relationship*" (Mikolov *et al.*, 2013a), permettant de vérifier que le modèle a correctement capturé les relations sémantiques et syntaxiques liant les mots, et une tâche plus appliquée dans un contexte réaliste, à savoir l'identification de thèmes dans des dialogues bruités dans le cadre du projet DECODA (Bechet *et al.*, 2012).

L'article est organisé comme suit : la section 2 détaille les deux architectures de réseaux de neurones artificiels *Word2vec* ainsi que la fonction de pondération intégrée dans ces architectures. Les expériences et leurs résultats sont décrits dans la section 3 avant de conclure dans la section 4.

## 2. Approche proposée

La section 2.1 présente le framework *Word2vec* en détaillant l'architecture réseaux de neurones artificiels "Sac-de-mots continus" (*Continuous Bag-Of-Words*, *CBOW*), et l'architecture *Skip-gram*. Les fenêtres de contextes dynamiques et l'approche s'appuyant sur les modèles de contextes continus sont définies et expliquées dans la section 2.2.

### 2.1. Les réseaux de neurones artificiels *Word2vec*

La méthode *Word2vec* est définie dans (Mikolov *et al.*, 2013a). Cette méthode propose deux réseaux de neurones artificiels simples (*i.e.* peu profonds) : l'architecture *CBOW* et l'architecture *Skip-gram*. Ces architectures ont besoin, pour s'entraîner, de mots centraux (ou mots d'attention) et leurs fenêtres de contexte respectives. Une fenêtre de contexte correspond aux  $n$  mots précédents et  $n$  mots suivants un mot central. La valeur de  $n$  est à adapter selon la tâche et les données.

Chacune de ces architectures est composée de trois couches. Une couche d'entrée, une couche cachée et une couche de sortie.

Le schéma 1 présente l'architecture *CBOW* et le schéma 2 l'architecture *Skip-gram* pour le segment de texte : " $w_{-2} w_{-1} w_{central} w_{+1} w_{+2}$ ", et dont le mot d'attention est  $w_{central}$ . La couche d'entrée contient un sac-de-mots contenant la fenêtre de contexte pour le *CBOW* ou le mot central pour le *Skip-gram*. La couche cachée contient la projection de l'entrée dans la matrice globale des poids. La couche de sortie est la prédiction du modèle. Soit un mot pour l'architecture *CBOW*, et un contexte pour l'architecture *Skip-gram*. Cette prédiction est uniquement utilisée pour calculer l'erreur des réseaux et la rétro-propagation du gradient. Cette rétro-propagation permet de corriger la matrice globale en rapprochant dans l'espace multi-dimensionnel les mots de leurs contextes respectifs.

La couche de sortie des réseaux est composée de neurones artificiels avec une fonction d'activation *Softmax*. Dans le but de réduire la complexité algorithmique de cette fonction, les auteurs dans (Mikolov *et al.*, 2013a) ont proposé deux alternatives : le *Softmax hiérarchique* (*Hierarchical softmax*) et l'*échantillonnage négatif* (*negative sampling*). Elles ont permis un accroissement important de la vitesse de traitement du réseau. D'après (Mikolov *et al.*, 2013a), cette augmentation de la vitesse de traitement permet aux modèles d'apprendre sur de plus importantes quantités de textes dans un temps raisonnable, un plus grand nombre de données d'apprentissage impliquant de meilleures représentations (cf. schéma 3).

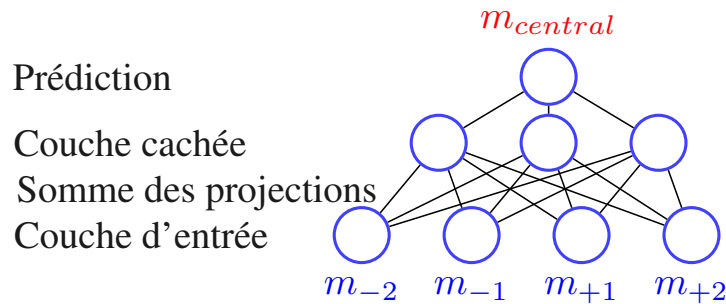


**Figure 1.** Réseau de neurones CBOW sans pondération qui prédit un mot sachant les mots dans la fenêtre de contexte (deux mots avant et deux mots après dans ce schéma).

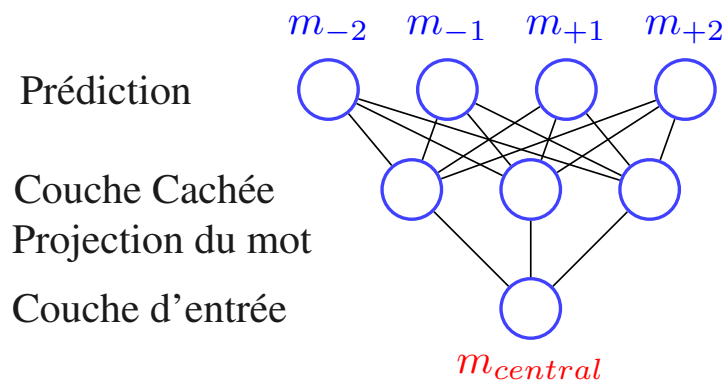


**Figure 2.** Réseau de neurones Skip-gram sans pondération qui tente de prédire sachant un mot son contexte (deux mots avant et deux mots après dans ce schéma).

#### 2.1.1. Sacs-de-mots continus (CBOW)

L'architecture du CBOW est un réseau de neurones artificiels simple et log-linéaire. La couche d'entrée de ce réseau de neurones utilise des "sacs-de-mots" binaires représentant une fenêtre de contexte. Dans cette configuration, les vecteurs d'entrée sont des vecteurs de la taille du vocabulaire avec un 1 dans la colonne  $i$  si le mot  $i$  est présent dans le document, 0 sinon. Chaque mot est alors projeté dans la matrice globale, l'ensemble des représentations est ensuite additionné pour former une unique couche cachée. Cette couche cachée passe par la couche de sortie et les fonctions d'activation type "Softmax" tentent de prédire le mot au coeur de la fenêtre. L'erreur de prédiction



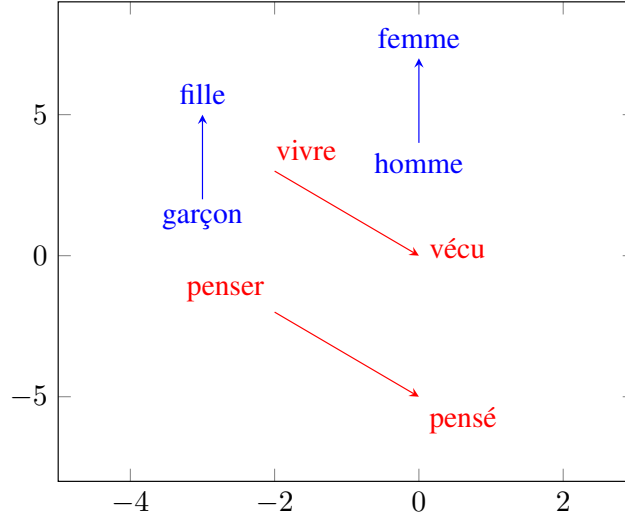


**Figure 3.** Exemple de relations inter-mots

est ensuite utilisée pour corriger les matrices de poids via une rétro-propagation de gradient. Cette architecture essaie de maximiser la vraisemblance ci-dessous :

$$\frac{1}{T} \sum_{t=1}^T \log p(w_t | w_{t-\frac{c}{2}} \dots w_{t+\frac{c}{2}}) \quad [1]$$

dans laquelle  $T$  est la taille des données d'apprentissage et  $c$  est la taille maximum de la fenêtre de contexte. L'architecture *CBOW* est plus efficace que son homologue *Skip-gram* et capture une meilleure représentation des mots fréquents (Mikolov *et al.*, 2013a).

### 2.1.2. Architecture *Skip-gram*

L'architecture *Skip-gram* est également un réseau de neurones artificiels simple et log-linéaire. Ce réseau apprend, contrairement au *CBOW*, à prédire une fenêtre de contexte sachant le mot au cœur de celle-ci (voir figure 2). La couche d'entrée du réseau ne contient donc que la représentation en sac-de-mots binaire du mot au cœur du contexte. Ce mot est projeté dans la matrice de poids globale, puis transmis à la couche de sortie qui va prédire un mot. Cette prédiction est ensuite corrigée par rétro-propagation pour chacun des mots de la fenêtre de contexte. Un réseau *Skip-gram* maximise la vraisemblance suivante :

$$\frac{1}{T} \sum_{t=1}^T \sum_{j=t-c, j \neq t}^{t+c} \log p(w_j | w_t) \quad [2]$$

Comparativement au *CBOW*, les réseaux *Skip-gram* apprennent de meilleures représentations sémantiques et sont plus adaptés aux mots peu fréquents (Mikolov *et al.*, 2013b).

## 2.2. Fenêtre de contexte dynamique (DW) et Fonction de pondération par contextes continus (CCM)

Dans (Mikolov *et al.*, 2013a), la fenêtre de contexte dynamique est définie et attribuée au modèle *Skip-gram*. Mais le “Word2vec toolkit”<sup>1</sup> applique cette méthode à la fois pour le *CBOW* et le *Skip-gram*. La fenêtre de contexte dynamique permet au modèle *Skip-gram* d’ignorer aléatoirement des mots aux extrémités de la fenêtre de contexte. Une réduction de la fenêtre implique une réduction du nombre de rétro-propagations et donc une accélération du traitement. En contrepartie, le réseau ignore parfois des relations de plus longue distance. En ignorant de façon régulière les mots éloignés du centre de la fenêtre de contexte, le réseau applique une forme de pondération linéaire aux relations éloignées. En plus d’ignorer des relations, cette méthode attribue à tous les mots dans la fenêtre de contexte la même importance (binaire). Une fonction de pondération log-linéaire s’appuyant sur les modèles de contextes continus (Bigot *et al.*, 2013) est proposée comme alternative à la fenêtre de contexte dynamique. Cette nouvelle fonction de pondération donne alors un poids à chaque mot qui ne dépend que de la distance séparant les mots du contexte à pondérer du mot au cœur du contexte. La fonction de pondération de distance est définie par :

$$\frac{\alpha}{b + \beta \log(d)} \quad [3]$$

où  $d$  est la distance, en nombre de mots, séparant le mot à pondérer et le mot de référence ;  $\alpha$ ,  $b$  et  $\beta$  sont les paramètres de la fonction de distance qui définissent l’impact des mots les plus éloignés. Le graphique 4 montre la différence d’importance accordée aux mots d’un contexte selon son éloignement au mot central, pour l’approche classique avec fenêtre de contexte dynamique (dynamic window ou DW) et avec la pondération par contextes continus *CCM*. Notons que l’importance attribuée par la fenêtre dynamique n’est vraie que pour un nombre de tirage infini d’un couple *mot-contexte* en particulier.

La fonction de contextes continus est capable de mettre en avant les mots les plus proches tout en conservant les relations plus distantes (cf. schéma 4). En effet, une approche par pondération permet de conserver tous les mots du contexte (cf. figure 4-courbe rouge), alors qu’avec un re-échantillonnage, les relations ont seulement une probabilité d’être capturée (cf. figure 4-histogramme bleu). Une méthode par pondération a deux effets sur l’apprentissage : d’un côté, en pondérant les mots lors de l’apprentissage au lieu de les négliger, elle permet au réseaux de neurones artificiels d’utiliser cette information de distance ; d’un autre côté, ignorer des mots est aussi utile pour rendre le modèle plus rapide. La méthode par pondération ralentit les réseaux *Skip-gram* de 20%. Par contre, un impact négligeable sur le réseau *CBOW* a été observé dans le cadre de nos expériences.

1. <https://code.google.com/p/word2vec>

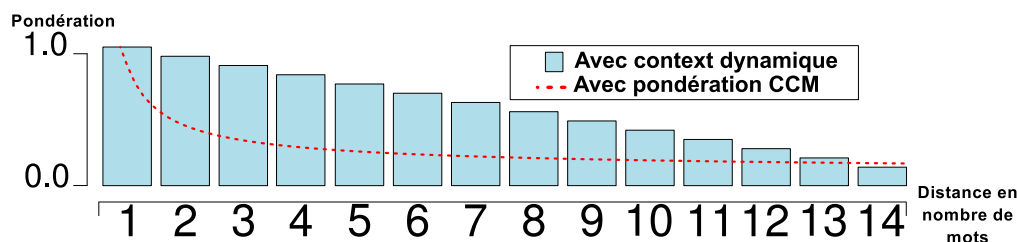


**Figure 4.** Importance des mots dans un contexte avec et sans modèle de contextes continus. L’histogramme présente l’espérance de la pondération attribuée globalement pour un contexte donnée par ré-échantillonnage du contexte. La courbe rouge présente la pondération effectivement attribuée dans une fenêtre de contexte par la fonction de contextes continus.

### 3. Expériences et Résultats

Dans cette section, nous allons d’abord analyser les différences des représentations produites par des modèles avec et sans fonction de pondération *CCM* (cf. section 3.1). Deux expériences sont expliquées ensuite pour évaluer l’efficacité de l’approche avec une pondération par contextes continus (cf. sections 3.2 et 3.3).

#### 3.1. Impact sémantique des modèles de contextes continus (*CCM*)

En comparant les voisins de plusieurs exemples de mots (cf. tableau 1), nous pouvons remarquer que, avec la pondération *CCM* que nous proposons, les mots ont tendance à être regroupés par thématique, alors que les mots avec la fenêtre classique “DW” ne le sont pas. Par exemple, pour le mot “Holidays”, nous trouvons d’abord la thématique des *loisirs* avec (“holiday”, “vacation” and “festivities”) puis en second plan la thématique *religion* (“thanksgiving”, “easter” et “christmas”). Comme cette séparation thématique est une observation de l’effet de la fonction de pondération *CCM* plus qu’une évaluation, l’impact des modèles avec pondération *CCM* est mesuré plus en détail par deux expériences différentes. La première expérience, détaillée dans la section 3.2, évalue la capacité du modèle à apprendre les relations sémantiques et syntaxiques des données d’apprentissage. Bien que cette évaluation permette de mesurer la qualité intrinsèque des modèles, elle en dit peu sur leur efficacité dans des cas d’utilisation plus pratiques. La seconde expérience, présentée dans la section 3.3, a pour but de mesurer l’efficacité du modèle dans une tâche d’identification de thématique conversationnelle.

#### 3.2. Evaluation des relations sémantiques et syntaxiques

Le “*Semantic-Syntactic Word relationship test set*” est défini dans (Mikolov et al., 2013a). L’objectif de cette suite de tests est de vérifier que la représentation

**Tableau 1.** Exemple de contexte des mots sans (DW) et avec pondération CCM.

| Holidays      |              | Meat          |                |
|---------------|--------------|---------------|----------------|
| Avec Distance | Avec DW      | Avec Distance | Avec DW        |
| holiday       | vacations    | chicken       | pork           |
| vacation      | thanksgiving | beef          | not-pasterised |
| festivities   | vacation     | pork          | mutton         |
| thanksgiving  | christmas    | milk          | eggs           |
| easter        | celebration  | eggs          | cattle         |
| christmas     | easter       | seafood       | chicken        |

des mots a bien capturé les relations sémantiques et syntaxiques liant les mots. Ce test est composé d'environ 19 000 questions divisées en deux catégories (Sémantique et Syntaxique). Chaque question est composée de deux paires de mots partageant une même relation. Par exemple les mots : *USA - Dollars* et *Europe - Euros* partagent la relation **est la monnaie de**. Chaque question est considérée comme correcte si l'équation [4] est vraie dans l'espace multidimensionnel.

$$w_{1c1} + w_{2c1} = w_{1c2} + w_{2c2}. \quad [4]$$

La nature polysémique des mots, ainsi que l'absence de contrainte sur l'espace de projection, impliquent que l'équation [4] est pratiquement impossible. L'utilisation des propriétés, notamment algébriques, de l'espace de projection permettent de relâcher quelques contraintes sur la formulation de la question qui devient  $w_{1c2}$  **est-il le mot le plus proche de**  $w_{1c1} - w_{2c1} + w_{2c2}$ . La métrique utilisée, appelée couramment "précision", évalue un modèle par la proportion de bonnes réponses par rapport au nombre de questions. Tous les modèles utilisés pour cette expérience sont entraînés avec le même corpus de données en anglais composé de :

- l'ensemble de données "One Billion Word Language Modeling Benchmark" (31 millions de documents - 700 millions de mots),
- premier milliard de caractères d'un corpus issu de Wikipedia (124 303 documents - 124 millions de mots),
- corpus GigaWord anglais de 1994 à 2011 (190 millions de documents - 3 771 millions de mots),
- Brown Corpus (57 341 documents - 1 million de mots).

Tous les documents du corpus sont passés en minuscule. Puis les mots sont segmentés par la présence d'espaces et de tirets. Enfin, la ponctuation a été retirée. Après pré-traitement, l'ensemble du corpus contient 4 milliards de mots pour un vocabulaire d'environ 1 million de mots uniques. L'entraînement d'un modèle *Skip-gram* sur ces données a été réalisé pendant environ 12 heures sur une machine possédant 8 cœurs.

Pour cette expérience, plusieurs réseaux de neurones artificiels avec des paramètres différents ont été évalués. Le modèle faisant office de *baseline* (comme défini dans (Mikolov *et al.*, 2013a)) est composé d'une couche cachée de 300 neurones et d'une fenêtre de contexte de taille 10. Deux tailles de couches cachées (120 et 300) et trois tailles de contextes (10, 15 et 100) sont comparées. Comme moins de 1 % des documents du corpus dépassent les 100 mots, un contexte de taille 100 est ici considéré comme équivalent à l'utilisation du document entier comme contexte. Enfin, deux fonctions de pondération sont utilisées (Distance 1 et Distance 2) définies par l'équation [5] :

$$Distance1 = \frac{1 + \log(2)}{1 + \log(d)} \quad \text{et} \quad Distance2 = \frac{\log(10)}{5 * \log(d)} \quad [5]$$

où  $d$  est la distance entre le mot auquel on attribue un score et le centre du contexte en nombre de mots.

### Résultats

Les tableaux 2 et 3 montrent que l'utilisation de la fonction de pondération par *CCM* obtient globalement de meilleurs résultats. L'écart de performance le plus important est obtenu en considérant un document entier comme contexte, atteignant un gain absolu de précision de 7 % pour le *CBOW* et de 7,7 % pour le *Skip-gram*. Nous pouvons aussi noter, dans le tableau 2, que plus le contexte est large, plus le gain est important. De plus, le tableau 3 montre que la Distance 1 est plus favorable au modèle *Skip-gram*, avec des gains atteignant 2,1 % en absolu, alors que la Distance 2 est favorable au modèle *CBOW*, avec des gains absolus en précision jusqu'à 4,3 %. Enfin, le tableau 3 met en évidence que plus la couche cachée est large (i.e. plus le nombre de neurones est important) plus les modèles sont capables de retenir de l'information portée par les contextes continus.

**Tableau 2.** Précision (%) en fonction de la taille des contextes (10, 15 and 100 mots).

|                            | Skip-gram |      |      | CBOW |      |      |
|----------------------------|-----------|------|------|------|------|------|
| Taille de la couche cachée | 300       |      |      |      |      |      |
| Taille du contexte         | 10        | 15   | 100  | 10   | 15   | 100  |
| avec DW                    | 50.0      | 50.9 | 43.7 | 39   | 38.9 | 36.9 |
| avec Distance 1            | 55.0      | 53.7 | 51.4 | 39.9 | 39.6 | 43.9 |

**Tableau 3.** Précision (%) sans ( avec DW) et avec pondération CCM (Distance 1 et Distance 2) dans des espaces plus petits.

|                            | <i>Skip-gram</i> |      | <i>CBOW</i> |      |
|----------------------------|------------------|------|-------------|------|
| Taille du contexte         | 10               |      |             |      |
| Taille de la couche cachée | 120              | 300  | 120         | 300  |
| Avec DW                    | 43.9             | 50.0 | 29.0        | 39.0 |
| Avec Distance 1            | 45.1             | 55.0 | 30.3        | 39.9 |
| Avec Distance 2            | 40.0             | 52.1 | 31.5        | 43.5 |

**Tableau 4.** *Description des thèmes du corpus DECODA.*

| label                 | Nombre d'échantillons |            |            |
|-----------------------|-----------------------|------------|------------|
|                       | Train                 | Dev        | Test       |
| Problème d'itinéraire | 145                   | 44         | 67         |
| Objets trouvés        | 143                   | 33         | 63         |
| Horaires              | 47                    | 7          | 18         |
| Carte de transport    | 106                   | 24         | 47         |
| État du trafic        | 202                   | 45         | 90         |
| Prix du ticket        | 19                    | 9          | 11         |
| Infractions           | 47                    | 4          | 18         |
| Offres spéciales      | 31                    | 9          | 13         |
| <b>Total</b>          | <b>740</b>            | <b>175</b> | <b>327</b> |

### 3.3. Classification de conversations transcrites automatiquement

La seconde évaluation mesure les performances des deux types de représentation pour une tâche de classification automatique. Cette expérience est réalisée au moyen du corpus du projet DECODA (Bechet *et al.*, 2012 ; Morchid *et al.*, 2015 ; Morchid *et al.*, 2014a ; Morchid *et al.*, 2014b), ayant pour objectif d'identifier le thème abordé dans une conversation téléphonique. Le corpus DECODA est composé de 1 067 conversations téléphoniques découpées en trois morceaux, un corpus d'entraînement (Train) de 740 dialogues, un corpus de développement (Dev) de 175 dialogues, et 327 dialogues dans un corpus de Test. Les conversations ont été manuellement réparties parmi 8 thèmes.

Le système de reconnaissance de la parole utilisé est Speeral (Linares *et al.*, 2007). Le modèle acoustique est estimé sur 150 heures de documents parlés en condition téléphonique. Le vocabulaire du système est de 5 782 mots. Un modèle de langue tri-grammes est appris à partir d'un modèle de langue standard adapté avec les transcriptions du corpus de train. Une liste d'arrêt de 126 mots est utilisée<sup>2</sup> afin de supprimer les mots inutiles. Le système de transcription atteint un taux d'erreur-mot de 33,8 % sur le train, de 45,2 % sur le dev., et 49,5 % sur le test. Ces taux d'erreur-mot particulièrement élevés sont principalement dus à des disfluences verbales et à des mauvaises conditions acoustiques (par exemple, bruits de fond ou communication depuis un smartphone).

La projection des conversations dans l'espace multidimensionnel Word2vec est réalisée de la manière suivante : tout d'abord, les 1000 mots les plus discriminants sont sélectionnés en utilisant un combinaison de TF-IDF associé à un critère de pureté de Gini. Chaque dialogue est ensuite associé à un vecteur de scores représentant la distance entre la somme des mots du dialogue et chacun des mots discriminants. Enfin, les vecteurs de distance sont utilisés pour attribuer un thème à chaque conversation au moyen d'une approche de classification automatique. Pour mesurer l'efficacité de la

2. <http://code.google.com/p/stop-words/>

pondération *CCM* proposée, 4 modèles sont utilisés : deux architectures *CBOW*, une avec la Distance 2 et une avec DW, et deux modèles *Skip-gram*, un avec la Distance 1 et un avec DW. Comme le projet DECODA est en français, ces modèles sont entraînés sur le corpus français suivant :

- le corpus français GigaWord (17 millions de documents - 500 millions de mots),
- une partie de Wikipedia (16 millions de documents - 400 millions de mots),
- des extraits de presse française issus de l’Agence France Presse (AFP), Le Monde et Le Soir (56 millions de documents - 737 millions de mots),
- un ensemble de documents extraits d’Internet (4 millions de documents - 108 millions de mots),
- des transcriptions manuelles issues de campagnes d’évaluation récentes (ESTER, EPAC, ETAPE et REPERE) (411 000 documents - 379 millions de mots).

Ce corpus subit la même phase de pré-traitement que celui présenté dans la section 3.2. Ce corpus contient approximativement 2 milliards de mots pour un vocabulaire d’environ 3 millions de mots uniques. Pour cette tâche, deux différents types de classifieurs sont utilisés. Le premier est un *Gradient Tree Boosting* (*GBT*) (Pedregosa *et al.*, 2011 ; Friedman *et al.*, 2001). Le classifieur *GBT* est une généralisation des algorithmes de *boosting* utilisant une fonction de coût. Ce classifieur est employé comme baseline pour ses performances ainsi que le peu de paramétrage nécessaire. Le second classifieur utilisé est un réseau de neurones artificiel, appelé *Multilayer Perceptron* (*MLP*) (Ruck *et al.*, 1990 ; Bastien *et al.*, 2012), composé de 3 couches dans cette expérience. Une couche d’entrée de 95 neurones artificiels, 32 neurones artificiels dans la couche cachée et 8 neurones dans la couche de sortie avec des sigmoïds puis softmax pour les fonctions d’activation. Enfin, ces réseaux utilisent la méthode de *dropout* pour la régularisation. Les deux classifieurs sont entraînés et évalués séparément avec chacun des 4 modèles Word2vec.

### Résultats

Les précisions reportées dans le tableau 5 sont mesurées à la fois sur le corpus de dev. et le corpus de test avec respectivement les modèles *Skip-gram* et *CBOW*. Ce tableau montre que toutes les configurations testées voient leurs performances s’améliorer en utilisant l’information de distance contextuelle (*CCM*). Nous constatons que les résultats du modèle *Skip-gram* avec le *GBT* augmentent de 10 %, et que les résultats avec le *MLP* augmentent considérablement la précision avec un gain absolu de 20 % et un score maximum de 70 %. De même, les résultats utilisant les modèles *CBOW* voient leurs précisions doubler grâce à la pondération *CCM* que nous proposons. Le classifieur *GBT* obtient un score de 60 % et le *MLP* passe de 31 % à 71 % en termes de précision. Les résultats du *MLP* sont mesurés toutes les 10 époques d’apprentissage à la fois sur le corpus de dev. et le corpus de test. Ces résultats sont représentés dans le graphique 5. Nous remarquons que les modèles utilisant la pondération *CCM* obtiennent de meilleurs résultats et convergent plus rapidement que les autres modèles.

Ces expériences montrent qu’utiliser une pondération *CCM* dans un modèle Word2Vec produit des représentations avec une plus forte influence thématique. Elle

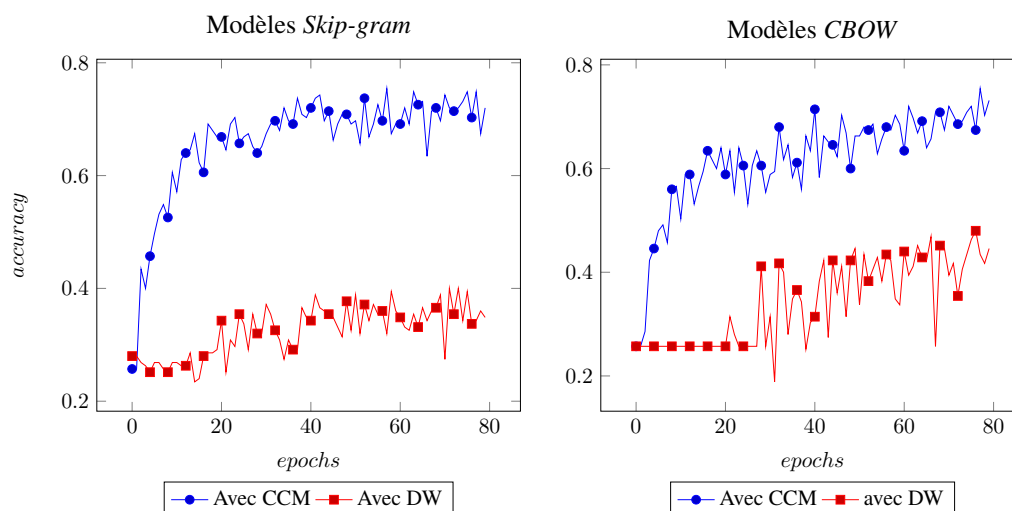


**Figure 5.** Précision du classifieur MLP au fil des époques (0 à 80) pour les architectures Skip-gram et CBOW.

**Tableau 5.** Précision de la classification (%) sur la classification de thèmes en utilisant 2 classifieurs différents (GBT et MLP).

|                     | Skip-gram |           | CBOW |           |
|---------------------|-----------|-----------|------|-----------|
|                     | Dev       | Test      | Dev  | Test      |
| GBT avec DW         | 39        | 42        | 28   | 27        |
| GBT avec CCM        | 56        | 52        | 66   | 60        |
| MLP avec DW         | 50        | 50        | 41   | 37        |
| <b>MLP avec CCM</b> | 75        | <b>70</b> | 74   | <b>71</b> |

montrent aussi que plus le contexte est important, plus l’information contenue dans la pondération *CCM* a un effet bénéfique. De plus, les réseaux de neurones apprenant à projeter les mots dans un espace multidimensionnel ont besoin d’un plus grand nombre de neurones artificiels pour capturer l’information supplémentaire contenue par la pondération *CCM*. Cette information apporte des modèles améliorés, comme le montre le “test sémantique et syntaxique” (cf. section 3.2), et fournit aux outils de classification des caractéristiques plus adaptées pour la classification thématique de données textuelles.

#### 4. Conclusion

La fenêtre de contexte dynamique proposée de manière classique utilise les mots, représentés par des sacs-de-mots binaires, tout en ignorant aléatoirement les mots en bordure de contexte. Dans ce contexte, tous les mots sont traités de façon égale. Cet article propose une alternative originale sous la forme d’une fonction de pondération s’appuyant sur les modèles de contextes continus capable de préserver les relations distantes. Cette méthode a été évaluée au moyen d’un test de similarité syntaxique



et sémantique, où un gain de 7 % a été observé, ainsi qu'une tâche de classification thématique de dialogues apportant un gain de plus de 20 %. Ces expériences ont aussi montré que les modèles intégrant notre proposition de pondération du contexte continu des mots est utile pour la classification thématique de documents textuels. Nous prévoyons d'étendre ce travail en étudiant l'impact de différents types de fonctions de pondération du contexte continu des mots et en ajoutant la même information pour d'autres formes de représentations distribuées.

## 5. Bibliographie

- Bastien F., Lamblin P., Pascanu R., Bergstra J., Goodfellow I. J., Bergeron A., Bouchard N., Bengio Y., « Theano : new features and speed improvements », , Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012.
- Bechet F., Maza B., Bigouroux N., Bazillon T., El-Beze M., De Mori R., Arbillet E., « DECODA : a call-centre human-human spoken conversation corpus. », *LREC*, p. 1343-1347, 2012.
- Bengio S., Heigold G., « Word embeddings for speech recognition », *Proceedings of the 15th Conference of the International Speech Communication Association, Interspeech*, 2014.
- Bengio Y., Ducharme R., Vincent P., « A Neural probabilistic language model », *Journal of Machine Learning Research*, vol. 3, p. 1137-1155, 2003.
- Bigot B., Senay G., Linarès G., Fredouille C., Dufour R., « Combining Acoustic Name Spotting and Continuous Context Models to improve Spoken Person Name Recognition in Speech », *Interspeech*, 2539-2543, 2013.
- Blei D. M., Ng A. Y., Jordan M. I., « Latent dirichlet allocation », *the Journal of machine Learning research*, vol. 3, p. 993-1022, 2003.
- Collobert R., Weston J., Bottou L., Karlen M., Kavukcuoglu K., Kuksa P., « Natural language processing (almost) from scratch », *The Journal of Machine Learning Research*, vol. 12, p. 2493-2537, 2011.
- Do Q.-K., Allauzen A., Yvon F., « Modèles de langue neuronaux : une comparaison de plusieurs stratégies d'apprentissage », *TALN 2014*, 2014.
- Friedman J., Hastie T., Tibshirani R., *The elements of statistical learning*, vol. 1, Springer series in statistics Springer, Berlin, 2001.
- Linarès G., Nocéra P., Massonie D., Matrouf D., « The lia speech recognition system : from 10xrt to 1xrt », *Text, Speech and Dialogue*, Springer, p. 302-308, 2007.
- Mesnil G., Mikolov T., Ranzato M., Bengio Y., « Ensemble of Generative and Discriminative Techniques for Sentiment Analysis of Movie Reviews », 2015.
- Mikolov T., Corrado G., Chen K., Dean J., « Efficient Estimation of Word Representations in Vector Space », *Proceedings of the International Conference on Learning Representations (ICLR 2013)*, p. 1-12, 2013a.
- Mikolov T., Sutskever I., Chen K., Corrado G. S., Dean J., « Distributed representations of words and phrases and their compositionality », *Advances in Neural Information Processing Systems*, p. 3111-3119, 2013b.

- Morchid M., Dufour R., Bouallegue M., Linarès G., « Author-Topic based Representation of Call-Center Conversations », *International Spoken Language Technology Workshop (SLT) 2014*, IEEE, 2014a.
- Morchid M., Dufour R., Bousquet P.-M., Bouallegue M., Linarès G., De Mori R., « Improving Dialogue Classification using a Topic Space Representation and a Gaussian Classifier based on the Decision Rule », *International Conference on Acoustic, Speech and Signal Processing (ICASSP) 2014*, IEEE, 2014b.
- Morchid M., Dufour R., Linarès G., Hamadi Y., « Latent Topic Model based Representations for a Robust Theme Identification of Highly Imperfect Automatic Transcriptions », *International Conference on Intelligent Text Processing and Computational Linguistics (CICLing) 2015*, 2015.
- Pedregosa F., Varoquaux G., Gramfort A., Michel V., Thirion B., Grisel O., Blondel M., Prettenhofer P., Weiss R., Dubourg V., Vanderplas J., Passos A., Cournapeau D., Brucher M., Perrot M., Duchesnay E., « Scikit-learn : Machine Learning in Python », *Journal of Machine Learning Research*, vol. 12, p. 2825-2830, 2011.
- Ruck D. W., Rogers S. K., Kabrisky M., Oxley M. E., Suter B. W., « The multilayer perceptron as an approximation to a Bayes optimal discriminant function », *Neural Networks, IEEE Transactions on*, vol. 1, n° 4, p. 296-298, 1990.
- Sahlgren M., « The distributional hypothesis », *Italian Journal of Linguistics*, vol. 20, n° 1, p. 33-54, 2008.
- Vaswani A., Zhao Y., Fossium V., Chiang D., « Decoding with Large-Scale Neural Language Models Improves Translation. », *EMNLP*, Citeseer, p. 1387-1392, 2013.

Papier absent

Papier absent

Papier absent

Papier absent

Papier absent





---

# Apprentissage d'inférences par édition d'arbres pour répondre à des questions

Martin Gleize\* — Brigitte Grau\*\*

\* LIMSI-CNRS, Université Paris-Sud

\*\* LIMSI-CNRS, ENSIIE

---

**RÉSUMÉ.** La sélection de réponse en recherche d'information précise met nécessairement en oeuvre un appariement de passages avec la question. Nous proposons un algorithme qui consiste à engendrer et apprendre les inférences utiles pour rapprocher les passages de texte à des couples (questions, réponse candidate). Ceux-ci sont sélectionnés au moyen d'une expansion lexicale utilisant WordNet et des vecteurs de mots. Ils sont représentés par leur arbre de dépendances syntaxique, enrichi au moyen de plusieurs ressources lexico-sémantiques dont WordNet et ConceptNet. Un algorithme de recherche en faisceau calcule des séquences d'édition transformant les passages en choix de réponse. Des traits calculés à partir de ces séquences d'édition sont utilisés pour construire deux classifieurs chargés respectivement de valider ou invalider chaque choix de réponse. Cette méthode a été évaluée sur la tâche "Entrance Exams" à CLEF 2015 qui propose de répondre à des QCMs sur des textes et a obtenu le deuxième meilleur score de précision des systèmes participants.

**ABSTRACT.** In order to answer question, we propose a matching algorithm that consists in generating and learning inferences needed to rely text passages to pairs (question, candidate answer). We first retrieve relevant passages, through lexical expansion involving WordNet and word vectors, that are enriched by lexico-semantic resources. Then a tree edit model is used on graph representations of the passages and answer choices to extract edit sequences. Finally, features are computed from those edit sequences and used in two machine-learned models, one for validating answers and one for invalidating answers, in order to take the final decision. This method was evaluated in the Entrance Exams task at QALD@CLEF, that consists in answering QCMs on given texts, yielding the second best precision score on the task.

**MOTS-CLÉS :** système de question-réponse, questions à choix multiples, algorithme d'édition d'arbres, inférence textuelle.

**KEYWORDS:** Question-answering system, Multiple Choice Questions, Tree edit algorithm, textual entailment.

---

## 1. Introduction

Différentes solutions peuvent être mises en oeuvre pour rechercher la réponse à une question. On peut rechercher dans des textes ou bien dans des bases de questions dont on connaît déjà la réponse, dans des FAQ ou sur des sites communautaires. Mais quelle que soit la tâche, on en revient toujours à devoir apparier l'information exprimée par la question avec celle qui est contenue dans un passage de texte. Cet appariement peut s'exprimer comme un problème d'implication textuelle afin de valider, ou sélectionner, la réponse pertinente : le passage sélectionné implique-t-il l'hypothèse formée par la question ?

La sélection de réponse a donné lieu à des méthodes qui s'appuient sur différents niveaux d'analyse de la langue, et produisent de ce fait des représentations différentes des énoncés (questions et passages). Les premières ont défini des mesures de similarité globale sur une représentation de type "sac de mots" (Ferret *et al.*, 2001 ; Magnini *et al.*, 2002 ; Ittycheriah *et al.*, 2001) ou ont formulé la sélection de réponse comme un problème de classification ou de re-ordonnancement (Cui *et al.*, 2005 ; Grappy *et al.*, 2011) appris automatiquement. Afin de tenir compte des relations entre mots des énoncés lors de l'appariement, des modèles d'alignement d'arbres de dépendances (Wang *et al.*, 2007 ; Yao *et al.*, 2013) ou de leur transformation par éditions successives (Kouylekov *et al.*, 2007 ; Heilman et Smith, 2010 ; Moschitti *et al.*, 2007) ont été proposés. Cependant, les connaissances sémantiques, même si elles sont souvent prises en compte, sont ajoutées en dehors de ces modèles d'appariement.

Une véritable intégration de syntaxe et sémantique existe dans quelques travaux, comme le modèle de Wang et Manning (2010), qui définit des opérations d'édition avec inclusion de variations sémantiques sur les mots. Pour aller au delà du niveau lexical et élargir les types d'inférences envisagées, nous proposons d'enrichir les arbres de dépendances par des relations sémantiques entre couples de concepts. Ces relations nous sont fournies par la base de triplets ConceptNet, une ressource de connaissance de sens commun structurée. Un algorithme de recherche en faisceau calcule des séquences d'édition transformant les arbres des passages en ceux des choix de réponse. Des traits calculés à partir de ces séquences d'édition sont utilisés pour construire deux classifieurs chargés respectivement de valider ou invalider chaque choix de réponse.

Afin de rendre l'évaluation plus directe et pertinente, il est possible de fixer ces choix de réponses, en terme de leur nombre et de leur difficulté, dans le cadre de la réponse à des QCM de compréhension de texte. La conférence CLEF 2015 propose la tâche "Entrance Exam". Elle évalue les systèmes sur les mêmes QCM que les étudiants passent à l'entrée à l'université, avec l'hypothèse que différents types d'inférences soient nécessaires pour y répondre. Notre méthode a été évaluée sur cette tâche à CLEF 2015 et a obtenu le deuxième meilleur score avec une précision de 0,36. Ces travaux ont été publiés précédemment en anglais à la conférence sus-citée (Gleize et Grau, 2015).

## 2. État de l'art

Les travaux récents en question-réponse se focalisent sur la sélection d'un passage réponse, étant donné un ensemble de passages candidats. Les approches sacs de mots ont été améliorées sur cette tâche par des modèles qui cherchent à apprendre l'alignement latent entre les deux représentations, celle du passage et celle de la question. Différentes mesures ont été conçues reposant sur un décompte des alignements entre les relations de dépendance des deux fragments de texte. Wang *et al.* (2007) apprennent cet alignement par un modèle probabiliste génératif fondé sur une grammaire quasi-synchrone. Ce formalisme permet de tenir compte de transformations locales, entre des équivalences de syntagmes tels que "leader of France" et "French president". Ces modèles réalisent un alignement mot à mot où un mot de la question est associé à un mot du passage, ce qui ne permet pas de trouver des paraphrases sous-phrastiques de tailles différentes, ce qui a conduit Yao *et al.* (2013) à étendre un CRF par un modèle semi-markovien qui permet d'introduire cette flexibilité.

L'édition d'arbres de dépendances pour une tâche de sélection de passage a été proposée pour la première fois par (Punyakanok *et al.*, 2004). Elle a été intégrée à un système de validation de réponse par (Kouylekov *et al.*, 2007). Heilman et Smith (2010) en ont proposé une version évaluée sur la sélection de phrases réponses, qui apprend un classifieur sur des traits extraits des séquences d'opérations d'édition. Celles-ci sont produites par un algorithme glouton appliquant une fonction noyau sur les arbres pour évaluer leur similarité. Dans (Moschitti *et al.*, 2007), les auteurs proposent une fonction noyau sur les arbres qui étend la définition de sous-arbres similaires. Enfin, le modèle de Wang et Manning (2010) apprend l'alignement latent de deux arbres de dépendances en définissant des opérations d'édition qui incluent des transformations sémantiques, intégrant ainsi syntaxe et sémantique.

Ces modèles sont évalués<sup>1</sup> sur le jeu de test de (Wang *et al.*, 2007) qui contient 89 questions de TREC et où les réponses sont énoncées par une seule phrase.

De ce fait les modèles proposés cherchent à aligner deux représentations de phrases, alors que souvent les informations données dans la question sont réparties dans des passages de plusieurs phrases. Dans (Sachan *et al.*, 2015), les auteurs proposent un modèle d'alignement intégrant, entre autres, cette contrainte. Il a été conçu pour la tâche de compréhension de texte, "Machine Reading", et est évalué sur le corpus MCTest<sup>2</sup> qui est un ensemble de QCMs construit par crowdsourcing.

---

1. Voir [http://aclweb.org/aclwiki/index.php?title=Question\\_Answering\\_State\\_of\\_the\\_art](http://aclweb.org/aclwiki/index.php?title=Question_Answering_State_of_the_art) pour obtenir les scores des différents systèmes

2. <http://research.microsoft.com/mct>

### 3. Présentation générale de la méthode

Le but général de notre système est de choisir la réponse correcte parmi celles qui sont proposées. Voici un extrait de texte, suivi par un item du QCM, composé d'une question et de proposition de réponses dont une seule est correcte.

Texte : It was early morning. Peter Corbett helped Mark Wellman out of his wheelchair and onto the ground. They stood before El Capitan, a huge mass of rock almost three-quarters of a mile high in California's beautiful Yosemite Valley. It had been Mark's dream to climb El Capitan for as long as he could remember. But how could a person without the use of his legs hope to try to climb the highest vertical cliff on earth ? [...]

Question : What had Mark Wellman long desired to do ?

- 1) To accomplish one of the most difficult rock climbs in the world.(correcte)
- 2) To be the first to conquer El Capitan.
- 3) To climb the highest mountain in California.
- 4) To help his friend Peter climb El Capitan.

La réponse correcte provient généralement d'une formulation assez éloignée de ce qui la justifie dans le texte et repose sur la capacité à trouver les inférences nécessaires, inférences qui relèvent de connaissances générales et non de connaissances encyclopédiques ou de connaissances d'un domaine. Par ailleurs, les réponses incorrectes peuvent être partiellement retrouvées dans le texte, et il nous a paru intéressant d'être en mesure d'invalider les réponses fausses. Généralement les systèmes cherchent à valider des réponses, i.e. trouver les caractéristiques permettant de rapprocher deux extraits de textes, et non à invalider des réponses, i.e. trouver des caractéristiques qui rendent compte d'un changement de sens entre deux passages.

Notre objectif est de valider des réponses correctes sans les invalider, et inversement d'invalider des réponses fausses, sans les valider, en référence à un passage support. La sélection de ces passages est effectuée en rapport avec la question. En effet, en étudiant les textes, il nous est paru que la question amène à délimiter un espace de recherche dans lequel les candidats réponse, corrects ou non, peuvent être analysés.

Le système que nous proposons est donc composé de quatre modules (cf. figure 1) : recherche de passages, enrichissement sémantique des graphes de dépendances, édition de graphes par une recherche en faisceau, sélection de réponse reposant sur l'apprentissage de classifieurs pour valider/invalider des réponses.

### 4. Sélection de passages

La sélection de passage répond à deux objectifs conjoints : réduire l'espace de recherche de manière à pouvoir appliquer des processus coûteux en temps, et sélectionner un contexte en relation avec le couple (question, réponse candidate) pour justifier

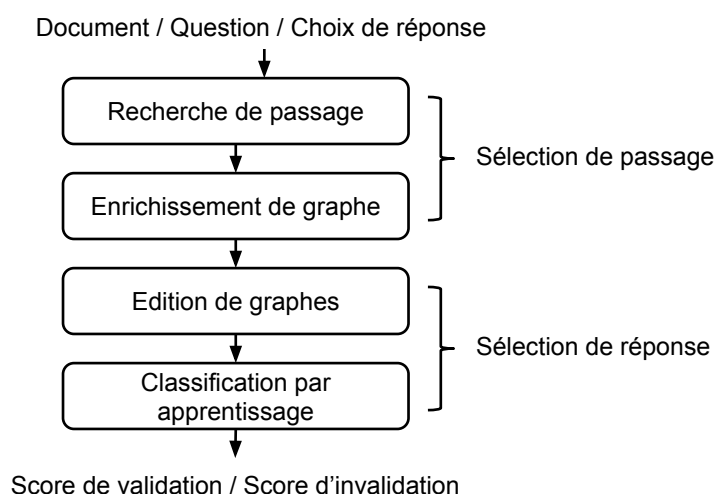


**Figure 1.** *Architecture du système*

ou non une réponse. L'objectif est de sélectionner des passages qui rendent compte au mieux du sens du couple. C'est pourquoi nous favorisons les passages qui permettent un alignement avec le maximum de mots du couple requête.

#### **4.1. Recherche de passages**

Les mots du couple (question, réponse candidate) forment la requête de base, sous forme de lemmes. Comme il est rare qu'ils soient présents tel quels dans le texte, nous appliquons une forme d'expansion. Nous enrichissons chaque lemme avec les informations de coréférence et les lemmes en relation sémantique avec lui dans WordNet (synonymes, antonymes, hyperonymes, hyponymes). Si un mot ne peut toujours pas être relié au texte, nous calculons une mesure de similarité fondée sur les représentations distribuées construites par (Huang *et al.*, 2012) et nous associons un vecteur de 50 traits à chacun des lemmes des mots du texte et de la requête selon la méthode proposée par l'auteur. Nous apparions les mots de la requête avec les mots du passage ayant la plus forte similarité par la fonction *cosinus*. Nous tenons compte des bi-termes en les représentant par la somme de leurs vecteurs, ce qui nous permet des alignements 1-2, 2-1 et 2-2.

Les passages sont délimitées par une fenêtre glissante de taille fixe<sup>3</sup> et sont ordonnés en fonction de leur score, calculé par la formule 1. Ils sont ensuite étendus aux frontières de phrases.

3. La taille choisit permet de capturer 3 phrases environ, nombre qui résulte d'une étude du corpus

$$score(passage) = \frac{\#matchedWords}{\#queryWords} \times \sum_{i=1}^{n-1} \frac{score(w_i) + score(w_{i+1})}{dist(i, i+1)^2} \quad [1]$$

Chaque mot du passage  $w_i \in \{w_1, \dots, w_n\}$  apparié avec un mot de la requête reçoit un score d'alignement  $score(w_i)$  (1 si les lemmes sont égaux, 0,9 si ils sont synonymes dans WordNet, 0,8 pour les autres relations de WordNet et sinon le score de similarité de leur vecteurs, pondéré par leur ISF (Inverse Sentence Frequency)). La formule est divisée par le carré de la distance des mots trouvés dans le passage afin de favoriser les passages les plus compacts. Pour tenir compte de l'absence potentielle de mots de la requête, nous multiplions le score du passage par la proportion de mots de la requête qui lui sont reliés.

Cette méthode retrouve un grand nombre de courts passages, souvent en position de recouvrement, mais l'algorithme en faisceau que nous appliquons ensuite pour les analyser permet de traiter de nombreux passages.

## 4.2. Représentation et enrichissement des passages

Les passages sont représentés par la structure syntaxique des phrases qui les composent, i.e. l'arbre de dépendances, reliées suivant leur séquence dans le passage. Ils sont ensuite enrichis par des connaissances permettant de réaliser des inférences. Le but de cet enrichissement est de pouvoir appliquer les règles d'édition sur les différents types de connaissances et opérer ainsi des inférences syntaxiques ou sémantiques en se reposant sur la même méthode.

Avant de décrire l'enrichissement des graphes, nous allons préciser les pré-traitements appliqués au texte.

### 4.2.1. Prétraitement des textes

Nous utilisons la suite CoreNLP de Stanford pour analyser les textes. Chaque phrase du texte, les questions et les réponses sont annotées par les catégories morpho-syntaxique des mots (Toutanova *et al.*, 2003). Les graphes de dépendance sont engendrés par l'application de l'analyseur de (Klein et Manning, 2003). Un outil de résolution de co-références (Recasens *et al.*, 2013) est appliqué sur le texte et sur les item des QCMs. En effet, les textes que nous traitons sont souvent de courts récits, faisant usage de nombreux pronoms personnels. De plus, tous les types de textes font usage d'anaphores nominales. On trouve souvent des références à un personnage sous différentes appellations générales. Une particularité des QCM consiste à employer les termes "the author" ou "the writer" dans les items (questions et réponses) quand le texte est rédigé à la première personne. Nous ajoutons de ce fait un traitement spécifique de ces tournures afin d'unifier les deux vocables. Notons que nous n'avons pas appliqué de module de reconnaissance d'entité nommées, les textes en comportant peu, et, en général, les questions posées n'attendent pas une réponse de ce type.

#### 4.2.2. Enrichissement des graphes

Afin de construire un graphe pour le passage, les racines des graphes de dépendances des phrases sont reliés avec un arc *followed-by* qui matérialise le fait qu'une phrase en suit une autre, et maintient la cohérence du passage.

L'enrichissement est fondé sur la ressource ConceptNet (Liu et Singh, 2004). ConceptNet est une base de triplets sémantiques construite manuellement qui comporte des connaissances générales de sens commun, destinée à être utilisée pour la compréhension automatique de textes. Les noeuds d'un triplet sont des mots ou des expressions courtes de la langue, et la relation entre eux est étiquetée. Les noeuds sont dénommés "concept" mais il est plus juste de les considérer comme des termes. Par exemple, ConceptNet contient des connaissances de base assez communes comme *MotivatedByGoal(learn, knowledge)* : on apprend dans le but d'acquérir des connaissances, ou *UsedFor(saxophone, jazz)* : un saxophone est utilisé pour le jazz. D'autres bases de triplet plus connues comme DBpedia ou YAGO se concentrent sur les entités nommées, qui ne sont ici qu'un problème rare.

Notre hypothèse est que ce type de relation d'inférence entre termes correspond aux connaissances mobilisées pour comprendre les textes présents dans les QCM. Ces connaissances de sens commun sont généralement maîtrisées par les êtres humains, mais sont difficiles à saisir par une machine. C'est pourquoi nous avons voulu enrichir les graphes des passages par ces relations, de manière à pouvoir combler les pas de raisonnement manquant pour passer d'un mot à un autre, ou d'un sous-graphe à un autre.

Les *Concepts* de ConceptNet sont principalement nommés par des mots uniques, comme "saxophone" ou "jazz". Il est donc aisé de relier ses triplets à nos graphes, ce lien se faisant sur les noeuds communs. Il n'est cependant pas aussi simple d'intégrer les relations, car leur étiquette est généralement formée de plusieurs mots, comme "UsedFor", "MotivatedByGoal", et nous voulons obtenir une représentation unifiée des passages. C'est pourquoi nous avons rattaché aux graphes des phrases les arbres d'analyse des formes de surface associées à la relation. Ces formes de surface sont les phrases originales en langue naturelle qui expriment la relation, comme "a saxophone is used for jazz". Nous rattachons l'arbre d'analyse de ces expressions chaque fois qu'un concept possède un mot de tête présent dans le graphe de dépendance du passage. Nous avons retenu de ConceptNet les relations exprimant une implication textuelle, soit : *IsA*, *PartOf*, *MemberOf*, *UsedFor*, *CapableOf*, *Causes*, *HasPrerequisite*, *MotivatedByGoal*, *Desires*.

## 5. Sélection de réponse

Notre objectif est de caractériser une séquence de transformations appliquées au passage permettant d'obtenir une réponse candidate. Ces transformations portent sur le graphe enrichi du passage, et les transformations en sont des éditions ce qui nous a conduit à proposer un algorithme d'édition d'arbre. Les opérations d'édition appli-

quées permettront ensuite de fournir des traits à deux classifieurs, l'un pour valider des réponses, l'autre pour invalider des réponses.

## 5.1. Edition de graphes par une recherche en faisceau

### 5.1.1. Opérations d'édition

Le principe de l'algorithme consiste à appliquer au graphe différentes opérations d'édition itérativement, le modifiant ainsi à chaque itération, de manière à ce que le graphe soit de plus en plus proche de l'arbre cible représentant l'hypothèse à atteindre, i.e. une réponse candidate. La liste d'édérations consécutives constitue une *séquence d'édition*. La table 1 présente les opérations d'édition choisies. Les opérations "delete", "insert", "rename" sont les opérations minimales d'un algorithme d'édition. L'opération "move" est plus inhabituelle car elle permet d'effectuer d'importants changements dans la phrase juste en rattachant un noeud à un nouveau noeud père. Si ce noeud est la racine d'un sous-arbre ayant des descendants, cela fait déplacer un large extrait de texte. Les opérations n'utilisent pas les étiquettes des relations de dépendance, mais celles-ci seront utilisées dans les traits extraits.

La figure 2 présente un exemple de l'application consécutive de trois opérations.

| Opération d'édition                         | Description                                       |
|---------------------------------------------|---------------------------------------------------|
| Delete( $d$ : Tree)                         | Supprime le noeud $d$ et le remplace par son fils |
| Insert( $i$ : Word, $p$ : Tree)             | Insère le mot $i$ sous son nouveau père $p$ .     |
| Rename( $t$ : Tree, $w$ : Word)             | Remplace le mot attaché au noeud $t$ par $w$ .    |
| Move( $m$ : Tree, $op$ : Tree, $np$ : Tree) | Déplace le sous-arbre $m$ sous $op$ sous $np$ .   |

**Tableau 1.** Opérations d'édition d'arbre de dépendance

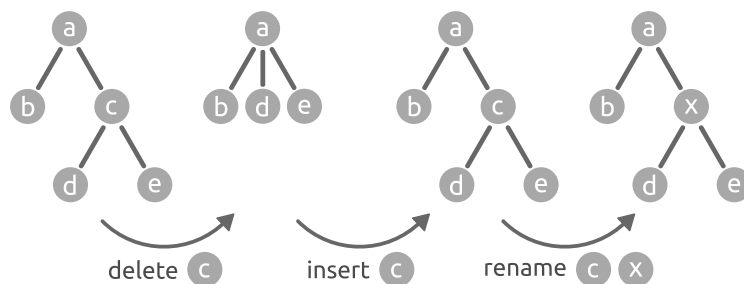


**Figure 2.** Exemple d'application successive d'opérations d'édition

L'exploration des éditions à appliquer se ramène à un problème de recherche de chemin dans un graphe. Cependant il n'est pas possible d'appliquer toutes les transformations possibles à chaque itération, pour des problèmes de complexité en temps et en espace. Aussi il est nécessaire de savoir sélectionner les plus prometteuses.



### 5.1.2. Recherche en faisceau

Le problème principal de l'algorithme calculant les éditions est de savoir quand appliquer quelle opération d'édition, savoir quel élément ajouter ? Nous appliquons des heuristiques simples pour limiter le nombre d'opérations à considérer (par exemple ne pas insérer de noeud qui n'aide pas à atteindre la cible, ne pas renommer ou supprimer un noeud qui est présent dans l'arbre cible). Cependant un grand nombre d'opérations restent possibles. Aussi, au lieu de choisir à chaque pas les opérations à appliquer, nous les appliquons toutes et choisissons ensuite de développer les plus prometteuses en nous appuyant sur une bonne heuristique.

La *recherche en faisceau* opère une recherche en largeur d'abord, ce qui minimise l'espace mémoire requis. A chaque pas, toutes les éditions sont appliquées et les arbres obtenus sont ordonnés selon un coût heuristique. Cependant seul un nombre limité d'arbres est conservé à chaque étape, qui correspond à la largeur du faisceau. Cette méthode permet d'ajuster, *via* ce paramètre, la probabilité de trouver des séquences d'édition utiles et les coûts en temps et en espace.

L'heuristique doit permettre d'estimer la proximité du noeud exploré avec la cible. Nous avons implémenté à cette fin la fonction noyau sur les arbres partiels définie dans (Moschitti, 2006) pour calculer la similarité entre l'arbre courant et l'arbre cible. Comme toute fonction noyau d'arbre, elle calcule le nombre de sous-arbres communs entre les deux arbres mais cette version s'applique à des arbres  $n$ -aires, ce que sont les arbres de dépendance. Le calcul du noyau est normalisé par  $\tilde{K}(x, y) = \frac{K(x, y)}{\sqrt{K(x, x)}\sqrt{K(y, y)}}$ , pour  $K$  le noyau et  $x, y$  les arbres.

### 5.1.3. Algorithme

L'algorithme développe un graphe représentant une partie de l'espace de recherche, où chaque noeud est un arbre et les arcs sont les opérations d'édition appliquées. Au début de l'algorithme, ce graphe est initialisé par les  $p$  arbres des passages sélectionnés qui ont été enrichis. L'arbre cible est celui de la réponse lorsqu'elle est sous forme de phrase ou celui de la question plus la réponse lorsque la réponse est la fin d'une phrase question. Ensuite chaque opération d'édition est appliquée à chaque passage :

- "Insert" et "Rename" ne peuvent ajouter que des noeuds qui existent dans l'arbre cible ;
- Move peut uniquement déplacer un noeud sous un noeud père tel que le lien parent  $\rightarrow$  enfant est présent dans l'arbre cible.

Les arbres modifiés sont ajoutés au graphe, et l'heuristique est calculée pour chacun d'eux, permettant de les ordonner. Les  $n$  meilleurs sont gardés et l'algorithme est réitéré. Il s'arrête quand  $k$  séquences d'édition différentes ont été trouvées ou après  $l$  itérations. Dans nos expérimentations, nous avons retenu :  $p = 10$ ,  $n = 50$ ,  $k = 10$  et  $l = 200$ .

## 5.2. Validation et invalidation de réponses

L'objectif est de classer les séquences d'édition par l'apprentissage de deux classifieurs : l'un pour décider si la séquence permet de valider une réponse correcte, l'autre pour décider si elle permet d'invalider une réponse fausse. Il s'agit donc d'en extraire des traits caractéristiques qui permettent ces classifications. En pratique, les traits et les algorithmes sont les mêmes pour apprendre les deux classifieurs. Seuls les corpus d'apprentissage sont différents.

### 5.2.1. Extraction des traits

La plupart des traits dénombrent des opérations d'édérations particulières dans la séquence, sur des unigrammes ou des bigrammes, et sont résumés table 2. Les informations présentes dans les arbres de dépendances du passage et non utilisées lors de la recherche en faisceau sont utilisées à cette étape : les relations de dépendances, les co-références ou le fait qu'une édition s'applique sur un concept de ConceptNet ou de WordNet.

### 5.2.2. Apprentissage des classifieurs

Les deux modèles, pour la validation et l'invalidation de réponse candidate<sup>4</sup>, sont appris à partir des mêmes traits mais sur des corpus d'entraînement différents. Nous voulons éviter d'apprendre comment transformer un passage de texte pris au hasard en une réponse prise au hasard aussi. En tant que lecteur, on ne peut valider une réponse en lisant un quelconque passage du texte, ni en rejeter une si le passage support n'est en rien relié à la question et à la réponse. Aussi, nous avons annoté manuellement la pertinence de passages relativement à un couple (question, réponse) afin de construire les corpus d'apprentissage. Un *passage pertinent* est défini comme l'extrait minimum qui permet d'exprimer à la fois la question et les éléments de réponse, i.e. de justifier la réponse si elle est correcte ou d'invalider celle-ci. Les réponses incorrectes ne sont bien sûr généralement pas exprimées dans leur totalité. Parfois même, une réponse n'est pas exprimée du tout dans le texte. Deux réponses liées à la même question peuvent partager le même passage justificatif.

Nous construisons les corpus d'apprentissage en suivant la sémantique décrite figure 3. Un passage est associé à un choix de réponse ; aussi passage *choixX* fait référence au passage associé à n'importe quel choix de réponse incorrecte (X) et passage *choix1* à une réponse incorrecte particulière. Pour la validation de réponse, les exemples positifs sont les couples (passage, réponse correcte). Pour l'invalidation de réponse, les exemple positifs sont les couples (passage correct, réponse incorrecte) et (passage pertinent pour la réponse incorrecte considérée, réponse considérée), et ce pour chacune des réponses incorrectes. Les autres combinaisons forment les exemples négatifs pour les deux classifieurs. Les couples positifs pour un classifieur donnent lieu à des couples négatifs pour l'autre. Ces combinaisons, surtout pour les réponses

4. Nous appelons réponse les choix de réponses, i.e. les réponses candidates.

| Trait                                                                                                                                            | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|--------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| editTotal                                                                                                                                        | Nombre total d'édérations dans la séquence                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| deleteTotal<br>deleteVerb<br>deleteNoun<br>deleteProperNoun<br>deleteSubject<br>deleteObject<br>deleteRoot<br>deleteNegation<br>deleteConceptNet | Nombre total d'opérations de suppression, édition qui supprime un verbe, un nom, un nom propre, un sujet (indiqué par la relation de dépendance subj), un complément d'objet, la racine d'un arbre, une négation (indiquée par la relation de dépendance neg) et un élément ajouté au graphe à partir de ConceptNet                                                                                                                                                                  |
| insertTotal<br>insertVerb<br>insertNoun<br>insertProperNoun<br>insertNegation                                                                    | Analogue aux informations ci-dessus, pour les opérations d'insertion                                                                                                                                                                                                                                                                                                                                                                                                                 |
| renameTotal<br>renameVerb<br>...<br>renameSyn<br>renameAnt<br>renameHypHyp<br>renameStrongWordVectorSim<br>renameCoref<br>renameNonCoref         | Analogue aux informations ci-dessus, pour les opérations de renommage + les éditions qui renomme un mot en son synonyme, ou son antonyme ou son hyperonyme/hyponyme dans WordNet, édition qui remplace un mot par un autre trouvé fortement similaire selon leurs vecteurs (au dessus d'un seuil fixé empiriquement), éditions qui renomment un pronom en son référent d'après la résolution des coréférence de Stanford, et éditions qui renomment un pronom par d'autres référents |
| moveTotal<br>moveVerb<br>...<br>moveConceptNet<br>moveMoreThan2Nodes                                                                             | Analogue aux informations ci-dessus, pour les opérations de déplacement + les éditions qui déplacent plus de 2 noeuds                                                                                                                                                                                                                                                                                                                                                                |
| Combinaisons sur les bigrammes                                                                                                                   | Nombre de paires d'édérations successives dans la séquence                                                                                                                                                                                                                                                                                                                                                                                                                           |
| dependencyEditSequence                                                                                                                           | Nombre de paires d'édérations successives appliquées à 2 noeuds d'une relation de dépendance                                                                                                                                                                                                                                                                                                                                                                                         |
| originalTotal<br>originalVerb<br>...                                                                                                             | Proportion de mots, verbes, noms, noms propres originaux qui <b>ne sont pas</b> édités dans la séquence                                                                                                                                                                                                                                                                                                                                                                              |

**Tableau 2.** Traits extraits d'une séquence d'édition

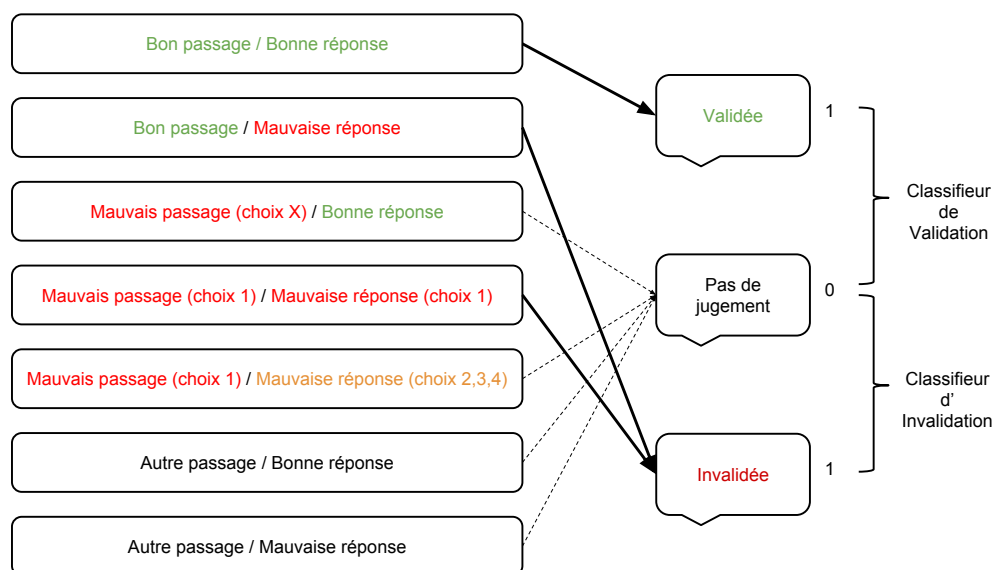


**Figure 3.** *Semantique des paires (passage, réponse)*

incorrectes, signifient que le passage choisi est suffisant pour faire état d'une incohérence de la réponse par rapport à la question.

## 6. Expérimentations et résultats

### 6.1. Tâche "Entrance Exams" à QALD@CLEF 2015 : données et évaluation

Nos données sont celles distribuées pour la tâche "Entrance Exams" à CLEF 2015. Elles sont extraites des examens d'entrée à l'université de Tokyo, pour tester la maîtrise de l'anglais langue étrangère. Le jeu d'entraînement est composé des jeux de test 2013 et 2014. Chacun comporte 12 textes, et 4 à 6 items par QCM, soit environ 120 questions auxquelles il faut répondre. Le test de 2015 comporte 19 textes et un total de 89 questions. Il y a 4 choix de réponse par item.

Les systèmes sont évalués en terme de précision : la fraction de réponses correctes.

### 6.2. Sélection d'une réponse

Les algorithmes renvoient des scores pour les séquences d'édition retenues par couple (question, réponse), soit un score de validation et un score d'invalidation, et nous retenons la séquence ayant la plus forte valeur parmi les  $\max(\text{validationScore}, \text{invalidationScore})$ . Idéalement nous voudrions obtenir une séquence d'édition caractéristique d'un processus de validation ou invalidation fiable de manière à pouvoir classer la réponse comme correcte ou incor-

recte. Ensuite la réponse dont la séquence d'édition a la plus grande valeur de `validationScore` – `invalidationScore` est choisie : nous choisissons comme réponse celle pour laquelle la séparation entre validation et invalidation est la plus nette.

### 6.3. Résultats

Nous avons expérimenté deux modèles de classification, la régression logistique et les forêts d'arbre aléatoires, implémentées dans Weka (Hall *et al.*, 2009). La table 3 montre les résultats obtenus pour chaque modèle, et les résultats obtenus par les autres participants à l'évaluation. Une précision de 0,36 nous place en 2ème position avec le modèle des forêts aléatoires. La baseline est 0,25 et correspond à répondre au hasard. Le premier système repose sur l'utilisation de processus et de connaissances propriétaires (Laurent *et al.*, 2015).

|                       | # questions correctes | # questions erronées | # tests avec précision $\geq 0.5$ | précision |
|-----------------------|-----------------------|----------------------|-----------------------------------|-----------|
| Forêt aléatoire       | 32                    | 57                   | 8                                 | 0,36      |
| Régression logistique | 28                    | 61                   | 4                                 | 0,31      |
| Synapse               | 52                    | 37                   | 16                                | 0,58      |
| cicnlp                | 27                    | 62                   | 6                                 | 0,30      |
| NTUNLG                | 26                    | 63                   | 6                                 | 0,29      |
| CoMiC                 | 26                    | 63                   | 5                                 | 0,29      |

**Tableau 3.** Résultats

### 6.4. Analyse d'erreurs

#### 6.4.1. Analyse quantitative

La tendance générale de notre système est de se comporter mieux quand les séquences d'édition sont courtes, avec une précision de 40% quand les séquences choisies sont de longueur  $< 6$  en moyenne.

Nous avons aussi effectué des tests concernant les ressources utilisées pour enrichir les graphes, WordNet et ConcepNet. Nous avons essayé d'enrichir aussi les réponses, dans le but de permettre des inférences d'ordre 2 (un mot est lié à un autre *via* un mot intermédiaire). Mais cela a dégradé les performances. Nous avons aussi effectué des tests d'ablation, d'abord ConcepNet puis WordNet. Enlever l'une ou l'autre ressource montre une légère baisse, mais pas autant que l'on pourrait l'espérer. Des études plus poussées sont nécessaires pour conclure.

#### 6.4.2. Analyse qualitative

Il est difficile d'effectuer une réelle analyse qualitative aussi nous allons juste donner quelques exemples (passage, réponse fournie) illustratifs de la tâche.

Dans le passage suivant, le système est trompé par la réponse 3, proche du texte, et ConceptNet ne permet pas de relier "held" à "trapped", et "its original nature" au passage (le lien figure plus loin dans le texte).

Several years ago, certain scientists developed a way of investigating the nature of the atmosphere of the past by studying air caught in the ice around the North or South Pole. According to their theory, when snow falls, air is trapped between the snowflakes. The snow turns to ice with the air still inside.

Certain scientists claimed that

- 1) atmospheric gases increase the yearly amount of snow
- 2) falling snowflakes change the chemical balance of the air
- 3) the action of atmospheric gases causes snow to turn into ice
- 4) the air held between snowflakes keeps its original nature (correct)

Dans l'exemple suivant notre système choisit la réponse 3 au lieu de 1 qui aurait pu l'être si "wrong" avait pu être lié à "mistake". Mais dans ConceptNet cette relation est du type *RelatedTo*, que nous n'avons pas retenue car trop imprécise.

Everyone stared. That was embarrassing enough, but it was worse when I finished my coffee and got ready to leave. My face went red - as red as his hair - when I realized I'd made a mistake.

The woman's face turned red

- 1) because she realized that she had been quite wrong about the boy (correct)
- 2) because she realized that the boy was poor and hungry
- 3) because she saw everyone staring at her
- 4) because she hated being shouted at

Dans ces deux cas, une meilleure caractérisation du passage correct aurait été utile, car dans le premier nous omettons la phrase qui contient la réponse correcte et dans le second le passage sélectionné apparaît bien avant le passage lié à la question+réponse correcte. Mais nous n'avons pas voulu intégrer le fait que généralement l'ordre des questions des QCM suit l'ordre des phrases, dans la mesure où nous sommes intéressés par décider de l'adéquation d'un passage sélectionné pour valider ou non une réponse.

Enfin le dernier exemple montre une réponse correcte à laquelle le système répond majoritairement par invalidation. Les réponses 1 et 4 sont nettement invalidées, grâce respectivement à la présence de la négation et de la 1ère phrase exprimant le contraire. La réponse 2 est moins nette, mais avec une tendance pour l'invalidation. La réponse 3 est finalement choisie par défaut, car elle n'est ni validée ni invalidée.

Kate was an energetic woman who expected people always to be doing something, and she found plenty of jobs for Fred to do. This made him feel part of the household, but now he really wanted to be able to sit and reflect on the events of his life. If he had continued to live alone, he would have had the time to do this to his heart's content. One afternoon he felt he simply had to get away from the house. "I'm going for a walk," he said, closing the door behind him. Leaving the town, he walked across the fields and followed a slow-moving stream toward the hills. After a while he came to a pool in the stream under some trees. Here, he thought, was a place he could come to when he needed to reflect on the past. Although the stream seemed unlikely to have any fish, he would simply tell Kate he had found a place to go fishing. When he mentioned the stream that night, his son-in-law, Jim, said in disbelief, "There aren't any fish there. That stream runs dry half the summer."

Why did Fred tell Kate that he had found a place to go fishing ?

- 1) He didn't feel part of the household with Kate and Jim.
- 2) He enjoyed fishing very much and was glad to be able to do it again.
- 3) He wanted a way to leave the house without hurting Kate's feelings.
- 4) He was bored in the house because there were few things to do.

## 7. Conclusion

Nous avons proposé un algorithme permettant d'apparier deux textes afin de décider si l'un implique l'autre. Dans le but de tenir compte de variations syntaxiques et sémantiques, nous avons représenté le texte à apparier par un graphe d'arbres de dépendances, enrichi par des informations sémantiques issues de WordNet, de représentations distribuées et d'arbres issus de ConceptNet pour tenter de réaliser des inférences autres que de la similarité sémantique de mots.

Nous avons intégré cet algorithme dans un système destiné à valider et invalider des réponses afin de répondre à des questions de QCM dans le cadre de l'évaluation CLEF "Entrance Exams". Nous avons obtenu la 2ème place à cette évaluation en 2015.

Dans le futur nous voulons améliorer l'enrichissement de notre graphe, car cette méthode permet de traiter de manière unifiée les variations linguistiques de toute nature, mais la couverture actuelle des ressources n'est pas encore suffisante. Nous envisageons d'y ajouter des paraphrases. Nous envisageons aussi de développer un meta-classifieur pour prendre la décision finale par question.

## 8. Bibliographie

- Cui H., Sun R., Li K., Kan M.-Y., Chua T.-S., « Question answering passage retrieval using dependency relations », *SIGIR*, 2005.
- Ferret O., Grau B., Hurault-Plantet M., Illouz G., Jacquemin C., « Document selection refinement based on linguistic features for QALC, a question answering system », *RANLP*, 2001.

- Gleize M., Grau B., « LIMSI-CNRS@CLEF 2015 : Tree Edit Beam Search for Multiple Choice Question Answering », *Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum, Toulouse, France, September 8-11, 2015.*, 2015.
- Grappy A., Grau B., Falco M.-H., Ligozat A.-L., Robba I., Vilnat A., « Selecting answers to questions from Web documents by a robust validation process », *WI*, 2011.
- Hall M., Frank E., Holmes G., Pfahringer B., Reutemann P., Witten I. H., « The WEKA data mining software : an update », *ACM SIGKDD explorations newsletter*, vol. 11, n° 1, p. 10-18, 2009.
- Heilman M., Smith N. A., « Tree edit models for recognizing textual entailments, paraphrases, and answers to questions », *NAACL*, 2010.
- Huang E. H., Socher R., Manning C. D., Ng A. Y., « Improving word representations via global context and multiple word prototypes », *ACL*, p. 873-882, 2012.
- Ittycheriah A., Franz M., Roukos S., « IBM's Statistical Question Answering System-TREC-10. », *TREC*, 2001.
- Klein D., Manning C. D., « Accurate unlexicalized parsing », *ACL*, p. 423-430, 2003.
- Kouylekov M., Negri M., Magnini B., Coppola B., « Towards entailment-based question answering : ITC-irst at CLEF 2006 », *Evaluation of Multilingual and Multi-modal Information Retrieval*, Springer, p. 526-536, 2007.
- Laurent D., Chardon B., Nègre S., Pradel C., Séguéla P., « Reading comprehension at Entrance Exams 2015 », *CLEF 2015 Working Notes*, 2015.
- Liu H., Singh P., « ConceptNet—a practical commonsense reasoning tool-kit », *BT technology journal*, vol. 22, n° 4, p. 211-226, 2004.
- Magnini B., Negri M., Prevete R., Tanev H., « Mining Knowledge from Repeated Co-Occurrences : DIOGENE at TREC 2002. », *TREC*, 2002.
- Moschitti A., « Efficient convolution kernels for dependency and constituent syntactic trees », *Machine Learning : ECML 2006*, Springer, p. 318-329, 2006.
- Moschitti A., Quarteroni S., Basili R., Manandhar S., « Exploiting syntactic and shallow semantic kernels for question answer classification », *ACL*, 2007.
- Punyakanok V., Roth D., Yih W.-t., « Mapping dependencies trees : An application to question answering », *Proceedings of AI&Math 2004*, p. 1-10, 2004.
- Recasens M., de Marneffe M.-C., Potts C., « The life and death of discourse entities : Identifying singleton mentions », *NAACL-HLT*, p. 627-633, 2013.
- Sachan M., Dubey A., Xing E. P., Richardson M., « Learning Answer-Entailing Structures for Machine Comprehension », *ACL*, 2015.
- Toutanova K., Klein D., Manning C. D., Singer Y., « Feature-rich part-of-speech tagging with a cyclic dependency network », *ACL-HLT*, Association for Computational Linguistics, p. 173-180, 2003.
- Wang M., Manning C. D., « Probabilistic tree-edit models with structured latent variables for textual entailment and question answering », *COLING*, 2010.
- Wang M., Smith N. A., Mitamura T., « What is the Jeopardy Model ? A Quasi-Synchronous Grammar for QA. », *EMNLP-CoNLL*, vol. 7, p. 22-32, 2007.
- Yao X., Van Durme B., Callison-Burch C., Clark P., « Semi-Markov Phrase-Based Monolingual Alignment. », *EMNLP*, p. 590-600, 2013.



Papier absent

Vers une approche utilisant l'apprentissage de métrique pour du  
clustering semi-supervisé interactif d'images (CIFED)

Papier absent

*Viet Minh Vu, Hien Phuong Lai, Muriel Visani*

## Papier absent

Vers une approche utilisant l'apprentissage de métrique pour du  
clustering semi-supervisé interactif d'images (CIFED)

Papier absent

*Viet Minh Vu, Hien Phuong Lai, Muriel Visani*

Papier absent



---

# Apprentissage de Modèles de Langue Neuronaux pour la Recherche d'Information

Nicolas Despres\* — Sylvain Lamprier\* — Benjamin Piwowarski\*

\* Sorbonne Universités, UPMC Univ Paris 06, CNRS, LIP6 UMR 7606, 4 place Jussieu 75005 Paris. Email: prenom.nom@lip6.fr

---

**RÉSUMÉ.** La recherche d'information (RI) ad-hoc se heurte à différentes difficultés, notamment liées à des discordances de vocabulaire entre requête et documents, ainsi qu'à la prise en compte de dépendances séquentielles entre les termes de la requête. Les récents modèles de langue neuronaux sont capables de capturer différents types de dépendances, grâce à une représentation distribuée des mots, mais nécessitent de gros volumes de données pour être entraînés efficacement. Jusqu'alors, ces modèles n'ont pas été utilisés directement pour des tâches de RI classiques, pour lesquelles l'estimation d'un modèle de langue pour chaque document est requise. Nous proposons une approche basée sur des transformations spécifiques (à chaque document) d'une représentation générique (apprise sur l'ensemble du corpus), pour définir des modèles de langue neuronaux pour la RI ad-hoc.

**ABSTRACT.** Information Retrieval (IR) faces different difficulties, notably those related to vocabulary mismatch issues and term dependencies. In the last few years, language models based on neural networks have been proposed to deal with both term dependencies and vocabulary mismatch issues in complex natural language processing tasks. However, to be efficient, these models require huge amounts of training data. They have thus never been employed for IR ad-hoc tasks directly, where the estimation of one language model per document is required. We propose an approach based on the specialization of a generic language model, learned on the whole document collection, by a set of document-specific parameters, to define neural language models fitted for ad-hoc IR tasks.

**MOTS-CLÉS :** Apprentissage de Représentation, Recherche d'Information, Réseau de Neurones.

**KEYWORDS:** Representation Learning, Information Retrieval, Neural Network.

---

## 1. Introduction

Pour améliorer l'efficacité des recherches, la Recherche d'Informations (RI) vise depuis longtemps à prendre en compte les longues dépendances entre les termes et à aborder les questions de discordance de vocabulaire. Ces deux problèmes ont été abordés par différentes approches, des plus empiriques aux plus théoriques, mais aucune de celles proposées jusqu'ici ne résolvent les deux problèmes *à la fois*. Cet article propose une approche basée sur les récentes avancées dans les modèles de langue neuronaux.

La prise en compte des dépendances entre les termes de la requête (comme les mots composés, par exemple) permet d'augmenter la précision d'un système de RI. Plusieurs modèles probabilistes ont été proposés (Fagan, 1987 ; Song et Croft, 1999 ; Metzler et Croft, 2005), mais, dans tous les cas, le problème se résume à calculer des estimations précises des modèles de langue n-grammes (ou l'une de ses variantes), à savoir des modèles de langue où la distribution de probabilité d'un terme dépend d'une séquence finie de termes le précédant.

La prise en compte des relations sémantiques (comme la synonymie) augmente le rappel en permettant de retrouver des documents pertinents qui ne contiennent pas exactement les termes de la requête, mais utilisent des termes sémantiquement liés. Ceci est particulièrement important en raison de l'asymétrie entre les documents et les requêtes. Deux approches différentes sont principalement utilisées pour faire face à ce problème. La première consiste à utiliser le (pseudo) retour de pertinence pour ajouter à la requête des termes qui ne figuraient pas initialement. La seconde utilise des modèles distribués, comme l'indexation sémantique latente (Deerwester, Scott *et al.*, 1990), où les termes et les documents sont représentés dans un espace latent, qui peut être probabiliste ou vectoriel. Cependant, aucune de ces approches ne prend en compte les dépendances entre les termes.

Dans cet article, nous montrons que les modèles de langue neuronaux, qui exploitent une représentation distribuée des mots dans un espace continu, gèrent naturellement les dépendances entre termes, et ont donc un potentiel intéressant en RI. Les modèles de langue neuronaux (Bengio *et al.*, 2003) ont été utilisés avec succès dans de nombreuses tâches de traitement du langage naturel. Leur atout principal réside dans leur capacité à considérer conjointement les longues dépendances entre termes et les relations sémantique des mots. Cependant, ces modèles de langue requièrent un très grand nombre de données pour être appris efficacement, et la plupart des ces travaux se sont concentrés sur la mise au point de modèles de langue génériques (i.e. d'un ensemble de documents). Cela rend alors impossible leur application directe aux tâches de RI ad-hoc. L'apprentissage par maximisation de vraisemblance d'un modèle individuel pour chaque document de la collection, comme c'est le cas dans le cadre des approches de RI basées sur des modèles de langues n-grammes classiques, ne permet pas l'estimation correcte de paramètres, puisque issus d'un volume trop restreint d'observations de dépendances.



Une alternative intéressante a été proposée par (Le et Mikolov, 2014) qui ont récemment publié un modèle de langue neuronal dans lequel ils suggèrent de représenter le contexte (un document ou un paragraphe) comme un vecteur qui modifie le modèle de langue, ce qui évite la coûteuse construction d'un modèle individuel pour chaque document considéré et permet de tirer parti de dépendances générales observées sur le corpus. Notre travail est basé sur leurs conclusions. Nous suivons une approche dans laquelle les dépendances entre mots sont modifiées par les représentations des documents considérés. Nos contributions sont les suivantes : (i) Nous généralisons le modèle proposé dans (Le et Mikolov, 2014), en définissant de nouvelles manières de prendre en compte les spécificités individuelles de chaque document ; (ii) Nous appliquons ce modèle aux tâches ad-hoc de recherche d'information.

## 2. Travaux connexes

Les premières approches pour le traitement des dépendances entre les termes dans le domaine de la RI consistaient à étendre le modèle de représentation de texte dit en « sac de mot », en incluant les bigrammes dans le vocabulaire. Une telle approche a été suivie pour les modèles vectoriels (Fagan, 1987) puis pour les modèles de langue (Song et Croft, 1999 ; Srikanth et Srihari, 2002 ; Gao *et al.*, 2004), où les auteurs ont proposé d'utiliser un mélange des modèles de langue unigramme et bigramme, la différence demeurant dans la manière d'estimer le modèle de langue bigramme ou la manière de sélectionner les bigrammes. Cette approche n'a pas eu le succès escompté, probablement parce que cela entraîne l'utilisation d'observations peu nombreuses (Zhai, 2008) menant à des estimations inexactes des probabilités de bigramme. Dans cet article, nous étudions une autre manière de modéliser les distributions de n-grammes, avec  $n$  non limité à 1 ou 2.

Des modèles probabilistes plus sophistiqués ont été proposés pour prendre en compte des dépendances plus longues : Les modèles de proximité orientés région combinent le score de plusieurs modèles (Metzler et Croft, 2005 ; Bendersky et Croft, 2012 ; Blanco et Boldi, 2012), chacun traitant d'une dépendance spécifique entre les termes de la question. Ces travaux sont en quelque sorte orthogonaux au nôtre, puisque nous ne cherchons pas à combiner différentes sources d'informations, mais plutôt à examiner si un modèle de langue paramétrique peut capturer une séquence de termes caractéristiques.

L'une des techniques les plus utilisées pour répondre à la question de la discordance de vocabulaire est l'ajout (artificiel) de termes à la question basé sur un ensemble de (pseudo) documents pertinents fondé sur le pseudo-retour de pertinence (Manning *et al.*, 2008). Ce procédé a montré, dans certains cas, une amélioration des résultats de la recherche, au risque de dériver de la question initiale. L'emploi du pseudo-retour de pertinence est orthogonal à notre approche, mais pourrait être envisagé comme une extension pour estimer un modèle de langue des questions pertinentes (Lavrenko, 2010).

Une analyse globale peut être utilisée pour enrichir la représentation du document en s'appuyant sur la cooccurrence d'informations à l'échelle de la collection : Les techniques de réduction de dimension telles que LSI, ont été proposées pour répondre aux questions de discordance de vocabulaire (Deerwester, Scott *et al.*, 1990). L'idée est de représenter à la fois le document et la question dans un espace latent et de calculer un score de pertinence basé sur cette représentation. Toutefois, dans la pratique, ces modèles ne fonctionnent pas bien car de nombreux termes spécifiques au document sont considérés comme du bruit (Wang *et al.*, 2013). Il est donc nécessaire de combiner les scores de ces modèles latents avec les scores des modèles standards de RI tels que BM25 pour observer une amélioration de l'efficacité (Wang *et al.*, 2013 ; Deveaud *et al.*, 2013). Dans cet article, nous utilisons une combinaison d'un modèle de langue neuronal spécifique au document avec un modèle multinomial unigramme standard.

L'idée d'utiliser des réseaux de neurones pour construire des modèles de langue a émergé ces dix dernières années. Ce thème de recherche s'inscrit dans le récent domaine de « l'apprentissage de représentations ». Bengio et al. (Bengio *et al.*, 2003) présente l'un des premiers travaux sur les modèles de génération de texte (à l'échelle des mots) construit avec des réseaux de neurones. Ce type de modèle est caractérisé par une représentation distribuée des mots dans un espace vectoriel et par un état (un vecteur de  $\mathbb{R}^n$ ) qui représente le contexte<sup>1</sup>. Cet état définit une distribution de probabilité sur les mots, et peut être mis à jour après chaque nouvelle observation (mot), permettant ainsi de définir un modèle de langue sur un texte complet. Cette manière de représenter du texte dans un espace latent a été largement explorée depuis avec des applications en détection de sentiment (Socher *et al.*, 2012) ou à la traduction automatique (Schwenk, 2013).

Plus proche de la RI, l'idée de représenter le contexte par un état dans un espace vectoriel a été exploitée par Palangi et al. (Palangi *et al.*, 2014) qui a proposé d'utiliser l'état obtenu à la fin du document (resp. de la question) comme une représentation vectorielle de l'ensemble du document (resp. de la question). Le score de pertinence est alors égal au cosinus entre le vecteur de la question et celui du document. Palangi et al. ont entraîné le modèle sur les clics des utilisateurs du moteur de recherche, et ont observé que leur modèle était en mesure de mieux ordonner les documents.

Contrairement à ces travaux, la représentation d'un document n'est pas le résultat d'un encodage successif du contexte mais conditionne le processus génératif, i.e. un jeu de paramètres est utilisé pour modifier un modèle probabiliste génératif préalablement appris.

Ces modèles probabilistes génératifs que nous appellerons *paramétriques* ont d'abord été appliqués dans le cadre de la reconnaissance du locuteur (Wilson et Bobick, 1999). Leur utilisation est nécessaire car ces systèmes doivent s'adapter assez rapidement à un nouveau locuteur. L'idée générale de ces modèles est de représenter le contexte comme un vecteur dans un espace vectoriel de petite dimension. Ce

---

1. ce contexte peut être calculé de façon récursive, en considérant l'état précédent ainsi que le mot courant, ou bien en utilisant une fenêtre avant le mot à prédire

vecteur modifie le modèle génératif appris sur l'ensemble des locuteurs. Comme la distribution de sortie dépend non seulement de l'état, mais aussi du contexte, un modèle peut exprimer un grand nombre de distributions de probabilité avec un nombre limité de paramètres supplémentaires.

Cette idée a été exploitée par Le et Mikolov (Le et Mikolov, 2014). Ils ont étudié les performances d'un modèle de langue paramétrique sur une tâche d'analyse des sentiments et une tâche de récupération de documents connexes, où les relations entre les questions sont modélisées par la distance entre leurs représentations respectives dans l'espace projeté considéré. Nous proposons dans cet article d'étendre cette approche en concevant des modèles plus génériques dédiés aux tâches ad-hoc de RI.

### 3. Réseaux de neurones pour la RI

Dans cette section, nous rappelons d'abord les principes des modèles de langue classiques de RI avant de présenter notre contribution. Comme la formulation de la plupart des modèles fait appel aux séquences, pour clarifier et raccourcir les notations, nous définissons  $X_{i...j}$  comme la séquence  $X_i, X_{i+1}, \dots, X_{j-1}, X_j$  et supposons que la séquence est vide quand  $i > j$ .

Les modèles de langue sont des modèles probabilistes générant du texte – considéré comme une séquence de termes. Si un texte est composé d'une séquence de termes  $t_{1...n}$ , où chaque  $t_i$  correspond à un mot dans un vocabulaire prédéfini, on peut calculer la probabilité  $P(t_{1...n}|M)$  d'observer cette séquence étant donné le modèle de langue  $M$ . Appliqué à la RI, l'approche la plus courante (Zhai, 2008) consiste à faire l'hypothèse que la pertinence d'un document  $d$  pour une question  $q$  est égale à la probabilité que le modèle de langue du document  $M_d$  génère la question.

$$P(d \text{ pertinent pour } q) = P(q|M_d)$$

où  $M_d$  est le modèle de langue du *document* à proprement parler, qui est le modèle (appartenant à la famille de modèle  $\mathcal{M}$ ) maximisant la probabilité d'observer le document  $d$  composé des termes  $d_{1...N}$ . On a ainsi :

$$M_d = \operatorname{argmax}_{M \in \mathcal{M}} P(d|M) = \operatorname{argmax}_{M \in \mathcal{M}} \sum_{i=1}^N \log P(d_i|d_{1...i-1}, M) \quad [1]$$

Parmi les différentes familles de modèles génératifs, la famille des modèles de n-grammes multinomiaux est la plus utilisée en RI, avec  $n$  habituellement égal à 1. Les modèles multinomiaux suivent l'hypothèse de Markov en supposant que l'occurrence d'un terme ne dépend que des  $n - 1$  termes précédents.

Avec  $n = 1$ , nous obtenons un modèle unigramme simple, qui ne tient pas compte du contexte de l'expression (à noter que dans ce cas le dénominateur est égal à la longueur du document). Par exemple, dans un document à propos de Boston, le terme « sentier » est plus susceptible de se produire avant le mot « liberté » que dans d'autres

documents. Il est important de prendre en compte cette information pour construire des modèles de RI plus précis, puisqu'un *bon* modèle pour un document traitant de Boston donnerait une probabilité plus élevée au terme « sentier » lorsqu'il apparaît avant le mot « liberté », et les documents correspondants obtiendraient donc un score plus élevé avec les questions contenant la séquence « le sentier de la liberté ». Des travaux comme ceux de (Song et Croft, 1999) ont exploré l'utilisation de modèles de langue avec  $n > 1$ . Dans ce cas, les modèles sont capables de capturer les dépendances entre termes, mais généralement au prix d'une complexité plus élevée et d'une perte des capacités de généralisation, en raison de la rareté des données – plus les séquences sont longues, plus elles sont improbables dans un document  $d$ , quand bien même ces séquences sont fortement liées au contenu de  $d$  du point de vue de l'utilisateur. Avec  $n \geq 2$ , les probabilités estimées sont dans la plupart des cas égales à 0.

Même pour les unigrammes ( $n = 1$ ), l'estimation donnée par la méthode du maximum de vraisemblance peut être mauvaise et ignorer des documents simplement parce qu'ils ne contiennent pas un terme de la question, alors même qu'ils contiennent plusieurs occurrences de tous les autres. Pour éviter ce type de problèmes, des techniques de lissage sont utilisées pour empêcher la probabilité d'être nulle (et donc qu'un document ait un score égal à 0) en mélangeant le modèle de langue du document avec le modèle de langue de la collection <sup>2</sup> noté  $M_C$ . Un modèle de langue de la collection  $M_C$  correspond au modèle de langue qui maximise la probabilité d'observer les séquences de termes contenus dans les documents de la collection  $C$ .

On utilise classiquement le lissage de Jelinek-Mercer qui consiste en un mélange du modèle de langue du document et du modèle de langue de la collection. Étant donné un coefficient de lissage  $\lambda \in [0, 1]$ , le modèle de langue du document devient :

$$P(t_i | t_{i-n+1} \dots t_{i-1}, \lambda, d, C) = (1 - \lambda)P(t_i | t_{i-n+1} \dots t_{i-1}, M_d) + \lambda P(t_i | t_{i-n+1} \dots t_{i-1}, M_C) \quad [2]$$

Mais même pour de faibles valeurs de  $n$ , le lissage à partir de la collection pourrait ne pas être efficace. Dans cet article, nous proposons de développer de nouveaux modèles de langue incluant des méthodes de lissage plus sophistiquées qui sont en mesure de traiter à la fois de longues dépendances entre les termes et le problème de discordance de vocabulaire.

### 3.1. Modèles de langue neuronaux pour la RI

Les représentations distribuées de mots et de documents sont connues depuis longtemps en RI (Deerwester, Scott *et al.*, 1990). Elles permettent de surmonter le problème de la parcimonie des données que nous venons d'évoquer en s'appuyant sur les relations spatiales entre les objets représentés. Cela a été exploité en RI pour faire

2. Il peut s'agir de la collection contenant le document, ou d'une toute autre collection de documents.

face au problème de discordance de vocabulaire, mais l'idée de tirer parti de ce type de représentation pour les modèles de langue est plus récente (Bengio *et al.*, 2003). Ces modèles sont construits en utilisant des réseaux de neurones (d'où leur appellation) et offrent plusieurs avantages du fait de l'utilisation d'un espace continu : (i) la possibilité de considérer des dépendances plus longues ( $n > 2$ ); (ii) la disparition des problèmes liés à l'estimation des probabilités pour les mots peu fréquents dans un document.

Dans cet article, nous proposons d'inclure un tel modèle de langue dans la formule classique de calcul de la probabilité de chaque terme d'une question de RI. Nous proposons ainsi de remplacer le modèle de langue général de l'équation 2 par un modèle de langue neuronal *spécifique au document*  $d$  :

$$P(t_i|t_{i-n+1\dots i-1}, d) = (1 - \lambda)P_U(t_i|d) + \lambda P_{NN}(t_i|t_{i-n+1\dots i-1}, d) \quad [3]$$

où  $P_{NN}(t_i|t_{i-n+1\dots i-1}, d)$  est la probabilité d'observer le terme  $t_i$  après la séquence  $t_{i-n+1\dots i-1}$  dans le document  $d$  selon notre modèle de réseau de neurones et  $P_U$  est le modèle de langue uni-gramme. Ceci revient à introduire les dépendances entre les termes et la proximité sémantique dans un modèle de langue uni-gramme classique qui ne seraient pas en mesure de capturer de telles relations. Idéalement, nous aimerions que les meilleures performances soient atteintes pour  $\lambda = 1$ , mais dans la pratique la mixture permet de conserver une probabilité plus forte pour les termes spécifiques au document. Nous discutons ce point dans la conclusion de cet article.

Deux types de modèles sont détaillés ci-après : (i) Un modèle de langue neuronal *général* défini sur l'ensemble de la collection (section 3.1.1); (ii) Un modèle de langue neuronal *spécifique* au document estimé sur l'ensemble de la collection et le document étant classé (section 3.1.2). Notez que dans ce cas, nous sommes intéressés par la performance du modèle lorsque  $\lambda$  est proche de 1 (idéalement 1) car cela signifie que le modèle spécifique au document est suffisamment précis pour représenter pleinement le document. Ces deux modèles sont représentés dans la figure 1, où la partie noire correspond à la partie commune aux deux modèles et les parties vertes et bleues représentent respectivement le modèle général et celui spécifique au document.

### 3.1.1. Modèle de langue neuronal général

Il existe deux types de modèles de langue neuronal. Ceux qui prennent en compte un contexte virtuellement infini (réseaux récurrents) et ceux qui ne prennent en compte qu'un nombre limité de termes précédents (réseaux de convolution). Les travaux que nous présentons sont basés sur ces derniers, mais il serait intéressant d'étudier dans le futur des approches basées sur des réseaux récurrents.

Dans notre cas, l'entrée du réseau de neurones (Figure 1) correspond aux  $(n-1)^{\text{ème}}$  termes précédents. Pour les  $n-1$  premiers mots d'un document, nous utilisons un terme spécial dit de « remplissage » (*padding*). À chaque terme  $t_i$  correspond un vecteur  $z_i^t \in \mathbb{R}^{m_0}$ . Les  $n-1$  vecteurs sont ensuite transformés par une fonction  $\phi$  (décrite ci-dessous) en un vecteur d'état  $s$  dans  $\mathbb{R}^{m_f}$  où  $f$  est la dernière couche

du réseau de neurones correspondant au calcul du vecteur de contexte. Ce vecteur sert d'entrée au classifieur (HSM sur la figure 1) qui calcule la probabilité d'observer chacun des termes du vocabulaire en fonction de ce vecteur de contexte.

Lors de nos expériences, nous avons utilisé trois architectures de réseaux de neurones, i.e. trois différentes fonctions pour  $\phi$ . Ces trois modèles se distinguent par une complexité croissante, et devraient donc être à même de modéliser des dépendances plus fines.

#### 3.1.1.1. Model 1 (M1) : linear( $m_1$ ) – tanh

La première couche transforme linéairement les  $n - 1$  vecteurs  $z_j \in \mathbb{R}^{m_0}$  en un vecteur d'état  $s \in \mathbb{R}^{m_1}$ . Autrement dit, on a :

$$l_1 = \sum_{j=1}^{n-1} A_j z_j + b$$

où les  $A_j$  sont des matrices de dimensions  $m_1 \times k$  et  $b$  un vecteur de biais dans  $\mathbb{R}^{m_1}$ . La deuxième couche introduit une non-linéarité en calculant la tangente hyperbolique (tanh) de chaque composante de son entrée :

$$\forall j \ l_{2j} = \tanh(l_{1j})$$

Dans ce modèle, la fonction  $\phi$  a  $(n - 1) \times m_0 \times m_1$  paramètres.

#### 3.1.1.2. Model 2 (M2) : linear( $m_1$ ) – tanh – linear( $m_2$ ) – tanh

La deuxième fonction  $\phi$  que nous considérons est une extension de la première à laquelle nous ajoutons une couche linéaire (la matrice  $B$  de dimensions  $m_2 \times m_1$ ) et une couche non-linéaire (tanh). Dans ce modèle, la fonction  $\phi$  a  $(n - 1) \times m_0 \times m_1 + m_1 \times m_2$  paramètres.

#### 3.1.1.3. Model 3 (M2Max) : linear( $\kappa m_1$ ) – max( $\kappa$ ) – linear( $m_2$ ) – tanh

Pour construire le troisième modèle, nous remplaçons la deuxième couche (tangente hyperbolique) du modèle précédent par une autre fonction non-linéaire : une couche de regroupement maximale (*max-pooling*). Les couches de regroupement maximales sont utiles aux réseaux de neurones profonds, car elles introduisent un invariant (Kalchbrenner *et al.*, 2014), et permettent d'apprendre plus facilement le contexte dépend par exemple de deux mots qui peuvent être séparés par un espace ("le vase est bleu" et "le grand vase est bleu"). Cette couche est définie par un paramètre  $\kappa$  (valant 4 dans nos expériences)

$$\text{max-pooling}_{\kappa}(x)_j = \max\{x_{\kappa \times (j-1)}, \dots, x_{\kappa \times j - 1}\}$$

Dans ce modèle, la fonction  $\phi$  a  $(n - 1) \times \kappa \times m_0 \times m_1 + m_1 \times m_2$  paramètres.

Pour les trois modèles, d'une séquence de  $n - 1$  termes ( $t_{1 \dots n-1}$ ), nous obtenons ensuite un vecteur  $\phi(z_{1 \dots n-1})$  résumant l'information. Chaque vecteur  $z_i$  est ensuite

utilisé pour calculer la distribution de probabilité sur les termes  $p(t|z_i)$  – la probabilité que chaque mot du vocabulaire apparaisse après une séquence donnée de  $n - 1$  termes.

Dans notre modèle, nous utilisons une fonction exponentielle normalisée hiérarchique (HSM) qui permet de calculer la probabilité d’une classe (sachant le contexte) de manière efficace lorsque le nombre de classes est grand (Morin et Bengio, 2005). Le HSM est composé d’un arbre (binaire) dont les feuilles correspondent aux mots du vocabulaire. Cette fonction est définie par :

$$HSM_t(v) = \prod_{s \in \text{path}(t)} \frac{1}{1 + \exp(b_s(t) \times x_s \cdot v)} \quad [4]$$

où  $HSM_t$  représente la composante correspondant au mot  $t$  dans la distribution encodée par la couche HSM du réseau de neurones,  $v$  correspond au vecteur d’entrée de la fonction,  $\text{path}(t)$  correspond à l’ensemble des nœuds qui forment le chemin menant à la feuille qui représente le mot  $t$ ,  $x_s$  est le vecteur attaché au nœud interne  $s$  de l’arbre et  $b_s(t)$  vaut  $-1$  (resp.  $1$ ) si le chemin vers le mot  $t$  emprunte la branche gauche (resp. droite) du nœud  $s$ . Dans nos expériences, l’arbre de la fonction HSM est un arbre d’Huffman, mais d’autres choix sont possibles (en particulier, il serait intéressant de grouper ensemble les mots proches sémantiquement).

Ceci nous permet de calculer facilement la distribution de probabilité conditionnelle  $P_{NN}(t|t_{1...n-1})$  pour le prochain nœud  $w \in \Omega$ , connaissant la précédente séquence des  $n - 1$  mots observés :

$$P_{NN}(t|t_{1...n-1}) = HSM_t(\phi(t_{1...n-1})) \quad [5]$$

Enfin, pour obtenir un modèle opérationnel capable de calculer la probabilité de générer une question donnée, par rapport à notre modèle générique, il est nécessaire d’apprendre la représentation des mots, les paramètres du HSM et les paramètres de chaque couche. Cet ensemble de paramètres est noté  $\theta$ . Le problème d’apprentissage peut donc être formulé comme la maximisation de la probabilité d’observer le document  $d$  dans la collection  $\mathcal{D}$  de documents :

$$\Theta^* = \underset{\Theta}{\operatorname{argmax}} \sum_{d \in \mathcal{D}} \sum_{i=0}^{|d|} w(d_i) \log HSM_{d_i}(\phi(z_{i-n+1...i-1}^d)) \quad [6]$$

où  $w(d_i)$  est le poids utilisé pour minimiser l’importance des termes fréquemment rencontrés dans la collection (Mikolov *et al.*, 2013).

### 3.1.2. Modèle de langue neuronal dépendant du document

Le modèle de langue neuronal générique présenté précédemment prend en compte les dépendances et les relations sémantiques entre les mots pour l’ensemble des documents de la collection. Ce modèle de langue peut être une bonne alternative au modèle de langue uni-gramme multinomial de la collection utilisé pour le lissage (Equation [2]). Cependant, nous pensons que la prise en compte des spécificités du

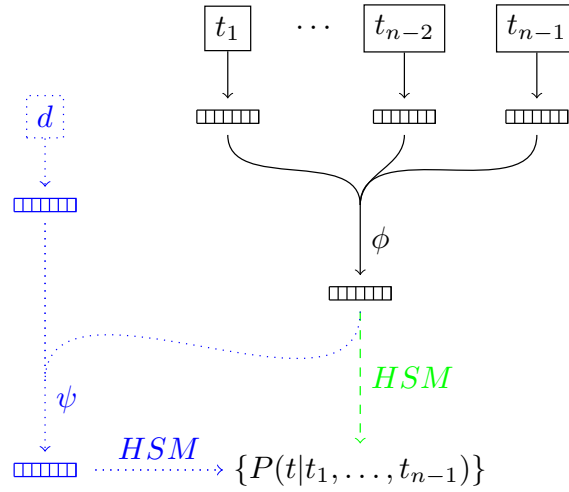


**Figure 1.** Architecture générale des modèles de langue neuronaux générique (en noir et vert hachuré) et spécifique au document (en noir et bleu pointillé).

document en cours de classement doit conduire à un meilleur modèle de langue, et donc à de meilleurs résultats de recherche, car les termes plus spécifiques au document devraient obtenir des probabilités plus fortes.

Apprendre un modèle de langue neuronal spécifique pour tous les documents de la collection est impossible pour les mêmes raisons que pour les modèles de langue à n-gramme avec  $n > 1$  : l'apprentissage des dépendances entre termes sur les séquences contenues dans un unique document serait sujet au sur-apprentissage, puisque l'échantillon de n-grammes est faible. Pour surmonter ce problème, nous suivons l'approche de (Le et Mikolov, 2014), où la distribution de probabilité d'un modèle de langue neuronal générique est modifiée par un petit ensemble de paramètres qui peuvent être appris de manière plus fiable à partir de l'observation d'un seul document. La dimension d'un tel vecteur (de 100 à 200) est typiquement beaucoup plus petite que le nombre de paramètres d'une distribution multinomiale classique (taille du vocabulaire). Cette approche est intéressante, car elle permet d'abord d'apprendre un modèle de langue sur l'ensemble de la collection, et ensuite d'apprendre seulement à le modifier pour un document spécifique. L'emploi de paramètres pour modifier le comportement d'un modèle probabiliste génératif a été utilisé dans de nombreux travaux en traitement du signal, comme la reconnaissance des gestes (Wilson et Bobick, 1999), où le modèle doit être rapidement adapté à un utilisateur spécifique. Ces modèles bénéficient d'une grande source d'informations (la collection de tous les gestes ou de tous les documents dans notre cas) pour être assez généraux et en même temps peuvent être rendus suffisamment spécifiques pour décrire un seul utilisateur ou un document.

La modification du modèle de langue neuronal est présentée dans la figure 1 par la partie bleue en pointillé. Un vecteur  $z_d \in \mathbb{R}^{m_f}$  spécifique au document est utilisé pour modifier le vecteur représentant le contexte du prochain mot au sein du modèle



de langue générique appris sur toute la collection. Le vecteur ainsi modifié est ensuite utilisé pour générer une distribution de probabilité en utilisant un HSM, comme pour le modèle de langue générique.

Dans cet article, nous considérons deux opérations de « fusion »  $\psi : \mathbb{R}^{m_f} \times \mathbb{R}^{m_f} \rightarrow \mathbb{R}^{m_f}$  qui associent le vecteur d'état  $s$  donné par  $\phi$  et le vecteur  $z_d$  spécifique au document à :

- leur somme, à savoir  $\psi(s, z_d) = s + z_d$
- leur produit composante par composante, c'est-à-dire  $\psi(s, z_d) = s \odot z_d$

La première est celle utilisée dans (Le et Mikolov, 2014), et correspond à un déplacement dans l'espace thématique afin de se rapprocher des mots spécifiques au document. La seconde suppose que certains composants du vecteur d'état sont liés à des champs sémantiques différents – si certaines composantes de  $z_d$  sont nulles, alors les champs sémantiques correspondants ne sont pas considérés.

Ces deux fonctions sont encore simples, mais elles peuvent sensiblement modifier la répartition du modèle de langue. En reprenant l'exemple sur Boston, les composantes de  $z_d$  biaiserait le modèle pour les mots susceptibles de se produire dans ce type de documents, et ainsi augmenteraient la probabilité de trouver « chemin » avant « liberté ». Ce biais est obtenu en transformant le vecteur d'état afin qu'il soit plus orthogonal aux vecteurs du HSM qui mènent au mot « chemin » qu'à ceux qui mènent aux autres mots associés à « liberté ».

Notez que, comme la solution à notre problème d'optimisation n'est pas unique, des solutions équivalentes (vis à vis de la fonction de coût) pourraient être obtenues par rotation – ce qui, par conséquent, aurait un impact sur le bénéfice de l'utilisation de ce type de modifications. Nos expériences montrent cependant qu'il y a un gain associé à l'utilisation de fonctions, même simples, comme la multiplication ou l'addition terme à terme. Nos futurs travaux exploreront des transformations du vecteur d'état plus sophistiquées et plus appropriées.

En théorie, ce modèle de langue spécifique au document pourrait être utilisé seul (c.-à-d. sans lissage, car il est basé sur un modèle de langue général) pour estimer la probabilité de générer une question. En pratique, comme montré dans les expériences ci-dessous, le modèle n'est pas encore suffisamment puissant pour y parvenir. Toutefois, combiné avec un modèle unigramme multinomial classique telle que proposé par l'équation 3, il permet d'observer des améliorations intéressantes pour les tâches ad-hoc de RI. C'est une première étape vers une solution formelle pour la prise en compte des discordances de vocabulaire et des dépendances entre les termes.

#### 4. Expériences

Nous avons utilisé les collections TREC-1 à TREC-8 pour nos expériences. Le modèle de base est BM25 (Robertson et Zaragoza, 2009) avec les paramètres usuels ( $k_1 = 1.2$  and  $b = 0.5$ ).

| Nom   | Modèle                                        | $\phi$  | word/HSM   |
|-------|-----------------------------------------------|---------|------------|
| M1    | linéaire(100) - tanh                          | 40,000  | 75,043,700 |
| M2    | linéaire(100) - tanh - linéaire(100) - tanh   | 50,000  | 75,043,700 |
| M2Max | linéaire(400) - max(4) - linéaire(100) - tanh | 170,000 | 75,043,700 |

**Tableau 1.** Modèles et nombre de paramètres pour la fonction  $\phi$  (3ème colonne) et la représentation des mots/le classificateur ()

Pour apprendre les modèles de langue, nous avons pré-traité l'ensemble des collections de documents de chaque TREC<sup>3</sup> en utilisant le stemmer de Porter et nous n'avons pas utilisé d'anti-dictionnaire. En revanche, nous avons retiré les mots apparaissant moins de 5 fois dans le jeu de données, car il est difficile d'apprendre une représentation fiable pour des termes apparaissant peu de fois. Ces derniers sont capturés par le modèle de langue uni-gramme (terme  $P_U$  de l'équation 3). La taille du vocabulaire ainsi obtenu est de 375 219 mots. Nous avons utilisé word2vec (Mikolov *et al.*, 2013)<sup>4</sup> pour pré-calculer les représentations des mots et les paramètres du HSM. Les paramètres de word2vec sont ceux par défaut (fenêtre de taille 5, pas d'apprentissage de 0.5). Nous avons utilisé 5 itérations sur le corpus afin d'apprendre une représentation fiable des mots apparaissant un petit nombre de fois.

Nous avons ensuite appris les modèles de langue génériques (Section 3.1.1) donnés dans le tableau 1. Nous pouvons voir également dans ce tableau que l'essentiel des paramètres est lié à la représentation des mots.

Lors de l'apprentissage, nous avons utilisé une descente de gradient stochastique par lots, avec un pas de gradient décroissant en fonction du nombre d'itérations :

$$\epsilon_k = \frac{\epsilon_0}{1 + k \times \delta}$$

où  $\epsilon_0 = 0.1$  et  $\delta = 2e - 4$ . Le nombre d'itération a été fixé de manière empirique à 100000, ce qui correspond à (1) avoir vu cinq fois chaque mot du corpus (2) une vraisemblance qui n'évolue presque plus.

Pour chaque question, nous avons utilisé BM25 pour sélectionner 100 documents. Pour ces 100 documents, les paramètres spécifique au document  $z_d$  ont été appris. Nous avons également utilisé une descente de gradient – dans ce cas, RProp (Riedmiller et Braun, 1993) a été utilisé, car il permet une convergence rapide sur un petit ensemble de paramètres (pas initial fixé à 0.1). Nous avons itéré jusqu'à obtenir une différence entre les paramètres décrivant le document de deux itérations successives inférieur à  $1e - 4$ .

3. Les besoins du HSM décide de la taille minimale du corpus (Chen *et al.*, 2015)

4. Le code source est disponible à l'adresse suivante <https://code.google.com/p/word2vec/>

Les résultats sont donnés Figure 2 où nous comparons les différents modèles proposés avec le modèle de langue basé sur un lissage Jelinek-Mercer ainsi que BM25, pour les métriques MAP et G-MAP (moyenne géométrique des précisions moyenne). La métrique G-MAP est intéressante car elle est plus sensible à l'amélioration des questions les plus difficiles que la métrique MAP.

Les neuf modèles sont indiqués en utilisant le nom du modèle (M1, M2 ou M2max) suivi de la méthode utilisée pour fusionner le vecteur de contexte avec le vecteur représentant le document (# correspond au modèle neutre – sans adaptation au document, + au modèle additif et \* au modèle multiplicatif). Les abscisses correspondent à la valeur de  $\lambda$  (équations 2 et 3).

Les résultats montrent que les performances des différents modèles de langue (neuronaux ou classiques) sont proches, et qu'en fonction du jeu de test les conclusions sont inversées. Il n'y a donc pas encore d'amélioration liée à l'utilisation de ces modèles. Des études qualitatives ont montré que (1) les mots très spécifiques au document mais apparaissant peu dans le corpus ont des probabilités qui sont sous-évaluées par les modèles neuronaux, des modèles permettant de modifier de manière plus fine le contexte sont donc nécessaires ; et (2) le classifieur HSM a des problèmes pour les mots dont la fréquence d'occurrence est faible. Des expériences complémentaires sont nécessaires pour résoudre ces problèmes.

Les autres observations que l'on peut faire au niveau des résultats est que l'influence des modèles neuronaux est plus grande sur les questions difficiles (la différence pour le G-MAP est plus importante que la différence pour le MAP). Nous espérons donc qu'en ayant un modèle plus fin pour les documents, cette différence va s'accroître.

Finalement, les modèles plus complexes semblent avoir de meilleures performances : M2 et M2Max sont souvent supérieurs ou comparables au modèle M1. Utiliser des modèles plus complexes, et des modèles récurrents, peut potentiellement être également une source d'amélioration des résultats.

## 5. Conclusion

Dans cet article, nous avons proposé d'utiliser des modèles de langue neuronaux paramétriques pour la recherche d'information. Nous avons proposé pour chaque modèle deux variantes, une correspondant au modèle de langue général (collection), ainsi qu'un modèle spécifique à un document décrit par un petit nombre de paramètres modifiant le processus génératif. Ces paramètres peuvent donc être appris même pour des documents de petite taille.

Les expériences conduites avec les collections TREC-1 à TREC-8, où nous avons comparé un modèle de langue classique avec les modèles neuronaux, montrent que les modèles actuels n'apportent pas d'amélioration en performance. Des analyses qualitatives des résultats montrent que l'utilisation de tels modèles demande à résoudre des

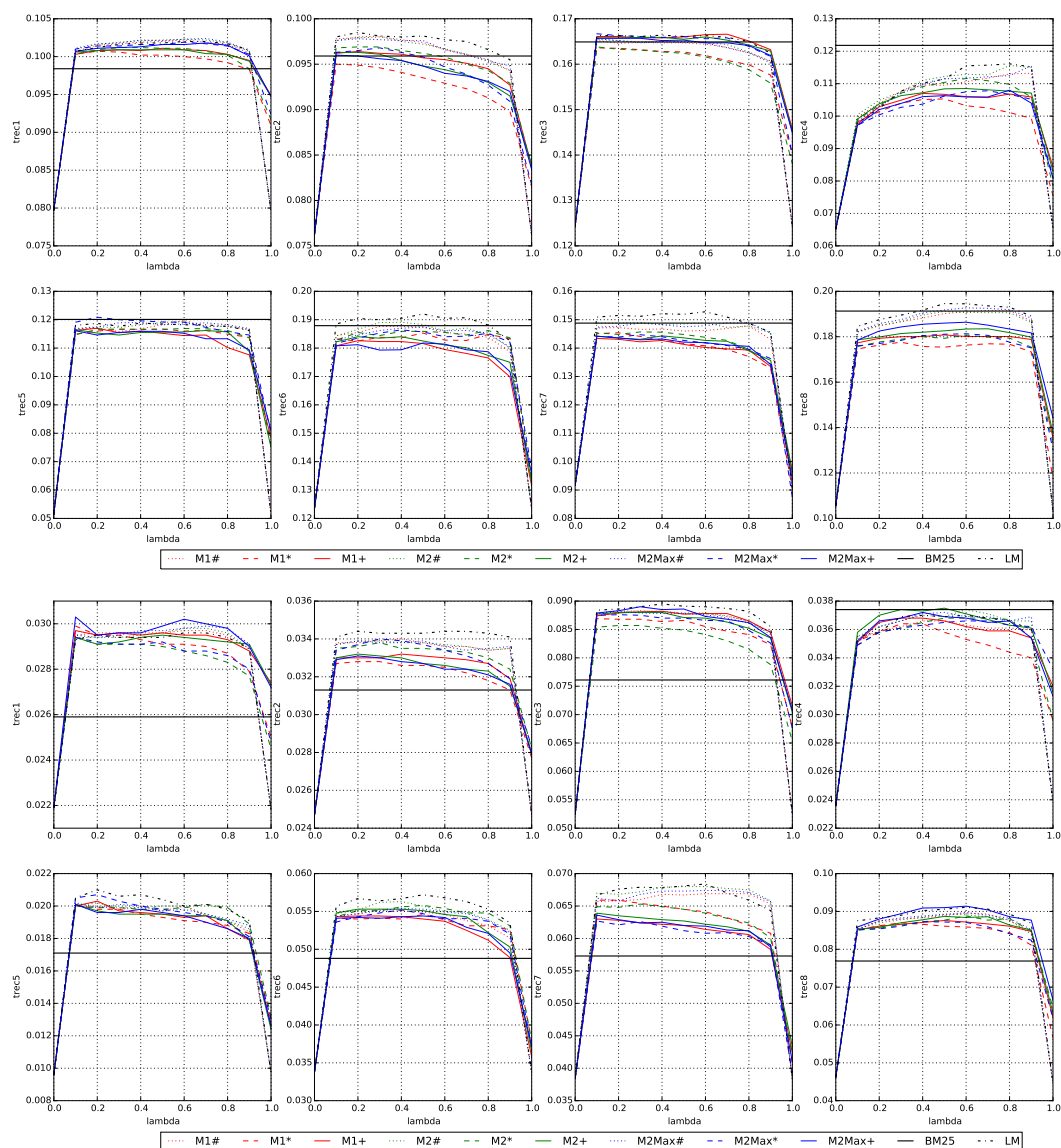


**Figure 2.** MAP (haut) et G-MAP (bas) pour TREC-1 à TREC-8. La performance de BM25 est donnée pour référence (courbe constante), et les résultats pour les différents modèles de langue sont donnés en fonction de  $\lambda$ . Pour les modèles basés sur les réseaux de neurones, les résultats des trois modèles M1, M2 et M2Max sont donnés pour les trois variantes : # correspond au modèle général, + (resp. \*) pour le modèle spécifique basé sur l'addition (resp. la multiplication).

problèmes qui ne se posent pas pour d'autres utilisations des modèles neuronaux. En particulier, la probabilité des mots spécifiques au document est sous-évaluée, ce qui explique pourquoi le modèle ne se comporte pas bien lorsqu'il est utilisé sans lissage : il paraît donc nécessaire d'utiliser des transformations plus complexes.

Bien que ces résultats ne montrent pas de différence substantielle, nous pensons que de tels modèles sont intéressants car l'idée de modifier un modèle génératif par un petit ensemble de paramètres décrivant un contexte particulier (document, paragraphe, phrase, etc.) permet d'envisager d'avoir à terme des modèles prenant en compte à la fois les spécificités d'un document et un modèle de langue général. Nous pensons que, outre l'étude de nouveaux opérateurs permettant de modifier le processus génératif, il serait important d'utiliser des techniques de pseudo-retour de pertinence, qui permettrait de ne plus avoir de problème d'asymétrie entre la question et les documents.

## 6. Bibliographie

- Bendersky M., Croft W. B., « Modeling higher-order term dependencies in information retrieval using query hypergraphs », *SIGIR '12*, 2012.
- Bengio Y., Ducharme R., Vincent P., Jauvin C., « A Neural Probabilistic Language Model », *Journal of Machine Learning Research*, 2003.
- Blanco R., Boldi P., « Extending BM25 with multiple query operators », *SIGIR '12*, 2012.
- Chen W., Grangier D., Auli M., « Strategies for Training Large Vocabulary Neural Language Models », *CoRR*, 2015.
- Deerwester, Scott, Dumais, Susan T, Furnas, George W, Landauer, Thomas K, Harshman, Richard, « Indexing by latent semantic analysis », 1990.
- Deveaud R., SanJuan E., Bellot P., « Unsupervised Latent Concept Modeling to Identify Query Facets », *OAIR'13*, 2013.
- Fagan J. L., « Automatic Phrase Indexing for Document Retrieval : An Examination of Syntactic and Non-Syntactic Methods. », *SIGIR'87*, 1987.
- Gao J., Nie J.-Y., Wu G., Cao G., « Dependence language model for information retrieval », *SIGIR'04*, 2004.
- Kalchbrenner N., Grefenstette E., Blunsom P., « A Convolutional Neural Network for Modelling Sentences », *arXiv.org*, 2014.
- Lavrenko V., *A Generative Theory of Relevance*, Springer, 2010.
- Le Q. V., Mikolov T., « Distributed Representations of Sentences and Documents », *ICML'14*, 2014.
- Manning C. D., Raghavan P., Schütze H., *Introduction to Information Retrieval*, Cambridge University Press, 2008.
- Metzler D., Croft W. B., « A Markov random field model for term dependencies », *SIGIR*, 2005.
- Mikolov T., Sutskever I., Chen K., Corrado G. S., Dean J., « Distributed Representations of Words and Phrases and their Compositionality. », *NIPS'14*, 2013.
- Morin F., Bengio Y., « Hierarchical Probabilistic Neural Network Language Model. », *AISTATS 2005*, 2005.

- Palangi H., Deng L., Shen Y., Gao J., He X., Chen J., Song X., Ward R., « Semantic Modelling with Long-Short-Term Memory for Information Retrieval. », *arXiv.org*, 2014.
- Riedmiller M., Braun H., « A direct adaptive method for faster backpropagation learning : the RPROP algorithm », *IEEE International Conference on Neural Networks*, 1993.
- Robertson S. E., Zaragoza H., *The Probabilistic Relevance Framework : BM25 and Beyond*, Foundations and Trends in Information Retrieval, 2009.
- Schwenk H., « Continuous Space Translation Models for Phrase-Based Statistical Machine Translation », *Proceedings of COLING 2012 : Posters*, 2013.
- Socher R., Manning C. D., Huval B., Ng A. Y., « Semantic compositionality through recursive matrix-vector spaces », *EMNLP-CoNLL '12 : Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 2012.
- Song F., Croft W. B., « A general language model for information retrieval », *CIKM'99*, 1999.
- Srikanth M., Srihari R. K., « Biterm language models for document retrieval. », *SIGIR'02*, 2002.
- Wang Q., Xu J., Li H., Craswell N., « Regularized Latent Semantic Indexing : A New Approach to Large-Scale Topic Modeling », *ACM TOIS*, 2013.
- Wilson A. D., Bobick A. F., « Parametric hidden Markov models for gesture recognition », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1999.
- Zhai C., « Statistical Language Models for Information Retrieval : A Critical Review. », *Foundations and Trends in Information Retrieval*, 2008.