# Apprentissage d'ordonnancement pour la prédiction d'activité sur les réseaux sociaux

**François Kawala[*] [**] — Ahlame Douzal-Chouakria[**] — Eric Gaussier[**] — Eustache Diemert[*]**

*Laboratoire d'Informatique de Grenoble[**]*
*Université Grenoble Alpes*
*CNRS - LIG/AMA*
*{francois.kawala,ahlame.douzal,eric.gaussier}@imag.fr*

*BestofMedia Group[*]*
*{fkawala,ediemert@bestofmedia.com}*

RÉSUMÉ. *Nous proposons dans cet article d'apprendre à classer les mots-clés selon leur activité à venir, et comparons deux approches : point-wise et pair-wise. Pour chacune d'elle nous étudions l'influence de l'ambiguïté et de la popularité des mots-clés sur ses capacités prédictives. A notre connaissance, c'est la première fois que ces facteurs sont étudiés dans ce contexte. Pour valider nos résultats nous fournissons un jeu d'apprentissage comprenant l'activité de 1497 mots-clés observée sur Twitter pendant une année. Nous proposons également un nouveau score de ranking définie selon les contraintes applicatives de la prédiction d'activité sur les réseaux sociaux en ligne.*

ABSTRACT. *We study in this paper different strategies to rank keyword activities through a comparison of pointwise and pairwise learning to rank approaches, as well as the impact of keyword ambiguity, keyword activity and keyword set size on the prediction results. It is the first time, to our knowledge, that such dimensions are evaluated in this framework. Our experiments are conducted on a large dataset built by monitoring Twitter over a year and including 1497 keywords. The different methods tested are evaluated using standard ranking scores as well as a newly defined, application driven quality measure.*

MOTS-CLÉS : *Médias sociaux ; Apprentissage d'ordonnancement ; Prédiction d'activité*

KEYWORDS: *Social media; Keyword activity ranking; Learning to rank*

CORIA 2014, pp. 337–351, Nancy, 18-21 mars 2014

## 1. Introduction

Les récents succès de la prédiction d'événements tangibles à partir de l'observation des réseaux sociaux en ligne (RSL) tels que la prédiction des revenus d'exploitation des films ou la prédiction des taux d'infection de la grippe au sein d'une population, sont fondés sur l'utilisation de mots-clés associés à ces événements. Plus précisément, ces approches utilisent la distribution de ces mots-clés parmi les messages échangés dans un RSL cible. Être capable de prédire l'activité d'un ensemble de mots-clés est donc crucial pour les applications susmentionnées, mais aussi pour les entreprises qui souhaitent suivre l'évolution de l'intérêt des consommateurs. Nous formalisons ici ces besoins en définissant un problème d'apprentissage d'ordonnancement de l'activité. Nous proposons de répondre à ce problème en tenant compte d'un certain nombre de contraintes. En particulier sur un RSL l'approche doit : (a) être utilisable dans différents réseaux sociaux en ligne, (b) être capable de traiter des réseaux sociaux arbitrairement grands, (c) ne pas requérir une connaissance complète du réseau des utilisateurs, cette information étant rarement disponible et coûteuse à maintenir à jour.

Dans cet article nous traitons le problème d'apprentissage d'ordonnancement des mots-clés selon leur activité à venir. Au delà des capacités d'ordonnancement des méthodes *point-wise* qui correspondent aux formulations précédentes de ce problème, nous étudions les méthodes *pair-wise*. Nous étudions également l'influence de l'ambiguïté et de la popularité des mots-clés sur les résultats obtenus par chacune des méthodes. Nos principaux résultats sont :

– L'ordonnancement des mots-clés les plus populaires est plus facile que celui des mots-clés peu populaires et ceci indépendamment de la distribution de la popularité ;

– Un niveau d'ambiguïté faible ou fort conduit souvent à des erreurs de prédiction importantes pour les mots-clés les moins populaires ;

– Les résultats de l'approche *point-wise* sont stables au regard du nombre de mots-clés à ordonner.

– L'approche *pair-wise* ne permet pas un incrément significatif des résultats au regard de ceux des méthodes *point-wise*.

Cet article est structuré comme suit. Nous présentons les travaux reliés dans la Section 2. La Section 3 présente le problème d'apprentissage d'ordonnancement ainsi qu'une définition des réseaux sociaux en ligne basée sur les contenus. Les résultats expérimentaux sont présentés dans la Section 4. Enfin nous conclurons ce travail dans la Section 5.

## 2. Travaux reliés

Notre travail s'appuie sur deux domaines : l'analyse des média sociaux (AMS) et l'apprentissage d'ordonnancement. Dans le domaine de l'AMS les travaux sont nombreux et visent à prédire les revenus du *"box-office"* (Asur *et al.,* 2010), les épidémies

de grippe ou bien même les résultats électoraux. Ces études, tout comme la nôtre, n'utilisent pas le graphe des utilisateurs du RSL mais les traces de leurs activités. Ces traces peuvent être explicites comme les "tweets" ou implicites comme l'historique des requêtes collecté par un moteur de recherche. Des descripteurs, par exemple le niveau d'activité au cours du temps (Ginsberg *et al.,* 2008), sont utilisés pour représenter les observations passées. Ces dernières sont associées à une réalisation, par exemple les recettes générées par un film. Apprendre la surjection depuis l'espace défini par les descripteurs vers l'espace des réalisations permet de prédire les réalisations à venir.

Les valeurs des descripteurs évoluent avec le temps, il est donc naturel de considérer que les exemples sont des séries temporelles multivariés. Plusieurs travaux (Lehmann *et al.,* 2012, Yang *et al.,* 2011) adoptent cette approche. Dans (Yang *et al.,* 2011) les auteurs proposent de rechercher des motifs temporels en utilisant des algorithmes de partitionnement comme le "$k$-spectral centroid clustering". La valeur optimale de $k$ est déterminée par une recherche exhaustive visant à maximiser la valeur de la silhouette ou l'index de Hartigan. Ces études concluent à l'existence d'un nombre restreint de motifs temporels et conduisent à penser qu'il est possible d'apprendre à prédire l'activité sur les RSL. Cependant la question de la prédiction n'est pas traitée.

D'autres études visent à prédire les périodes de forte activité d'un mot-clé, ou encore à détecter des événements très ponctuels. Par exemple (Sakaki *et al.,* 2010) étudie la possibilité de se servir des utilisateurs de Twitter pour détecter les tremblements de terre. Deux étapes sont nécessaires : apprendre un classificateur pour différencier les messages concernant un tremblement de terre, puis un filtre à particules estime, à partir de la localisation de chacun des messages retenus, la position de l'épicentre du séisme. Un problème similaire est traité dans (Pan *et al.,* 2011) à l'aide d'un modèle génératif, et dans (Kleinberg, 2003) à l'aide d'un automate à état infini. (Yao *et al.,* 2010) complète cette approche en apprenant les paramètres d'un modèle combinant ces deux approches. Notre travail est comparable en certains points à ces travaux. Cependant nous considérons des situations plus réalistes, car nous traitons un ensemble de 1497 mots-clés, qui n'est pas pré-filtré pour retenir les plus populaires uniquement. De plus nôtre approche, testée sur Twitter, est facilement transposable à d'autres médias sociaux. Enfin notre approche aboutit à l'ordonnancement de tous les mots-clés selon leur activité à venir, et ne se limite pas à une classification binaire {activité accrue, activité basale}.

Dernièrement un grand nombre de travaux (Hong *et al.,* 2011, Tsur *et al.,* 2012, Petrovic *et al.,* 2011, Zaman *et al.,* 2010) ont été dédiés à la prédiction des actions des utilisateurs d'un RSL (par exemple, retransmettre un message à ses voisins). Dans (Tsur *et al.,* 2012) par exemple l'objectif est de prédire le taux d'adoption d'un nouveau "hashtag" (*ie.* un mot-clé volontairement préfixé par le symbole #). Ces travaux supposent toutefois que le graphe des utilisateurs est connu, ce qui limite leur application.

Les méthodes d'apprentissage d'ordonnancement peuvent être classées dans trois catégories : *point-wise*, *list-wise*, et (Liu, 2009) *pair-wise*. Les méthodes *point-wise*

(Crammer *et al.,* 2001) supposent que chaque paire requête-document dispose d'un score ordinal. L'ordonnancement est alors formulé comme un problème de régression dans lequel la valeur du rang de chaque document est estimée comme une quantité absolue. Dans le cas où les jugements de pertinence sont donnés sous la forme de paires, il n'est généralement pas simple d'employer de telles méthodes. De plus les méthodes *point-wise* ne tiennent pas compte des dépendances inter-documents (Chapelle *et al.,* 2011).

En ce qui concerne les approches *list-wise* (Valizadegan *et al.,* 2009) on considère la totalité de la liste ordonnée des documents pour chaque requête comme une instance d'entraînement, une conséquence directe étant que ces approches sont capables de différencier les documents des différentes requêtes, et de considérer leur rang lors de l'apprentissage. Cependant les approches *list-wise* visent à optimiser directement un score d'ordonnancement, et font ainsi face à un problème majeur, avec les fonctions non convexes, non différentiables, et non continues.

Enfin, dans les approches *pair-wise* (Cohen *et al.,* 1999, Freund *et al.,* 2003, Joachims, 2002), la liste ordonnée est décomposée en un ensemble de paires de documents, de telle manière qu'un classificateur puisse être entraîné pour minimiser le nombre d'inversions dans l'ordonnancement. Pendant la phase de test le classificateur assigne un score positif ou négatif à chaque documents. Le score permet directement d'ordonner ces documents. Les SVM sont l'un des classificateurs les plus populaires pour réaliser la classification sur les paires de documents en vue de les ordonner (Joachims, 2002). D'autres classificateurs populaires ont été adaptés, comme *Rankboost* qui minimise la fonction de coût exponentiel sur les paires de documents (Freund *et al.,* 2003).

Les méthodes d'apprentissage d'ordonnancement ont été appliqués aux réseaux sociaux en ligne pour améliorer "l'expérience utilisateur", en proposant aux utilisateurs les publications les plus pertinentes pour eux. Par exemple (Duan *et al.,* 2010, Uysal *et al.,* 2011, Metzler *et al.,* 2011, Zhang *et al.,* 2012) ordonnent les "tweets" selon les centres d'intérêt de l'utilisateur, ou les événements en cours. Nous traitons un problème différent, qui est d'ordonner les mots-clés selon leur activité à venir dans un RSL cible.

## 3. Définition du problème

Soit $z$ un mot-clé, $\mathcal{Z}$ un ensemble de mots-clés et $\mathcal{Z}([t_0; t])$ cet ensemble de mots-clés observé pendant la période $[t_0; t]$. Notre objectif est d'apprendre une fonction d'ordonnancement $f$ qui produise pour un sous ensemble de mots-clés donné, un classement selon leurs activités respectives dans la période temporelle à venir.

$$f : \mathcal{Z}([t_0; t]) \rightarrow R(\mathcal{Z}, t + \delta)$$

où $R(\mathcal{Z}, t + \delta)$ représente un ordonnancement des mots-clés de $\mathcal{Z}$ pour la période $[t; t+\delta]$. La fonction d'ordonnancement $f$ est apprise sur un ensemble d'entraînement

constitué de vecteurs représentant les mots-clés pendant la période $[t_0; t]$ ainsi que la valeur de leur activité pendant la période $[t; t+\delta]$. Les descripteurs utilisés pour définir ces séries temporelles et l'activité sont présentés ci-après à la Section 3.2

Les approches *list-wise* visent à optimiser des scores spécifiques à la recherche d'information (et plus précisément des approximations de tels scores). Nous ne pouvons donc pas les utiliser directement dans ce contexte où nous pouvons supposer que tous les mots-clés sont identiquement pertinents. En conséquence, nous ne considérerons que les approches *pair-wise* et *point-wise*. Dans le cas des approches *point-wise* $f$ peut être vu comme la composée : de (a) une fonction de régression $f_1$, définie de $\mathcal{Z}([t_0; t])$ vers $\mathbb{R}$ fournissant un score d'activité pour la période $[t; t + \delta]$ pour chaque mot-clé, et (b) une fonction $f_2$ qui ordonne les mots-clés selon leur score respectif. Concernant l'approche *pair-wise* il est courant pour une paire de mots-clés $(z_1, z_2)$ de considérer la différence des deux vecteurs représentant chacun un mot-clé ; de cette façon le problème à résoudre est un problème de classification : dans l'ensemble d'apprentissage une paire de $(z_1, z_2)$ a le label $+1$ si et seulement si l'activité associée à $z_1$ durant la période $[t; t + \delta]$ est plus grande que l'activité de $z_2$ pendant la même période. Les mots-clés sont ensuite ordonnés selon le score calculé par le classificateur appris (voir (Joachims, 2002) pour un exemple détaillé).

Nous définissions formellement les descripteurs utilisés pour représenter les mots-clés au cours du temps. Pour ce faire nous allons tout d'abord présenter des concepts généraux.

### 3.1. *Représenter le réseau social en ligne par son contenu*

Le graphe des utilisateurs peut être utile, cependant (a) il n'est pas toujours possible de l'obtenir, et (b) si il est disponible il n'en est pas moins coûteux à maintenir à jour. Nous proposons donc de représenter les réseaux sociaux sans utiliser le graphe des utilisateurs.

**Définition 3.1** *(Contribution $\langle z, u, \tau \rangle$) Chaque publication dans un réseau social est une contribution $\langle z, u, \tau \rangle$ mentionnant un ensemble de mots-clés $z \subseteq \mathcal{Z}$ , associée à un utilisateur $u \in \mathcal{U}$, crée au temps $\tau \in \mathcal{T}$. $\mathcal{C}$ désigne l'ensemble des toutes les contributions existantes.*

**Définition 3.2** *(Discussion $d_t$) Une discussion $d_t$ est définie comme une séquence de contributions ordonnées selon le temps :*

$$d_t = \left\{ \langle z^1, u^1, \tau^1 \rangle, \ldots, \langle z^{l_{d_t}}, u^{l_{d_t}}, \tau^{l_{d_t}} \rangle \right\}, \text{ avec } \tau^{l_{d_t}} \leq t$$

*$\mathcal{D}$ correspond à l'ensemble de toutes les discussions. La fonction* `pair` $: \mathcal{D} \times \mathcal{C} \mapsto \mathcal{D}$ *est utilisée pour ajouter une contribution à une discussion.*

**Définition 3.3** *(Fonctions activity, users, et keywords) Nous définissions trois fonctions fournissant les informations nécessaires à la définition des descripteurs.*

*1)* $\texttt{activity} : \mathcal{Z} \times \mathcal{T} \mapsto \mathbb{N}^+$ *fournit l'activité associée à un mot-clé au temps* $t$ *:* $\texttt{activity}(z,t) = |\{\langle z,u,t\rangle \in \mathcal{C}\}|$. *Pour cette fonction nous utiliserons l'abréviation* $\texttt{A}(z,t)$ *;*

*2)* $\texttt{users} : \mathcal{D} \times \mathcal{T} \mapsto \mathcal{U}^n$ *fournit l'ensemble des utilisateurs participant à* $d_t$ *au temps* $t$ *:* $\texttt{users}(d_t,t') = \{u \in \mathcal{U} \mid \langle z,u,t'\rangle \in d_t\}$ *;*

*3)* $\texttt{keywords} : \mathcal{D} \mapsto \mathcal{Z}^m$ *fournit les mots-clés utilisés dans la conversation* $d_t$ *:* $\texttt{keywords}(d_t) = \{z \in \mathcal{Z} \mid \exists\langle z,u,\tau\rangle \in d_t\}$ *;*

En utilisant `users` et `keywords`, nous pouvons définir $\mathcal{U}_{t,z}$ qui correspond à l'ensemble des utilisateurs qui participent à des discussions mentionnant le mot-clé $z$ au temps $t$ :

$$\mathcal{U}_{t,z} \quad = \quad \{\texttt{users}(d_t,t) \mid d_t \in \mathcal{D} \wedge z \in \texttt{keywords}(d_t)\}$$

Nous pouvons illustrer cette approche avec Twitter. Il s'agit d'un réseau social en ligne où les utilisateurs échangent des messages textuels de taille limitée appelés "tweets". Le graphe des utilisateurs est dirigé, en effet les utilisateurs sont liés par la relation "follow" qui est asymétrique : quand l'utilisateur $u$ "suit" l'utilisateur $v$ alors $u$ reçoit toutes les publications de $v$ sans aucun effet pour ce dernier. Ainsi dans ce réseau social en ligne, un "tweet" correspond à une *contribution*, une discussion est alors un "tweet" accompagné de l'ensemble des réponses (*reply*), et des ré-émissions (*re-tweet*) qu'il a provoqué. La fonction `pair` correspond donc indistinctement à *reply* et *re-tweet*. Enfin les fonctions `users`, `activity`, et `keywords` sont de simples énumérations sur les discussions.

### 3.2. *Descripteurs*

Pour prédire l'activité des mots-clés nous utilisons un ensemble parcimonieux de descripteurs. Il contient le descripteur objectif lui-même `activity` ( Définition 3.3) ainsi que cinq autres descripteurs. Nous tirons avantage, pour l'approche *pair-wise*, de la possibilité de définir des descripteurs pour des paires de mots-clés. Cependant ceux-ci ne pourront pas être utilisés par l'approche *point-wise*.

**Descripteurs communs.**

1) **Nombre d'utilisateurs** (NU). Noté $\text{NU}(t,z) = |\mathcal{U}_{t,z}|$, il correspond à la quantité d'utilisateurs qui participent à des discussions mentionnant $z$ au temps $t$ ;

2) **Solde des utilisateurs** (UB). Ce descripteur correspond à la quantité d'utilisateurs qui participent pour la première fois au temps $t$ à des discussions qui mentionnent le mot-clé $z$.

3) **Niveau d'attention** (AL). $\rho = \text{NU}(t,z)$ ou $\rho = \texttt{A}(t,z)$ sont des estimateurs de l'attention que les utilisateurs consacrent au mot-clé $z$ au temps $t$. Nous normalisons ces quantités par l'attention accordée à tous les autres mots-clés au temps $t$. De cette

façon nous espérons obtenir un descripteur plus robuste aux variations qui touchent tous les mots-clés indistinctement. $\text{AL}(t,z) = \rho(t,z) \;/\; \sum_{z' \in \mathcal{Z}} \rho(t,z')$.

**Descripteurs pair-wise uniquement.** Les deux descripteurs suivants sont définis pour une paire de mots-clés $(z_1, z_2)$ ; en conséquence ils ne sont définis que pour les approches *pair-wise*. Nous considérons ici que $z_1$ est plus actif que $z_2$ pendant la période d'évaluation ($[t; t+\delta]$), et que $t_f$ est le temps de la dernière observation.

1) **Différence d'activité** (AD). Ce descripteur correspond à la différence des activités des deux mots-clés à la fin de période d'observation. Il est défini par $\text{AD}(z_1, z_2) = \mathtt{A}(z_1, t_f) - \mathtt{A}(z_2, t_f)$ ;

2) **Ordonnancement des niveaux d'activité** (AO). Ce descripteur fournit le nombre d'étapes temporelles durant l'observation ($[t_0; t]$) pour lesquelles l'activité de $z_1$ est plus grande que l'activité de $z_2$. Il est donc défini par $\text{AO} = \sum_{t=0}^{t_f} \mathbb{1}(\mathtt{A}(z_1, t) > \mathtt{A}(z_2, t))$ où $\mathbb{1}$ est la fonction indicatrice.

## 4. Protocole expérimental et validation

Pour nos expériences nous utilisons des données provenant de Twitter. Durant 51 semaines consécutives nous avons quotidiennement collecté des contributions rédigées en anglais. Pour ce faire nous avons utilisé l'API REST, et une liste de 1497 mots-clés provenant de Wikipedia. Nous obtenons ainsi 287 millions de contributions échangées par 30 millions d'utilisateurs. Nous avons ensuite défini trois jeux de données distincts (LOW,MED,HIGH) chacun contenant les données collectés pour 300 mots-clés en vue d'étudier l'influence de l'ambiguïté. Pour chaque jeu de données, les exemples sont générés à partir de séquences de 9 jours consécutifs partiellement recouvrantes (*cf.* Figure 1) dans lesquelles les 7 premiers jours sont considérés comme observés et servent à calculer les descripteurs alors que les 2 derniers jours sont utilisés pour calculer la réalisation correspondante.

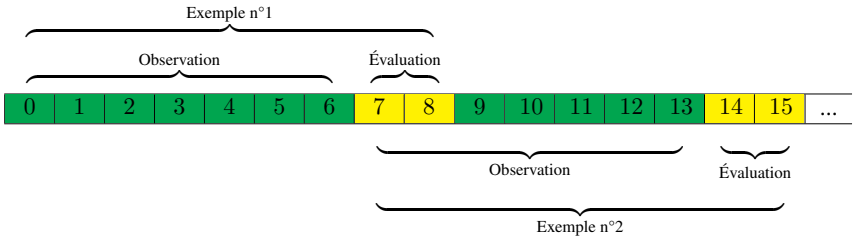


**Figure 1.** *Génération des exemples à partir d'une séquence d'observations consécutives, ici numérotées selon l'ordre d'observation.*

Nous considérons que le niveau d'ambiguïté d'un mot-clé peut être estimé par le nombre d'homonymes recensés sur sa page d'homonymie Wikipédia[1] (cette mesure a pour domaine $\mathbb{N}$). Ainsi le jeu de donné LOW contient les mots-clés n'ayant aucun homonyme, MED contient les mots-clés ayant entre 3 et 19 homonymes, enfin HIGH contient les mots-clés ayant entre 20 et 99 homonymes. Chaque jeu de données contient 51 fenêtres de 9 jours consécutifs, correspondant donc à 51 ordonnancements de 300 mots-clés ; cet ensemble est nommé $E_{\mathcal{Z}}$.

Pour évaluer les performances nous avons utilisé deux type de mesures. Premièrement les coefficients corrélation du rang de Kendall (Kendall, 1948) et de Spearman (Wolfe *et al.,* 1973). Ce dernier tient compte de la quantité d'erreur par paire. Ces deux mesures ont pour domaine $[-1, +1]$ avec $0$ signifiant l'absence de corrélation entre les deux ordonnancements. Ensuite nous avons défini la mesure de $n\text{-}surprise$ pour quantifier la capacité du système à prédire l'apparition de "nouveaux-venus" dans le $top_n$. Soit un ensemble de mots-clés $\mathcal{Z}$ et un ordonnancement $r$, alors $top_n(r, \mathcal{Z}) = \{e \in E_{\mathcal{Z}} : r(e) < n\}$. A partir de $r_{prec}$, ordonnancement observé lors de la période précédant l'évaluation, et de $r_{true}$, ordonnancement observé lors de la période d'évaluation nous définissons l'ensemble des "nouveaux-venus" :

$$nouveaux\text{-}venus(\mathcal{Z}, n) = \{e \in E_{\mathcal{Z}} : e \in top_n(r_{true}, \mathcal{Z}) \wedge e \notin top_n(r_{prec}, \mathcal{Z})\}$$

Cet ensemble contient les mots-clés des rangs $1$ à $n$ lors de l'évaluation qui ne faisaient pas partie des rangs $1$ à $n$ lors de la période précédant l'évaluation. Nous définissons la mesure de surpise en utilisant $r_{pred}$, ordonnancement prédit par le système pour la période d'évaluation :

$$surprise(\mathcal{Z}, n) = \frac{|top_n(r_{pred}, \mathcal{Z}) \cap nouveaux\text{-}venu(\mathcal{Z}, n)|}{|nouveaux\text{-}venu(\mathcal{Z}, n)|}$$

Ce score a pour domaine $[0, 1]$ et fournit des informations sur la capacité du système d'ordonnancement à gérer les changements soudains de niveau d'activité ; un score de $0$ indiquant une incapacité totale, et de $1$ indiquant l'inverse. Nous choisissons arbitrairement $n = 30$ (ce choix est guidé par l'application).

4.0.1. *Validation.*

Nous validons nos résultats par une validation croisée répétée 10 fois : après avoir divisé le jeu de données en 10 parts, neuf parts sont utilisées pour entraîner les modèles, la dernière est conservée pour évaluer les performances du modèle entraîné sur les neuf autres. Ces deux étapes sont répétées 10 fois de façon à ce que toutes les données soient utilisées exactement une fois pour l'évaluation. La moyenne des résultats obtenus est utilisée pour juger des performances d'une méthode pour un jeu de données. Pour chacun des trois jeux de données nous disposons de $15300$ (paires mots-clés, date) exemples pour les méthodes *point-wise*, et $\sim 3M$ pour les méthodes *pair-wise*.

1. http ://fr.wikipedia.org/wiki/Aide:Homonymie

4.0.2. *Apprentissage des paramètres.*

Les méthodes *point-wise* sont testées avec trois modèles d'apprentissage supervisé. Premièrement les machines à vecteur support pour la régression (SVR) avec un kernel linéaire ou RBF (Drucker *et al.,* 1997, Buhmann, 2003). Les hyper-paramètres $C$ et $\gamma$ (pour le kernel RBF) sont appris par recherche exhaustive parmi les valeurs suivantes : $C \in \{10^{-2}, 10^{-1}, 1, 10, 100\}$ et $\gamma \in \{10^{-3}, 10^{-2}, 1\}$. Nous choisissons les paramètres utilisés pour entraîner l'instance qui maximise le coefficient de corrélation du rang de Kendall. Ensuite les méthodes d'ensembles, comme les *random forest* pour la régression (RFR) (Breiman, 2001) et les *gradient boosted trees* (GBT) (Friedman, 2001). Les hyper-paramètres sont appris comme précédemment. Pour RFR le nombre d'arbre varie parmi $[5, 10, 15, 30, 50, 200]$, et le nombre minimal d'éléments d'un nœud parmi $[2, 4, 6, 8, 10]$.

L'approche *pair-wise* est testée en utilisant un modèle svmRank (Cao *et al.,* 2006). En effet celui-ci est particulièrement adapté à notre problème où chaque date a exactement le même nombre de mots-clés. Le paramètre $C$ est appris comme précédemment.

## 4.1. *Résultats*

Nous comparons les résultats des méthodes *point-wise*, correspondant à une régression, à ceux de la méthode *pair-wise* basée sur svmRank. Nous nous intéresserons ensuite aux effets de la sélection des mots-clés sur le pouvoir prédictif de chaque approche. Enfin nous évaluerons l'évolution des performances en fonction du nombre de mots-clés à ordonner.

4.1.1. *Comparaison des méthodes Pair-wise et Point-wise*

Nous comparons les résultats de chaque méthode à une méthode de prédiction naïve qui consiste à prédire le dernier ordonnancement de la période d'observation. La Figure 2 présente la moyenne des scores par jeu de données ; les barres verticales correspondent à l'écart type. La partie de gauche de cette figure, qui présente le score de Spearman, indique que l'approche *pair-wise* obtient des résultats identiques ou meilleurs que RFR la meilleure approche *point-wise*. La prédiction naïve obtient des résultats comparables aux autres méthodes mais est très instable comme l'indiquent les écarts types. La partie de droite montre que le score de surprise est plus stable, pour les trois jeux de données, que les scores de Spearman et Kendall. Les méthodes *point-wise*, et particulièrement RFR, obtiennent de meilleurs résultats que l'approche *pair-wise* pour MED et HIGH. La comparaison des méthodes *pair-wise* et *point-wise* ne nous permet pas de décider d'une approche à privilégier.

4.1.2. *Influence de l'activité et de l'ambiguïté*

Les mots-clés de MED et HIGH ont des niveaux d'ambiguïté variés. La Figure 3 montre que près de la moitié des mots-clés de MED ont moins de trois significations différentes, alors que le nombre de significations des mots-clés de HIGH est moins pi-

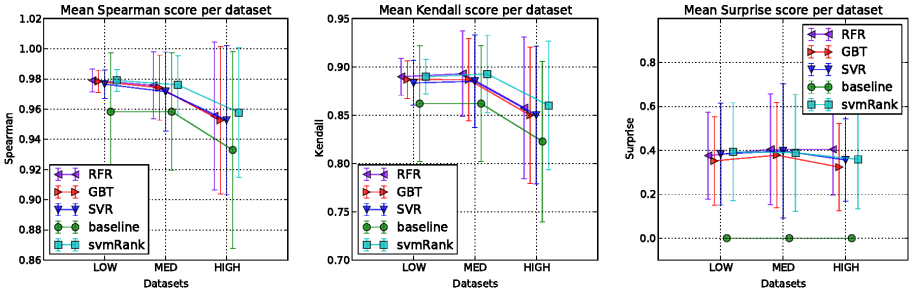


**Figure 2.** *Comparaison des approches : naïve, point-wise (*RFR,GBT,SVR*) and pair-wise (*SVMRANK*) avec le coefficient de corrélation de : Spearman (à gauche), Kendall (au centre), et le score de surprise (à droite).*

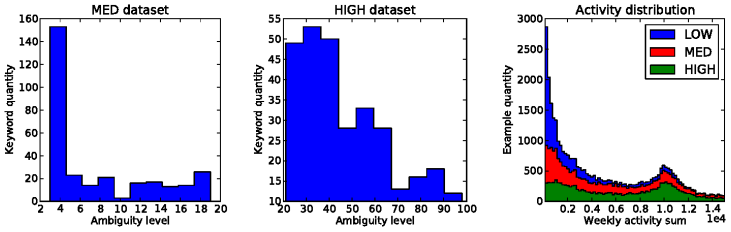


**Figure 3.** *A gauche et au centre : Distribution des niveaux d'ambiguïté (cf. section 4). Sur la droite, distribution de la somme de l'activité pendant la semaine d'observation (empilé de* LOW *(en haut) à* HIGH *(en bas)). Cet histogramme présente 35294 des 45900 exemples utilisés pour les méthodes point-wise (ie. paires mot-clé date)*

qué. Au sujet de l'activité observée sur la période d'évaluation (présentée sur la partie droite de la Figure 3) nous notons que LOW contient moins de mots-clés très actifs que MED et HIGH. En effet $50,6\%$ des mots-clés de LOW ont une activité comprise entre 0 et 500 contributions par semaine, contre $12\%$ pour MED et $5\%$ pour HIGH.

L'activité associée à un mot-clé est mesurée par le descripteur `activity`. Pour chaque exemple un modèle entraîné prédit un rang que nous pouvons comparer avec le rang réel pendant la période d'évaluation. Ces deux informations sont combinées pour produire une carte. Une carte présentant un prédiction parfaite correspond à la fonction identité. La Figure 4 présente des cartes de ce type pour les trois jeux de

données. Une toute première observation est que les mots-clés ayant les plus grands niveaux d'activité (*ie.* rangs 0 à 60) sont systématiquement mieux ordonnées que les autres. Concernant MED où l'activité est répartie plus uniformément, nous constatons que les mots-clés ayant une activité moyenne présentent des erreurs plus faibles (*i.e.* sont plus proche de la ligne identité) que les mots-clés faiblement actifs, quel que soit le rang. Enfin nous notons que, quelque soit le rang, les erreurs d'ordonnancement les plus importantes ont lieu pour les éléments au delà du rang 200.

Le niveau d'ambiguïté d'un mot-clé est mesuré par le degré sortant de sa page d'homonymie sur Wikipédia. Tous les mots-clés de LOW ont une ambiguïté nulle, nous considerons uniquement MED et HIGH. Comme précédemment nous produisons des cartes que nous utilisons pour étudier l'influence de l'ambiguïté sur l'erreur d'ordonnancement. La Figure 5 présente ces cartes pour trois méthodes : SVMRANK, RFR et la méthode naïve. Pour MED nous observons que les mots-clés classés entre 0 et 60 sont plus ambigus que les mots-clés classés entre 60 et 300 avec un niveau d'erreur comparable. L'observation la plus remarquable est que les mots-clés ayant des niveaux d'ambiguïté extrêmes sont systématiquement mal ordonnés (les plus éloignés de l'identité) quelle que soit la méthode, ou le jeu de données. Enfin dans la zone la plus difficile du classement, entre 200 et 300, les mots-clés les mieux classés ont généralement un niveau d'ambiguïté moyen. Nous observons que les mots-clés avec un niveau d'ambiguïté très faible ou très fort sont les plus difficiles à ordonner ; nous prévoyons donc de définir des descripteurs qui utilisent cette information.

#### 4.1.3. *Utilisation pour un grand nombre de mots-clés*

Nous avons évalué la robustesse de notre approche pour des ensembles de mots-clés arbitrairement grands. Pour ce faire nous étudions l'évolution des résultats de GBT, une méthode *point-wise* rapide, lorsque le nombre de mots-clés à ordonner augmente. A chaque itération nous ajoutons 100 mots-clés, un nouveau modèle est appris puis évalué. Nous exécutons 13 itérations de façon à aboutir à 1300 mots-clés. Cette expérience montre que les résultats sont stables au regard du nombre de mots-clés à classer, en effet le coefficient de corrélation du rang de Spearman augmente de 0.9737 pour 100 mots-clés à 0.9781 pour 600 mots-clés, puis décroît à 0.9765 pour 1300 mots-clés.

## 5. Conclusion

Nous avons étudié dans cet article le problème de l'ordonnancement des mots-clés en fonction de leurs activités à venir sur un réseau social en ligne, et ainsi avons reformulé le problème de prédiction de l'activité en utilisant le paradigme de l'apprentissage d'ordonnancement. Nous pouvons ainsi comparer des méthodes *point-wise* classique et une méthode *pair-wise*. Nous avons également présenté de façon générique les réseaux sociaux en ligne, sans tenir compte du graphe des ces utilisateurs, une information rarement disponible. Nous avons étudié l'influence de l'activité et de l'ambiguïté sur l'erreur de prédiction de différentes méthodes pour chaque classe

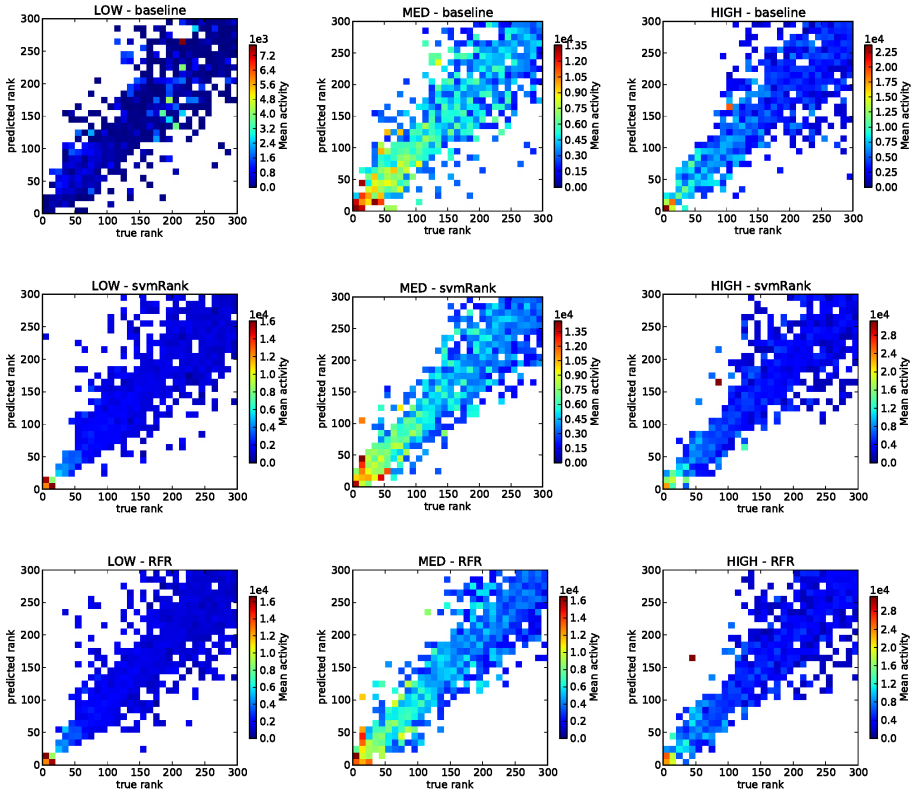


**Figure 4.** *Erreur d'ordonnancement (groupé par 10) en fonction de l'activité pendant la période d'observation. Méthode utilisée :* RFR *(en bas) ;* SVMRANK *(au milieu) ; méthode naïve (en haut).*

d'ordonnancement {*point-wise*, *pair-wise*}, deux facteurs ignorés dans les études précédentes. Nos résultats ne permettent pas de décider d'une méthode (*point-wise* ou *pair-wise*) à privilégier. Cependant nous observons que le nombre de mots-clés n'est pas corrélé avec la capacité prédictive du système et que :

– Les mots-clés avec une activité plus faible sont plus difficiles à ordonner ;

– Les mots-clés avec des niveaux d'ambiguïté extrêmes sont plus difficiles à ordonner quand leur activité n'est pas forte.

Afin de rendre nos résultats reproductibles nous avons publié les jeux de données crées pour ce travail. Enfin nous souhaitons poursuivre le travail sur l'ambiguïté, et le compléter par une étude de l'influence inter mots-clés dans le problème de prédiction de l'activité.

## 6. Bibliographie


Asur S., Huberman B. A., « Predicting the Future with Social Media », *CoRR*, 2010.

Breiman L., « Random forests », *Machine learning*, vol. 45, n° 1, p. 5-32, 2001.

Buhmann M. D., *Radial basis functions : theory and implementations*, vol. 12, Cambridge university press, 2003.

Cao Y., Xu J., Liu T.-Y., Li H., Huang Y., Hon H.-W., « Adapting ranking SVM to document retrieval », *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, p. 186-193, 2006.

Chapelle O., Chang Y., Liu T.-Y., « Future directions in learning to rank », *Journal of Machine Learning Research*, vol. 14, p. 91-100, 2011.

Cohen W. W., Schapire R. E., Singer Y., « Learning to order things », *Journal of Artificial Intelligence Research*, vol. 10, n° 1, p. 243-270, 1999.

Crammer K., Singer Y., « Pranking with Ranking », *Advances in Neural Information Processing Systems (NIPS 14)*, MIT Press, p. 641-647, 2001.

Drucker H., Burges C. J., Kaufman L., Smola A., Vapnik V., « Support vector regression machines », *Advances in neural information processing systems*, vol. 1, p. 155-161, 1997.

Duan Y., Jiang L., Qin T., Zhou M., Shum H.-Y., « An empirical study on learning to rank of tweets », *Proceedings of the 23rd International Conference on Computational Linguistics*, Association for Computational Linguistics, p. 295-303, 2010.

Freund Y., Iyer R., Schapire R. E., Singer Y., « An efficient boosting algorithm for combining preferences », *Journal of Machine Learning Research*, 2003.

Friedman J. H., « Greedy function approximation : a gradient boosting machine », *Annals of Statistics*, vol. 1, p. 1189-1232, 2001.

Ginsberg J., Mohebbi M., Patel R., Brammer L., Smolinski M., Brilliant L., « Detecting influenza epidemics using search engine query data », *Nature*, vol. 457, n° 7232, p. 1012-1014, 2008.

Hong L., Dan O., Davison B., « Predicting popular messages in twitter », *Proceedings of the 20th international conference companion on World wide web*, ACM, p. 57-58, 2011.

Joachims T., « Optimizing search engines using clickthrough data », *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, p. 133-142, 2002.

Kendall M. G., « Rank correlation methods. », 1948.

Kleinberg J., « Bursty and hierarchical structure in streams », *Data Mining and Knowledge Discovery*, vol. 7, n° 4, p. 373-397, 2003.

Lehmann J., Gonçalves B., Ramasco J., Cattuto C., « Dynamical classes of collective attention in twitter », *Proceedings of the 21st international conference on World Wide Web*, ACM, p. 251-260, 2012.

Liu T.-Y., « Learning to Rank for Information Retrieval », *Foundation Trends in Information Retrieval*, vol. 3, n° 3, p. 225-331, 2009.

Metzler D., Cai C., « USC/ISI at TREC 2011 : Microblog Track. », *TREC*, 2011.

Pan C., Mitra P., « Event detection with spatial latent dirichlet allocation », *Proceedings of the 11th annual international ACM/IEEE joint conference on Digital libraries*, ACM, p. 349-358, 2011.

Petrovic S., Osborne M., Lavrenko V., « Rt to win ! predicting message propagation in twitter », *5th ICWSM*, 2011.

Sakaki T., Okazaki M., Matsuo Y., « Earthquake shakes Twitter users : real-time event detection by social sensors », *Proceedings of the 19th international conference on World wide web*, ACM, p. 851-860, 2010.

Tsur O., Rappoport A., « What's in a hashtag ? : content based prediction of the spread of ideas in microblogging communities », *Proceedings of the fifth ACM international conference on Web search and data mining*, ACM, p. 643-652, 2012.

Uysal I., Croft W. B., « User oriented tweet ranking : a filtering approach to microblogs », *Proceedings of the 20th ACM international conference on Information and knowledge management*, ACM, p. 2261-2264, 2011.

Valizadegan H., Jin R., Zhang R., Mao J., « Learning to Rank by Optimizing NDCG Measure », *Advances in Neural Information Processing Systems (NIPS 22)*, p. 1883-1891, 2009.

Wolfe D. A., Hollander M., « Nonparametric statistical methods », *Nonparametric statistical methods*, 1973.

Yang J., Leskovec J., « Patterns of temporal variation in online media », *Proceedings of the fourth ACM international conference on Web search and data mining*, ACM, p. 177-186, 2011.

Yao J., Cui B., Huang Y., Jin X., « Temporal and social context based burst detection from folksonomies », *Proc. of AAAI*, p. 1474-1479, 2010.

Zaman T. R., Herbrich R., Van Gael J., Stern D., « Predicting information spreading in twitter », *Workshop on Computational Social Science and the Wisdom of Crowds, NIPS*, Citeseer, 2010.

Zhang X., He B., Luo T., Li B., « Query-biased learning to rank for real-time twitter search », *Proceedings of the 21st ACM international conference on Information and knowledge management*, ACM, p. 1915-1919, 2012.

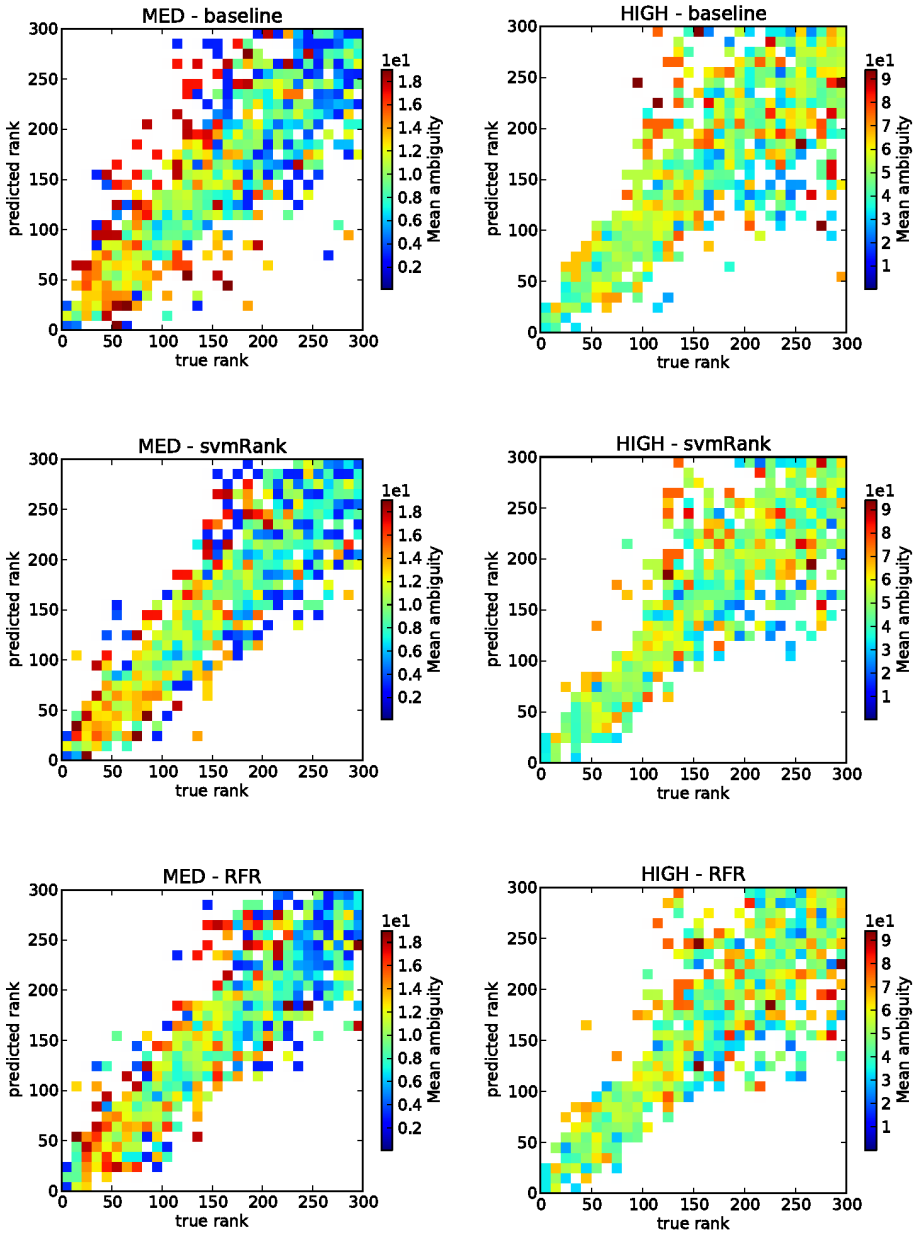


**Figure 5.** *Erreur d'ordonnancement (groupé par 10) en fonction de l'ambiguïté pendant la période d'observation. Méthode utilisée :* RFR *(en bas) ;* SVMRANK *(au milieu) ; méthode naïve (en haut).*