Recommandation par combinaison de filtrage collaboratif et d'analyse de sentiments

Mickaël Poussevin[1,2] — Elie Guardia-Sebaoun[2] — Vincent Guigue[2] — Patrick Gallinari[2]

[1] *Thales Communications & Security*
[2] *Laboratoire d'informatique de Paris 6 (LIP6), CNRS UMR 7606 Sorbonne-Universités, UPMC, Paris 6*

RÉSUMÉ. *Les domaines de la recommandation et de la classification de sentiments sont restés complètement disjoints jusqu'ici: d'un coté, la recommandation exploite les matrices d'interaction entre les utilisateurs et les produits, sous la forme de notes en faisant l'impasse sur les données textuelles, de l'autre, la fouille d'opinion exploite les revues/notes de consommateurs pour construire des modèles d'analyse de documents. Nous proposons dans cet article un modèle exploitant aussi des données d'interaction textuelles présentes dans les revues de consommateurs pour construire un modèle de recommandation novateur et performant.*

ABSTRACT. *Sentiment classification and recommender systems were until recently completely disjoint domains. Recommender systems exploit the users/items/rates matrix with omitting the available text information. Sentiment classification exploits text reviews and consumers rates to build models for document analysis. In this article we propose an unified model exploiting both text and user, items and rates to build a new and efficient recommender system.*

MOTS-CLÉS : *Recommandation, Classification de Sentiments*

KEYWORDS: *Recommender Systems, Opinion Mining*

CORIA 2014, pp. 27–42, Nancy, 18-21 mars 2014

1. Introduction

L'avènement du web participatif, où les utilisateurs écrivent des articles, des commentaires et des opinions, a engendré plusieurs domaines de recherches dans les années 2000 dont ceux de la recommendation et de la classification de sentiments. La recommandation collaborative se focalise sur la matrice (utilisateurs/produits/notes). En résumant, les notes attribuées par un utilisateur dans le passé forment son profil et la comparaison des profils permet de générer des propositions (les films qui ont été appréciés par des utilisateurs ayant des profils proches). A l'inverse, la classification de sentiment modélise les documents textuels pour caractériser leurs polarités en analysant des mots ou des groupes de mots qui seront considérés comme des marqueurs d'opinion.

Les rapprochements entre ces domaines de recherche sont très peu nombreux. Coté recommandation, les méthodes basées sur le contenu utilisent généralement des mots-clés ou des ontologies pour définir une typologies des items à recommander qui améliorera les performances du système (Adomavicius et Tuzhilin, 2005). Très peu de systèmes s'appuient sur des données textuelles et un système d'extraction d'information est en général inséré entre le texte et la tâche de recommandation elle-même. A l'inverse, la fouille d'opinion fait généralement l'impasse sur la modélisation des utilisateurs car le but est de construire un modèle universel et transférable d'analyse des textes (Pang et Lee, 2008). La plus grande partie des applications concernent les sondages et les problèmes de e-réputation et requiert un traitement quantitatif des documents textuels, les enjeux scientifiques sont centrés sur l'efficacité et la capacité de généralisation des modèles pour faire face à n'importe quel domaine et n'importe quel media (Blitzer *et al.*, 2007 ; Mejova et Srinivasan, 2012). Nous n'avons trouvé quelques systèmes de recommandation exploitant du texte. (Ganu *et al.*, 2009) propose un système multi-aspects où le texte des revues de restaurants permet d'extraire les points (présentation, cuisine,...) qui ont été appréciés ou pas pour améliorer la recommandation en se focalisant sur les aspects importants pour l'utilisateur. Deux autres études récentes se positionnent à l'intersection de la recommandation et de la fouille d'opinion. Comme dans l'article précédent, (Poirier *et al.*, 2010a) a eu recours à la classification de sentiments avant la phase de recommandation pour annoter des textes. Nous nous positionnons plus près des travaux (McAuley et Leskovec, 2013b) où le texte des revues d'un utilisateur est utilisé pour afiner son profile : l'idée est de combiner les informations de notes et des traces écrites du passé pour améliorer la recommandation.

Cet article est centré sur la problématique de recommandation collaborative : nous proposons une approche novatrice mêlant des techniques issues de la classification de sentiments et du filtrage collaboratif. Par rapport aux travaux cités précédemment, nous montrons ici l'intérêt de travailler sur le texte brut (par rapport à un espace latent) et nous introduisons des modèles différents pour les textes positifs et négatifs. En notant respectivement u et i les utilisateurs et les produits, nous distinguerons les notes $r_{u,i}$ et les documents $d_{u,i}$ associés aux couples (u, i). Nous cherchons à construire un

modèle $f(u,i)$ qui soit une bonne approximation de $r_{u,i}$. Ce modèle aura la forme suivante :

$$\begin{aligned} f(u,i) \quad = \quad & \lambda_0\phi_0 + && \text{historique moyen} && [1] \\ & \lambda_1\phi_1(u) + && \text{historique de l'utilisateur} && [2] \\ & \lambda_2\phi_2(i) + && \text{historique du produit} && [3] \\ & \lambda_3\phi_3(u,i) + && \text{historique joint} && [4] \\ & \lambda_4\phi_4(d_{u,i}) && \text{documents associés à l'utilisateur et au produit} && [5] \end{aligned}$$

La composante [1] représente le biais moyen : c'est aussi le modèle le plus basique de recommandation. Les composantes [2] et [3] permettent de construire un système de référence : il est établi que la note moyenne donnée à un produit dans le passé (indépendamment des utilisateurs) et la note moyenne d'un utilisateur (indépendamment des produits) sont des caractéristiques essentielles pour estimer $r_{u,i}$. La composante [4] a fait l'objet de nombreuses études (Adomavicius et Tuzhilin, 2005) et repose souvent sur les techniques de factorisations matricielles permettant de prédire les valeurs manquantes de la matrice $R = \{r_{u,i}\}$ (Wang et Zhang, 2013). La dernière composante est nouvelle : il s'agit de la contribution de cet article.

L'usage du texte comme donnée d'interaction dans les problèmes de recommandation reste marginal. C'était une des motivations lors de l'introduction des tenseurs dans les systèmes de recommandation : la gestion de nouvelles dimensions permettaient d'envisager une factorisation sur un tenseur (utilisateurs/produits/notes/revues) (Rendle et Schmidt-Thieme, 2010). Cependant, au delà de la possibilité théorique, nous n'avons pas trouvé d'application concrète dans la littérature. Les données textuelles ont toujours posé des problèmes spécifiques de par leur grande dimension, leur parcimonie et le manque de sémantique dans les représentations en sacs de mots : cela nous conforte dans l'idée de les traiter dans une sous-fonction séparée (le terme [5] de l'équation précédente).

Après avoir fait le point sur l'état de l'art en recommandation et sur les techniques d'analyse de sentiments en section 2, nous donnerons les détails de notre modèle en section 3. Nous proposons ensuite une série d'expériences démontrant l'intérêt de notre approche (section 4).

2. État de l'art

Notre travail se situe au confluent de deux domaines. Dans cette partie nous décrivons d'abord la recommandation avant de passer à l'analyse de sentiments.

2.1. *Recommandation*

Il est possible de définir le problème de la recommandation de deux manières différentes, la prédiction de notes ou la génération de listes d'items ordonnées. La prédiction de notes consiste à estimer la note que donnerait un utilisateur à un item (modélisant ainsi son intérêt pour ce dernier). La qualité est mesurée par la proximité entre la note prédite et la note réellement donnée (e.g. Erreur des moindres carrés). C'est le paradygme utilisé par le moteur GroupLens (Resnick *et al.*, 1994) ou dans le cadre du challenge Netflix (Bennett et Lanning, 2007 ; Koren, 2008) et c'est aussi l'approche que nous avons retenue. Dans le cas de l'ordonnancement, la recommandation est une liste d'items ordonnée par ordre d'intéret pour l'utilisateur. Seul les K items les plus pertinents sont renvoyés vers l'utilisateur (Breese *et al.*, 1998 ; McLaughlin et Herlocker, 2004). Il s'agit de l'implémentation utilisée sur les sites de e-commerce. Le modèle est alors évalué à l'aide de mesures TopK (rappel et précision) ou d'aire sous la courbe ROC.

Quatre principales approches de recommandation sont généralement distinguées dans la littérature : *content based, knowledge based, collaborative filtering* ainsi que les modèles hybrides. Plusieurs possibilités d'hybridation entre les différents systèmes sont présentés dans (Burke, 2007). Cet article se situe dans les techniques de filtrage collaboratif, qui est l'approche la plus répandue aujourd'hui. L'idée sous-jacente est que deux utilisateurs ayant des historiques proches (i.e. les utilisateurs ont noté les mêmes produits de manière similaire) tendent à avoir le même avis sur un item. Ainsi, cette méthode ne demande aucune connaissance des objets, juste l'historique des utilisateurs. Ceci présente toute fois une limite dans le cas de problèmes de démarrage à froid ou d'historiques trop petits (Schafer *et al.*, 1999). Proposée pour la première fois dans le cadre de la plateforme GroupLens (Resnick *et al.*, 1994), cette approche s'est ensuite divisée en deux sous-catégories : les méthodes à mémoire (*memory based*) et les méthodes à modèles (*model based*). La première, semblable à un algorithme des k-plus proches voisins, s'est complexifiée au cours du temps (Koren, 2008). Dans le second cas, des modèles de prédictions sont appris sur des données d'apprentissage (Bennett et Lanning, 2007). Les algorithmes les plus utilisés sont la factorisation matricielle et les RBM (*Restricted Boltzman Machine*) (Koren, 2008 ; Koren et Bell, 2011).

Les méthodes de factorisation matricielle permettent de représenter les utilisateurs et les items dans le même espace latent, décrivant ainsi de la même façon les caractéristiques de l'item et les goûts de l'utilisateur (qu'ils soient positifs ou négatifs). Ces descripteurs peuvent par exemple, dans le cas de films, représenter le genre ou le public auquel il est destiné (Koren *et al.*, 2009). Cette méthode permet, d'une part d'obtenir une meilleure représentation des données, et d'autre part, de faire intervenir un biais sur l'utilisateur (un utilisateur peut être plus sévère qu'un autre) ou sur les produits (ce qui équivaut à prendre en compte sa réputation) ou encore des information contextuelles (e.g. localisation, âge et sexe de l'utilisateur).

Contrairement à l'approche collaborative, les méthodes basées sur le contenu (*content based*) utilisent des descripteurs des objets et l'historique de l'utilisateur concerné (Pazzani et Billsus, 2007). Ces méthodes ont l'avantage de permettre facilement l'ajout de nouveaux items à la recommandation. Un item est proposé à un utilisateur s'il est proche de son profil (i.e. s'il est similaire aux items qu'il a précédemment aimés). Les descripteurs des items sont généralement représentés par un vecteur de mots-clés repondérés en fonction de l'importance de ces mots (Adomavicius et Tuzhilin, 2005 ; Poirier *et al.*, 2010b). Ces méthodes permettent de prendre plus facilement en compte les meta-données (e.g. dans le cadre du challenge Netflix, le genre du film, le réalisateur ou encore le casting (Bennett et Lanning, 2007)). Ces méthodes connaissent cependant des limites : il est impossible de proposer une recommandation personnalisée à un nouvel utilisateur. De plus, ces modèles sont intrinsèquement très dépendants de la qualité des descripteurs fournis.

2.2. *Analyse de sentiments*

Dans cet article, nous voulons démontrer le potentiel des données textuelles brutes pour améliorer la recommandation. En d'autres termes, il s'agit d'utiliser les mots employés par un utilisateur pour affiner son profil. La tâche est différente de la classification de sentiments (Pang *et al.*, 2002), mais les techniques sont très proches. Nous allons modéliser les mots utilisés par les auteurs pour caractériser ce qu'ils aiment ou pas à la manière d'un algorithme *naive Bayes*, par contre, nous ne chercherons pas à prédire la polarité d'un texte mais simplement à trouver les profils qui utilisent le même style pour affiner la recommandation.

Si l'analyse de sentiment a été un domaine porteur sur la dernière décennie (Pang et Lee, 2008), il a pris toute son ampleur avec l'apparition des plateformes sociales et de microblogging. La problématique du transfert est alors devenue centrale : il s'agit d'exploiter les revues annotées pour contruire des modèles de sentiments mais aussi de vérifier leur efficacité sur de nouveaux domaines (Blitzer *et al.*, 2007) ou de nouveaux media (Mejova et Srinivasan, 2012 ; Guardia Sebaoun *et al.*, 2013) pour pouvoir sonder automatiquement le web participatif. Ce besoin de généralisation des modèles est antinomique à la personnalisation requise dans notre application. Parmi les applications d'analyse de sentiments, nous nous rapprochons plutôt de la détection de spams d'opinion qui nécessite une caractérisation du comportement et du style des auteurs de fausses revues (Mukherjee *et al.*, 2012).

2.3. *Utilisation des revues textuelles dans la recommandation*

Comme nous l'avons dit en introduction, il n'existe à notre connaissance que peu d'articles utilisant des textes rédigés dans la recommandation. (Ganu *et al.*, 2009) et (Poirier *et al.*, 2010a) tirent parti des revues de consommateurs postées sur le web participatif. Dans les deux cas, la phase de classification de sentiments est antérieure à la recommandation elle-même. La proposition de (Poirier *et al.*, 2010a) consiste sim-

plement à créer un corpus évalué en sentiments à partir de textes libres pour pouvoir appliquer des techniques de recommandation. L'approche de (Ganu *et al.*, 2009) mise sur une vision multi-aspects des revues de consommateurs dans le domaine de la restauration. Le module de classification de sentiments permet de déterminer ce que les auteurs ont aimé ou pas : la présentation des plats, de la table ou la nourriture. Chaque revue est traitée pour remplir un formulaire de notations qui permet d'améliorer la recommandation. En effet, le système peut alors différentier les utilisateurs qui portent plus d'attention à certains aspects du restaurant.

Notre approche est différente dans cet article : nous souhaitons utiliser directement le texte pour mieux caractériser l'auteur de la revue. Le vocabulaire utilisé nous permettra de nous focaliser sur les aspects qui intéressent l'auteur, à la manière de (Ganu *et al.*, 2009), tandis que les informations de style permettront d'affiner la recommandation en mettant en avant les utilisateurs proches de l'auteur. Notre système est plus proche de (McAuley et Leskovec, 2013b) : mais alors que cette approche repose sur un traitement des données textuelles en variables latentes, nous démontrons l'intérêt d'utiliser directement le texte brut. La variante de LDA (Latent Dirichet Allocation) utilisée dans (McAuley et Leskovec, 2013b) vise encore une fois l'extraction des aspects de la revue sans tirer parti du style de l'auteur.

3. Modèles

Nous nous plaçons dans la cadre de la recommandation collaborative et nous cherchons à prédire les notes manquantes. Nous considérons des données sous la forme d'un quadruplet $(u, i, r_{u,i}, d_{u,i})$ où u et i sont respectivement l'index de l'utilisateur et de l'objet ou *item*, $r_{u,i}$ est la note laissée par l'utilisateur u à l'objet i et $d_{u,i}$ la revue textuelle accompagnant cette note. Par concision, nous utiliserons parfois le couple (u, i) à la place du quadruplet. La prédiction de la note données par l'utilisateur u à l'objet i par un modèle est notée $f(u, i)$. Nous utiliserons le modèle composite décrit en introduction pour construire notre prédiction, cette section donne les détails des différents sous-modèles mis en oeuvre. Nos bases de données sont séparées en trois groupes : m_{app} qui permet d'apprendre les sous-modèles, m_{val} qui permet de régler les hyper-paramètres d'agrégation λ et m_{test} pour évaluer les performances.

3.1. *Evaluations*

L'objectif étant de prédire le mieux possible la note donnée par un utilisateur à un objet, le critère utilisé pour l'évaluation des modèles est l'erreur quadratique moyenne sur l'ensemble des m_{test} critiques de test, notée MSE (*Mean Squared Error*) :

$$MSE = \frac{1}{\#m_{test}} \sum_{(u,i) \in m_{test}} (r_{u,i} - f(u,i))^2 \qquad [6]$$

Cet article s'intéressant également à la classification de sentiments, nous proposons une métrique d'évaluation des performances issue de ce domaine. Dans ce cas de figure, les revues ayant une note ambigüe (3/5) sont éliminées de l'ensemble de test et les notes positives et négatives sont regroupées dans deux classes : $c_+ = \{(u,i)|r_{u,i} > 3\}, c_- = \{(u,i)|r_{u,i} < 3\}$. L'enjeu est de mesurer la capacité du système à détecter ce que l'utilisateur aime ou pas sans proposer de quantification en calculant le pourcentage d'erreurs de classification :

$$Err_S = \frac{1}{\#m_{test}^{r_{u,i}\neq 3}} \left(\#\{(u,i) \in c_+ | f(u,i) < 3\} + \#\{(u,i) \in c_- | f(u,i) > 3\}\right) \quad [7]$$

3.2. *Modèles de référence : historiques des notes*

Nous proposons comme modèles de référence les trois approches basées sur l'historique : global, utilisateur et objet. Ces modèles simples estiment respectivement la note moyenne sur l'ensemble des critiques d'apprentissage, sur l'ensemble des critiques par utilisateur et par objet, comme présenté dans les équations [8], [9] et [10].

$$\phi_0 = \frac{1}{\#m_{app}} \sum_{(u,i)\in m_{app}} r_{u,i} \quad [8]$$

$$\phi_1(u) = \frac{1}{\#m_{app}^{(u)}} \sum_{(u,i')\in m_{app}} r_{u,i'} \text{ moyenne des notes données par } u \quad [9]$$

$$\phi_2(i) = \frac{1}{\#m_{app}^{(i)}} \sum_{(u',i)\in m_{app}} r_{u',i} \text{ moyenne des notes données à } i \quad [10]$$

En test ou validation, dans le cas où l'utilisateur et l'item à noter ne possède pas d'historique sur la base d'apprentissage, nous utilisons ϕ_0 comme prédiction.

3.3. *Factorisation matricielle*

Pour estimer à la fois les critères de notations des utilisateurs et les qualités d'un objet, nous utilisons la factorisation matricielle. L'idée sous-jacente est de regrouper les informations identiques, c'est-à-dire de trouver les personnes qui ont des notations proches sur les items. Chaque utilisateur est alors représenté par un vecteur υ_u et chaque objet par un vecteur ι_i : ces vecteurs sont de faibles dimensions par rapport aux nombres d'items et d'utilisateurs. La prédiction de la note $\phi_3(u,i)$ est obtenue en calculant le produit scalaire entre les représentations de u et i :

$$\phi_3(u,i) = \upsilon_u.\iota_i \quad [11]$$

L'apprentissage de ces représentations optimise l'erreur quadratique moyenne, sur l'ensemble des critiques de la base d'entraînement [12] avec une régularisation $\mathcal{L}_2$ sur les paramètres pour contrer un possible sur-apprentissage.

$$\{\upsilon_u, \iota_i\}^\star = \underset{\upsilon_u, \iota_i}{\operatorname{argmin}} \sum_{(u,i)} (r_{u,i} - \upsilon_u.\iota_i)^2 + \alpha_\upsilon \|\upsilon_u\|^2 + \alpha_\iota \|\iota_i\|^2 \qquad [12]$$

Les compromis de régularisation α sont optimisés en validation croisée sur les données d'apprentissage.

3.4. *Modèles de traitement du texte*

Nous décrivons les méthodes à variables latentes, qui cherchent à extraire les aspects des revues automatiquement puis nous détaillons notre approche basée sur l'analyse des documents bruts.

3.4.1. *Allocation latente de Dirichlet*

Dans (McAuley et Leskovec, 2013b), les auteurs utilisent une variante de *LDA*, (*Latent Dirichlet Allocation*) pour projetter les textes dans l'espace latent. Ils proposent d'intégrer directement la représentation latente des documents dans l'algorithme de factorisation matricielle présenté précédemment, en utilisant une technique d'optimisation alternée.

Nous proposons d'utiliser *LDA* comme une fonction de projection dans l'espace latent ψ. En notant $d_{u,*}$ et $d_{*,i}$ les concaténations respectives de l'ensemble des textes de l'utilisateur u ou sur l'objet i, nous définissons le modèle de concordance thématique suivant :

$$\phi_{L4}(u,i) = \psi(d_{u,*}) \cdot \psi(d_{*,i}) \qquad [13]$$

La fonction ψ est apprise sur l'ensemble d'apprentissage et nous utilisons les matrices de probabilités calculées pour projeter l'ensemble des documents.

3.4.2. *Information textuelle brute*

Après avoir passé l'ensemble des données d'apprentissage en sacs de mots, nous utilisons un modèle bayesien naïf pour représenter respectivement : les revues de l'utilisateur (b_u), les revues positives/négatives de u ($b_u^{(+)}$ et $b_u^{(-)}$) et les revues associées à l'item i (b_i, $b_i^{(+)}$ et $b_i^{(-)}$). La prédiction basée sur le texte brute est calculée comme une combinaison linéaire des comparaisons entre les modèles de sentiments de l'utilisateur u et de l'item i :

$$\begin{aligned}\phi_{T4}(u,i) \quad = \quad & \lambda_{t1}\cos(b_u, b_i) \quad + \lambda_{t2}\cos(b_u, b_i^{(+)}) \quad + \lambda_{t3}\cos(b_u, b_i^{(-)}) + \\ & \lambda_{t4}\cos(b_u^{(+)}, b_i) \quad + \lambda_{t5}\cos(b_u^{(+)}, b_i^{(+)}) \quad + \lambda_{t6}\cos(b_u^{(+)}, b_i^{(-)}) + \\ & \lambda_{t7}\cos(b_u^{(-)}, b_i) \quad + \lambda_{t8}\cos(b_u^{(-)}, b_i^{(+)}) \quad + \lambda_{t9}\cos(b_u^{(-)}, b_i^{(-)}))\end{aligned} \qquad [14]$$

Les coefficients λ sont optimisés sur les données de validation.

4. Expériences

4.1. *Données*

Les données utilisées sont des revues anglophones annotées extraites des sites *ratebeer.com*[1] et *amazon.com* [2]. Des bases de tailles différentes ont été créées à partir des corpus en sélectionnant des sous-ensembles d'utilisateurs et d'items comme le montre le tableau 1. Les critiques sont ensuite réparties aléatoirement en trois ensembles, un d'entrainement qui contient 80% des données, puis 10% en validation et 10% en test.

Nom	#Utilisateurs	#Objets	#Entraînement	#Validation	#Test
RB_U50_I200	52	200	7200	900	906
RB_U500_I2k	520	2000	388200	48525	48533
RB_U5k_I20k	5200	20000	1887608	235951	235960
RB_U30k_I110k	29265	110364	2339296	292412	292415
A_U200_I120	213	122	984	123	130
A_U2k_i1k	2135	1225	31528	3941	3946
A_U20k_I12k	21353	12253	334256	41782	41791
A_U210k_I120k	213536	122538	1580576	197572	197574
A_U2M_I1M	2135360	1225387	4642808	580351	580357

Tableau 1 – Tailles des bases de données utilisées.

4.2. *Apprentissage des modèles*

Les modèles de biais ϕ_0, ϕ_1 et ϕ_2 sont estimés sur la base d'apprentissage. Le modèle de factorisation matricielle ϕ_3 est appris par descente de gradient stochastique avec une régularisation $\mathcal{L}_2$ sur les données d'apprentissage. La factorisation matricielle requiert des valeurs α pour les compromis de régularisation qui sont optimisées en validation croisée. Le modèle latent ϕ_{L4} et le modèle de texte brut ϕ_{T4} sont également appris sur les données d'apprentissage.

Les hyper-paramètres λ sont optimisés sur l'ensemble de validation, au sens de l'erreur quadratique moyenne de reconstruction des notes. Les λ permettent de pondérer les différents modèles et ils sont aussi utilisés à l'intérieur du modèle de texte brut pour tirer parti des différentes similarités.

4.3. *Résultats*

Nous présentons dans cette section les résultats obtenus sur les différents corpus *Amazon* et *RateBeer*. Dans les tableaux de performances 2 et 3, nous présentons d'abord les 3 références ϕ_0, $\phi_1(u)$, $\phi_2(i)$ correspondant aux différentes notes

1. Revues sur des bières, collectées par (McAuley et Leskovec, 2013a)

2. Revues sur différents types de produits vendus sur le site Amazon, collectées par (Jindal *et al.*, 2010)

moyennes. Nous nommons les modèles $\phi_3(u,i)$, ϕ_{L4} et ϕ_{T4} mais ce sont en fait des modèles composites qui intègrent les biais :

– $\phi_3(u,i)$ correspond à $\lambda_0\phi_0 + \lambda_1\phi_1(u) + \lambda_2\phi_2(i) + \lambda_3\phi_3(u,i)$
– $\phi_{L4}(u,i)$ correspond à $\lambda_0\phi_0 + \lambda_1\phi_1(u) + \lambda_2\phi_2(i) + \lambda_3\phi_3(u,i) + \lambda_4\phi_{L4}(u,i)$
– $\phi_{T4}(u,i)$ correspond à $\lambda_0\phi_0 + \lambda_1\phi_1(u) + \lambda_2\phi_2(i) + \lambda_3\phi_3(u,i) + \lambda_4\phi_{T4}(u,i)$

4.3.1. *Evaluation en recommandation*

Nous comparons les 3 modèles de référence avec un filtrage collaboratif simple puis avec un filtrage enrichi des modèles d'analyse de textes. L'ensemble des expériences, évaluées au sens des moindres carrés, est présenté dans le tableau 2.

Base	ϕ_0	$\phi_1(u)$	$\phi_2(i)$	$\phi_3(u,i)$	ϕ_{L4}	ϕ_{T4}
RB_U50_I200	0,67575	0,65325	0,20913	0,19776	**0,19208**	0,19508
RB_U500_I2k	0,56850	0,52563	0,25089	0,22377	0,22182	**0,22087**
RB_U5k_I20k	0,67744	0,58782	0,30791	0,28466	0,27193	**0,27155**
RB_U30k_I110k	0,70296	0,60644	0,34876	0,33157	0,31070	**0,30889**
A_U200_I120	1,53480	1,56583	1,49159	1,97755	1,37034	**1,34089**
A_U2k_I1k	1,53155	1,30432	1,27850	1,21357	**1,05542**	1,06147
A_U20k_I12k	1,47107	1,28584	1,23608	1,21267	1,04996	**1,04524**
A_U210k_I120k	1,50721	1,44538	1,32229	1,29709	1,15504	**1,14716**
A_U2M_I1M	1,60510	1,63127	1,49281	1,48153	1,33138	**1,32666**

Tableau 2 – Résultats des modèles sur les différentes bases en erreur quadratique moyenne sur les critiques de test.

Le tableau précédent nous permet de tirer quelques conclusions importantes : parmi les modèles de référence, $\phi_2(i)$ est nettement plus performant que les autres. Le modèle ϕ_0 semble trop pauvre. La comparaison entre $\phi_1(u)$ et $\phi_2(i)$ montre que l'avis des utilisateurs est assez uniforme sur un produit donné alors qu'un utilisateur a un avis changeant d'un item à l'autre (ce qui semble assez intuitif).

Si le filtrage collaboratif apporte un gain significatif par rapport aux références (comme cela a été montré plusieurs fois dans la littérature), il est intéressant de constater que la prise en compte du texte permet systématiquement de réduire l'erreur par rapport à ces techniques. Sur *RateBeer* , le gain est significatif (entre 2,5 et 7% d'amélioration) mais sur *Amazon* il devient très important (entre 10 et 25% de gain).

La comparaison entre les deux approches textuelles est plus serrée : même si ϕ_{T4} est souvent plus performant, les écarts sont minimes. Par contre, les traitements sont beaucoup moins lourds (notamment sur les grandes bases) car il n'y a plus besoin de calculer le modèle LDA ni de l'appliquer.

L'analyse générale des performances par rapport à la taille des bases considérées est étonnante : sur *Amazon*, les performances s'améliorent avant de repartir à la baisse sur les grandes bases tandis que sur *RateBeer*, la tendance est totalement à la hausse. Plus il y a de données pour apprendre, plus les modèles sont mauvais ! En réalité,

l'explication est simple : nous avons réalisé les plus petites expériences sur les données les plus favorables (items largement commentés et utilisateurs les plus actifs) : plus nous avançons dans les données, plus les prédictions sont délicates, ce qui explique la tendance générale à la baisse des performances.

4.3.2. *Evaluation en classification*

Le travail présenté dans cet article étant proche des techniques de classification de sentiments, nous avons utilisé une évaluation du domaine, présentée en section 3.1. Les résultats obtenus sont donnés dans le tableau 3.

Base	ϕ_0	$\phi_1(u)$	$\phi_2(i)$	$\phi_3(u,i)$	ϕ_{L4}	ϕ_{T4}
RB_U50_I200	18,75	19,11	8,21	8,91	7,62	**7,17**
RB_U500_I2k	18,75	18,60	10,73	10,26	10,16	**9,92**
RB_U5k_I20k	25,03	24,65	14,43	14,32	12,54	**12,42**
RB_U30k_I110k	26,33	25,83	16,05	15,20	13,88	**13,70**
A_U200_I120	17,50	21,67	17,50	25,83	19,23	**16,92**
A_U2k_I1k	15,94	15,16	15,92	14,10	**11,38**	11,68
A_U20k_I12k	14,74	14,47	14,19	14,24	**11,26**	**11,28**
A_U210k_I120k	14,73	15,99	14,81	14,42	12,22	**12,05**
A_U2M_I1M	14,91	16,39	15,78	15,91	13,14	**13,05**

Tableau 3 – Résultats des modèles sur les différentes bases en erreur de classification (positif/négatif) sur les critiques de test.

En classification de sentiments, le gain entre $\phi_2(i)$ et $\phi_3(u,i)$ n'est plus aussi prononcé qu'avec la MSE : si la factorisation matricielle permette d'améliorer la prédiction de la note, le gain est trop marginal pour permettre un changement de classe de sentiments. A l'inverse, la prise en compte du texte permet des gains importants : de nombreuses erreurs de classification sont alors corrigées.

Depuis (Pang *et al.*, 2002), il est acté que le texte brut est une bonne base de travail pour la classification de sentiments. Même si les écarts par rapport aux approches à variables latentes sont faibles, le modèle $\phi_{T4}(u,i)$ est très majoritairement le plus performant pour un coût de calcul bien moindre.

4.4. *Analyse des prédictions*

Afin de mieux comprendre les apports du modèle utilisant l'information textuelle dans la décision finale, nous en avons comparé les sorties des modèles ϕ_3 et ϕ_{T4}.

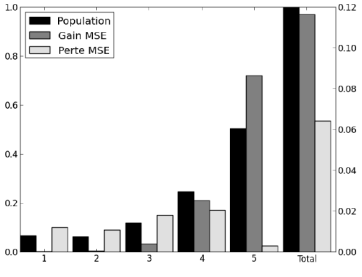
4.4.1. *Amazon.com sur A_U20k_I12k*

Nous avons analysé les différences entre les sorties de la factorisation matricielle seule et avec prise en compte du texte brut sur A_U20k_I12k et RB_U5k_I20k.

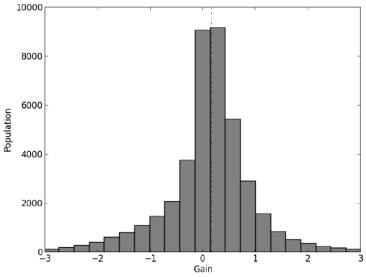
Nous avons d'abord étudié la distribution des gains à travers la formule suivante

$$G_{\text{+texte}}(u,i) = (r_{u,i} - \phi_3(u,i))^2 - (r_{u,i} - \phi_{T4}(u,i))^2 \quad [15]$$

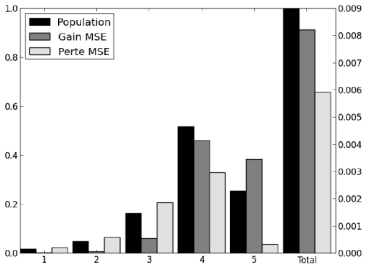
Le résultat est sans surprise, en forme de gaussienne de moyenne légèrement positive : la plupart des gains/pertes sont faibles mais il y a plus de gains que de pertes. Ces résultats sont illustrés sur les courbes 1b et 1d.



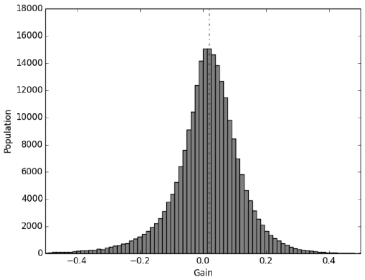
(a) **Amazon.** Population, gain en *MSE* et perte en *MSE* par note et totaux



(b) **Amazon.** Histogramme des valeurs du gain $G_{\text{+texte}}(u,i)$ pour l'ensemble des critiques de test et leur médiane (0,164179)



(c) **RateBeer.** Population, gain en *MSE* et perte en *MSE* par note et totaux



(d) **RateBeer.** Histogramme des valeurs du gain $G_{\text{+texte}}(u,i)$ pour l'ensemble des critiques de test et leur médiane (0,018727)

Figure 1 – Illustrations de la répartition des gains sur deux bases de données

Les figures 1a et 1c illustrent le gain et la perte par classe de notation en utilisant les équations suivantes :

$$Gain_{\text{MSE}} = \frac{1}{\#m_{test}} \sum_{(u,i)\in m_{test}} |G_{\text{+texte}}(u,i)|_{+} \quad \text{moyenne des gains} \quad [16]$$

$$Perte_{\text{MSE}} = \frac{1}{\#m_{test}} \sum_{(u,i)\in m_{test}} |G_{\text{+texte}}(u,i)|_{-} \quad \text{moyenne des pertes} \quad [17]$$

Note	5,0
ϕ_3	4,479377
ϕ_{T4}	4,764128

One of the best historical novels I've read This book is a wonderful tapestry of Norman/Angevin England and Wales. The characters are well-developed and complex. For example, historical treatments of King John invariably cast him as a villain, but here we see him as a character with many facets. The plot follows Joanna, or Joan, the illegitimate daughter of John, through her life from about age five to her late thirties. A reader of this book will learn much about culture clash, women, the Angevins, and England and Wales in the Middle Ages. The book is captivating – I was hardly able to put it down

Note	1,0
ϕ_3	1,296613
ϕ_{T4}	0,900824

Man ! This one gave me a hemorrhoid This is just an awful attempt at making music. This guys music literally irritates my [ears] when I hear it. What is really messed up about the whole situation is this guy is polluting the minds of the children with the poor lyrics and ignorant subject matter.

Note	1,0
ϕ_3	1,669823
ϕ_{T4}	0,700161

Not taking it back. After comparing the print quality in best mode to my HP 970 CSE inkjet in best mode from the same source there is no comparison. The HP wins in print quality hands down. The CX5200 with its pigment ink is printing unsaturated colors and not sharp in best mode on my first day of use. The HP dye based ink colors are deep and the print is super sharp. I'm not taking this machine back to the dealer for a refund because the wife says the long life durabright ink is required for her scrapbooking. The software install is buggy on an XP home machine and the software is fairly worthless as well. Fortunately my MS Picture It that came with the Dell works with the scanner.

Note	4,0
ϕ_3	2,963587
ϕ_{T4}	3,619592

Enjoy after repeated Play After spending hours actually forcing myself to listen to this CD, I have to begrudgingly admit that Alicia Keys MAY deserve some of the accolades she has received. The CD is set up so that each song compliments the one before. This is a nice album to mellow out and chill with.

Tableau 4 – Exemples de critiques où le texte apporte une meilleure classification sur A_U20k_I12k

Le résultat est plus inattendu : les corrections ont principalement lieu sur les notes 4 et 5 (qui sont majoritaires). Les notes 1, 2 et 3 sont moins bien prédites, mais comme elles sont minoritaires, la précision générale augmente. Le tableau 4 donne des exemples de corrections typiques que nous avons observés.

4.5. *Prédiction du texte des critiques*

L'utilisation conjointe du modèle de recommandation et d'un modèle de fouille d'opinion permet d'envisager de nouvelles tâches. Comme nous l'avons mentionné dans l'état de l'art, la modélisation de l'utilisateur au niveau du texte nous rapproche par exemple de la problématique de la détection de spam d'opinion. Une tâche importante de l'analyse de sentiments concerne le résumé des revues du web participatif. Avec le modèle que nous avons développé, nous sommes en mesure de proposer bien mieux : il est possible de prédire la revue qu'un consommateur écrirait sur un produit qu'il ne connaît pas encore.

Pour une revue $(u, i, r_{u,i}, d_{u,i})$, nous utilisons la prédiction $\phi_{T4}(u, i)$ pour sélectionner les critiques sur l'objet i écrites par d'autres utilisateurs u' dont la note donnée $r_{u',i}$ est proche de celle prédite. Les textes de ces critiques sont ensuite analysés pour en sélectionner les phrases contenant le plus de mots que l'utilisateur u a employés.

Nous n'avons pas approfondi le sujet en essayant de préserver une diversité dans les phrases extraites ni testé nos résultats avec les métriques classiques de résumé automatique (score rouge par exemple) mais nous avons trouvé les textes générés assez crédibles et nous en proposons une sélection dans le tableau 5.

5. Conclusion

Dans cet article, nous avons démontré l'intérêt de la prise en compte du texte dans les modèles de recommandation basés sur les revues de consommateurs. Nous avons développé plusieurs modèles et comparé des approches à variables latentes avec des approches d'analyse du texte brut : les performances sont globalement assez proches alors que le texte brut est plus simple et moins coûteux à manipuler. Nous en avons conclu qu'il était plus intéressant de travailler directement sur le texte.

Le texte permet de mieux modéliser les utilisateurs, ce qui permet d'affiner les modèles de prédiction de notes. Dans le détail, les expériences que nous avons réalisées sur des bases de données de différentes tailles issues de *ratebeer.com* et *amazon.com* montrent que l'amélioration des performances est liée à une meilleure estimation des bonnes notes (4 et 5). Comme celles-ci sont les plus nombreuses, elles permettent d'améliorer significativement l'ensemble du système.

Ce travail ouvre de nombreuses pistes à l'intersection de plusieurs domaines de recherche. Nous avons illustré les possibilités de notre système en étudiant rapidement la génération automatique et personnalisée de revues pour un utilisateur. Plus généralement, l'intégration de caractéristiques de styles dans le profil des utilisateurs permet d'envisager de nouvelles tâches comme la détection des auteurs utilisant de multiples identifiants par exemple.

Remerciements

Ce travail a été partiellement financé par les projets DIFAC (FUI 12) et AMMICO (ANR).

6. Bibliographie

Adomavicius G., Tuzhilin A., « Toward the next generation of recommender systems : A survey of the state-of-the-art and possible extensions », *IEEE trans. on Knowledge and data engineering*, vol. 17, n° 6, p. 734-749, 2005.

Bennett J., Lanning S., « The Netflix Prize », *KDD Cup Workshop 2007*, p. 3-6, 2007.

Blitzer J., Dredze M., Pereira F., « Biographies, Bollywood, Boom-boxes and Blenders : Domain Adaptation for Sentiment Classification », *ACL*, p. 440-447, June, 2007.

Breese J. S., Heckerman D., Kadie C., « Empirical Analysis of Predictive Algorithms for Collaborative Filtering », *Conference on Uncertainty in Artificial Intelligence*, p. 43-52, 1998.

Burke R., « Hybrid web recommender systems », *The adaptive web*, Springer, p. 377-408, 2007.

Texte
Great story and characters ; often mannered writing I am going to weigh in very briefly on this book. It has a great story, but it is marred by Arundhati Roy's too frequent displays of mannerism. Many of the characters are very original and the story is full of credible twists and turns, but every thirty pages or so, Ms. Roy lapses into several pages of writing in apretentious stream-of-conscience/surreal style that soon had me skimming toward the next section of narrative substance. Ms. Roy must have felt that she needed to show off how well she could write, but she ended up underscoring the fact that this was her first novel.
Prédictions
The individual stories of at least twelve characters are told and each story would be rather simple but the stories are all shuffled together with no regard for tense and this makes the book seem much more complex than it actually is.
Plus she moves forwards and backwards and sideways in time towards a central event which has been hinted at in countless ways but by the time you get to that event you are mad because all of the confusion could so easily have been avoided if she'd simply told the story, or each of the twelve stories, chronologically.
At times the repetitions and sentence fragments and other affectations become more of a hindrance than a benefit, but it seems that some Indian writers feel compelled to write in this sort of native style, and if it is inevitable, then better Roy's fairly controlled method than Rushdie's incomprehensible over-the-top method.

Texte
Has a pitch black pour with a super thick brown bubbly foam head. The head retention is just rediculous, sticks around for a long time. The aromas I got were chocolate malts, coffee, and a little bit of honey. The taste has a medium body mouthfeel to it, with a bitterish finish to it. From the first whiff you know exactly what its going to taste like. Taste like heaven.
Prédictions
The dark fruits that tend to dominate early on are still detectable (almost as if this were some weirdly lagered impy, shading into a Baltic porter) but are integrated superbly with the roasted malts, giving this one a really unique quality.
bottle, thanks to SS, black walnut color with soapy tan foam, aroma has a lot of alcohol and coffee, flavor is well balanced smoked meat, coffee, dried plum, cocoa, finish is well balanced with malty cocoa and coffee
The taste is just what I expected from DR ; a amazingly smooth silky body, tons of dark fruits, brown sugar, some dry roasted malts, coffee, and a bitter dark chocolate finish

Texte
This is a good read, it is a beautiful epic whose true force lies in the poignant details of its characters, richly detailed, woven into a wonderful tapestry. You all so get three books in one, a good bargin, and a good story for teen readers. OVERALL SCORE : (B+) READABILITY : (B), PLOT : (B-), CHARATERS : (A-), DIALOGUE : (B-), SETTING : (B+), ACTION/COMBAT : (B-), MONSTERS/ANTAGONISTS : (B-), ROMANCE : (B+), SEX : (n/a), AGE LEVEL : (PG)
Prédictions
Terry Brooks is probably the most talented of the derivative-of-Tolkien authors, a pleasant read for those who enjoy basic fantasy tales with a few original bits among the "Tough Guide to Fantasyland" plotlines.
Best of the subgenre Terry Brooks' Shannara series was one of the first doorstopper series, varying in originality and in quality but good brain-candy fun. Now the first three books of his trilogy are rereleased in a hardvcover three-in-one edition, not recommended for people with weak wrists, but for those who like good covers and big heavy tomes.

Texte
Pours a hazy reddish brown with a nice tan head. Aroma of yeast, fruits, caramel, spices, malt. Flavor is fruit, caramel, yeasts, rasins and malt. Good stuff.
Prédictions
The taste is medium sweet, with a Belgian yeast, roasted caramel malts, and some fruity notes
The nose is fairly sweet, with caramel malts, some dark fruits, Belgian yeast, and spice Appearance : The pour is a cloudy dark orange-brown with a thick and creamy, lasting, lacy beige crown
Smell is one complex mother of a dubble, quite unique, good smokeyness comes through with bacon and smoked dried oranges, sweet smokey malts, other dark but dullish fruit of plum and raisin, candied toffee notes and what I'm guessing is alot of wild yeast and a bit of cherry as well.

Tableau 5 – Prédictions de texte de critiques issues de notre modèle.

Ganu G., Elhadad N., Marian A., « Beyond the Stars : Improving Rating Predictions using Review Text Content. », *WebDB*, 2009.

Guardia Sebaoun E., Rafrafi A., Guigue V., Gallinari P., « Cross-Media sentiment Classification and Appication to Box-Office Forecasting », *OAIR*, 2013.

Jindal N., Liu B., Lim E.-P., « Finding unusual review patterns using unexpected rules », *CIKM*, p. 1549-1552, 2010.

Koren Y., « Factorization Meets the Neighborhood : A Multifaceted Collaborative Filtering Model », *ACM SIGKDD*, p. 426-434, 2008.

Koren Y., Bell R. M., « Advances in Collaborative Filtering. », *Recommender Systems*, p. 145-186, 2011.

Koren Y., Bell R., Volinsky C., « Matrix Factorization Techniques for Recommender Systems », *Computer*, vol. 42, n° 8, p. 30-37, August, 2009.

McAuley J. J., Leskovec J., « From amateurs to connoisseurs : modeling the evolution of user expertise through online reviews », *WWW*, p. 897-908, 2013a.

McAuley J., Leskovec J., « Hidden Factors and Hidden Topics : Understanding Rating Dimensions with Review Text », *ACM Conference on Recommender Systems*, p. 165-172, 2013b.

McLaughlin M. R., Herlocker J. L., « A Collaborative Filtering Algorithm and Evaluation Metric That Accurately Model the User Experience », *ACM SIGIR*, p. 329-336, 2004.

Mejova Y., Srinivasan P., « Crossing Media Streams with Sentiment : Domain Adaptation in Blogs, Reviews and Twitter. », *ICWSM*, The AAAI Press, 2012.

Mukherjee A., Liu B., Glance N., « Spotting Fake Reviewer Groups in Consumer Reviews », *ACM World Wide Web*, p. 191-200, 2012.

Pang B., Lee L., « Opinion Mining and Sentiment Analysis », *Foundations and Trends in Information Retrieval*, vol. 2, n° 1-2, p. 1-135, 2008.

Pang B., Lee L., Vaithyanathan S., « Thumbs Up ? : Sentiment Classification Using Machine Learning Techniques », *ACL Empirical Methods in NLP*, p. 79-86, 2002.

Pazzani M. J., Billsus D., « Content-based recommendation systems », *The adaptive web : methods and strategies of web personalization*, Springer-Verlag, p. 325-341, 2007.

Poirier D., Fessant F., Tellier I., « De la classification d'opinion à la recommandation : l'apport des textes communautaires », *TAL*, 2010a.

Poirier D., Fessant F., Tellier I., « Reducing the Cold-Start Problem in Content Recommendation through Opinion Classification. », *Web Intelligence*, IEEE, p. 204-207, 2010b.

Rendle S., Schmidt-Thieme L., « Pairwise Interaction Tensor Factorization for Personalized Tag Recommendation », *ACM International Conference on Web Search and Data Mining*, p. 81-90, 2010.

Resnick P., Iacovou N., Suchak M., Bergstrom P., Riedl J., « GroupLens : An Open Architecture for Collaborative Filtering of Netnews », *ACM Conference on Computer Supported Cooperative Work*, p. 175-186, 1994.

Schafer J. B., Konstan J., Riedl J., « Recommender Systems in e-Commerce », *ACM Conference on Electronic Commerce*, p. 158-166, 1999.

Wang Y.-X., Zhang Y.-J., « Nonnegative Matrix Factorization : A Comprehensive Review », *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, n° 6, p. 1336-1353, 2013.