Filtrage collaboratif sensible au contexte - Une approche basée sur LDA

Josiane Mothe, Ambinintsoa Jocelyn Rakotonirina

Filtrage collaboratif sensible au contexte

Une approche basée sur LDA

Josiane Mothe[1] , Ambinintsoa Jocelyn Rakotonirina[2]

1.ESPE, Université de Toulouse, Université de Toulouse Jean Jaurès
Institut de Recherche en Informatique de Toulouse, IRIT
UMR 5505 CNRS
118 Route de Narbonne, Toulouse, France
Josiane.Mothe@irit.fr

2. DMI, Université d'Antananarivo, Madagascar
Mathématiques Informatique et Statistique Appliquées, MISA
BP 906 Ankatso
Ambinintsoa26@outlook.com

RESUME. Les systèmes de recommandations visent à proposer aux utilisateurs des items en lien avec leur consultation en cours et qui peuvent retenir leur intérêt. L'intérêt des utilisateurs dépend du contexte dans lequel ils se trouvent. Dans ce travail, nous proposons un système hybride CBCF (Contex-aware Based Collaborative Filtering) qui combine le système de recommandations sensibles aux contextes et le filtrage collaboratif. Le contexte est ici défini comme l'objectif ou l'intention de l'utilisateur. Nous le modélisons par une approche LDA (Latent Dirichlet Allocation) qui génère un modèle de thèmes pour chaque intention. Nous avons évalué notre approche sur la collection Book-Crossing et montrons sa supériorité par rapport à plusieurs méthodes état de l'art.

MOTS-CLES : Système d'information, Recherche d'information, Accès à l'information, Système de recommandation, Latent Dirichlet Allocation, Filtrage collaboratif, Système de recommandation hybride

ABSTRACT. Recommender systems are designed to provide user with items related to their ongoing browsing and that may be of interest to them. User interest depends on the context. In this work, we propose a hybrid CBCF (Context-aware Based Collaborative Filtering) system combining context-sensitive and collaborative filtering. We define context as the objective or intent of the user. We model it by a LDA (Latent Dirichlet Allocation) approach which generates a topic model for each intention. We evaluated our approach using the Book-Crossing collection and demonstrated the superiority of our model over several state-of-the-art methods.

KEYWORDS: Information Systems, Information Retrieval, Recommender systems, Latent Dirichlet Allocation, Collaborative filtering, Hybrid recommender system.

1. Introduction

Depuis le début des années 1990, internet a changé la manière de consommer et de vendre : l'e-commerce est devenu un moyen commun de commerce. Sur un site de e-commerce, l'enjeu pour les entreprises est d'attirer plus de clients, de les aider à accéder rapidement aux items (produits, services, films, restaurants, etc.) pertinents et de transformer une visite sur le site en un achat.

Les systèmes de recommandations (SR) sont une solution pour recommander automatiquement des items aux utilisateurs qui peuvent être perdus dans un vaste choix. Les systèmes développés pour répondre à cet enjeu améliorent l'expérience client et augmentent le chiffre d'affaire des e-commerces (30% du chiffre d'affaire en 2011 chez Amazon.com selon Nick Tsinonis au sein de RecSys.com).

De nombreux travaux se sont intéressés aux SR mais certains défis restent à lever encore aujourd'hui comme le démarrage à froid qui désigne un manque d'information lors de l'ajout d'un nouvel utilisateur ou d'un nouvel item au système [Schein et al., 2002], la rareté ou la parcimonie des données explicites comme les notes des utilisateurs qui souvent n'évaluent pas les items [Adomavicius et Tuzhilin, 2005] et le manque, voire l'absence de diversité dans les recommandations des items [Chevalier *et al.*, 2016][Candillier *et al.*, 2011]. Au risque d'être intrusif, un SR se doit aussi d'être le plus pertinent possible pour le client. Le système devrait s'adapter aux situations car souvent les données sur les entités (utilisateurs, produits, etc.) sont dynamiques et évoluent [Louëdec *et al.*, 2015].

Dans la littérature les SR sensibles aux contextes sont utilisés pour traiter ce caractère variable des préférences. Selon [Dey, 2001], un contexte désigne n'importe quelle information qui peut caractériser la situation d'une entité (personne, produit, localisation, etc.). [Palmisano *et al.*, 2008] ont analysé l'influence des informations contextuelles dans la prédiction des comportements et dans la modélisation des utilisateurs (l'étude définit les contextes comme le but ou l' intention d'achats des utilisateurs dans un SR). En fait, les auteurs ont étudié le comportement des utilisateurs qui est susceptible de changer dans différents contextes. En effet, pour un site e-commerce, différents clients peuvent acheter un même produit pour différentes intentions. Par exemple, un champagne peut être considéré comme un produit de luxe adapté pour un cadeau par exemple, mais pour d'autres consommateurs il s'agit d'un produit essentiel pour une fête. Si le champagne est vu par les utilisateurs comme boisson de luxe, ils trouveront pertinents la recommandation d'autres produits de luxe, mais s'il est vu comme produit de fêtes, d'autres accessoires de fêtes seront pertinents. La notion de contexte est étduiée par ailleurs dans de nombreux domaines comme la prise en compte du contexte métier dans l'accès à l'information [Chaker et al., 2013] ou la prise en compte du contexte dynamique dans les profils utilisateurs [Canut et al., 2015].

Les études dans la plupart des SR utilise les notes des utilisateurs pour trouver la similarité entre les items et ne considère pas le contexte dans lequel se trouve l'utilisateur au moment de noter. Dans cet article, nous proposons un SR sensible au contexte utilisant les descriptions des items pour trouver la similarité entre ces items. Pour atteindre cet objectif, nous combinons deux approches :

- la modélisation des thèmes, qui permet de rechercher l'intention des utilisateurs à partir des descriptions textuelles d'un ou plusieurs items successifs qu'ils ont consultés ;
- un système de filtrage collaboratif basé sur le thème de l'utilisateur (ou profil utilisateur) extrait précédemment.

La suite de cet article est structurée comme suit. La section 2 présente l'état de l'art. La section 3introduit la motivation de la méthode que nous proposons, les jeux de données choisis, l'implémentation et l'évaluation de l'approche. La section 4 montre les résultats empiriques et les analyses. La section 5 conclut cet article.

2. Etat de l'art

Les SR peuvent être définis comme des programmes qui visent à recommander les éléments ou items à partir de l'item qui est consulté par l'utilisateur et des informations connexes. Trois types de SR ont été définis dans la littérature [Adomavicius *et al.*, 2005]. Les SR basés sur le contenu génèrent les recommandations à partir de l'historique des préférences de l'utilisateur associé aux caractéristiques (description, prix, couleur, etc.) des items courants [Pazzani et Billsus, 2007]. Le filtrage collaboratif forme un groupe d'utilisateurs qui a les mêmes préférences, ainsi, seuls les items les plus appréciés par le groupe sont pertinents [Adomavicius *et al.*, 2005]. Enfin, l'approche hybride combine les deux approches précédentes [Burke, 2007].

Selon [Adomavicius et Tuzhilin, 2011] malgré un nombre considérable de recherches faites sur les SR, la plupart des approches se focalisent sur la recommandation des items les plus pertinents pour les utilisateurs sans prendre en compte les informations contextuelles (exemple : le temps, la localisation ou la compagnie d'autres personnes). [Adomavicius et Tuzhilin, 2011] ont montré que les informations contextuelles pertinentes ont des influences importantes sur un SR. Il est donc important d'étudier les SR sensibles aux contextes.

La notion de contexte est intéressante pour les SR. Selon [Dey, 2001], un contexte est n'importe quelle information qui peut caractériser la situation d'une entité (personne, localisation, produit, etc.). [Ryan *et al.*, 1999] définissent le contexte comme l'identité de l'utilisateur, ressources de l'environnement proche, localisation de l'utilisateur et période temporelle d'exécution de l'interaction. Selon [Berry et Linoof, 1997], les contextes sont définis comme des évènements qui caractérisent les phases de la vie d'un client et qui peuvent influencer ses préférences, son statut et sa valeur pour une entreprise. Des études comportementales en marketing ont montré que la prise de décision des clients dépend des contextes dans lesquels ils se trouvent [Adomavicius et al., 2005]. En effet, selon les contextes comme la localisation, les saisons, l'humeur, etc. le même client peut choisir différents produits.

Plusieurs recherches ont été menées dans différents domaines pour évaluer l'impact des contextes dans les SR. Ces recherches ont été faites sur les contextes observables (localisation, compagnie, période, etc.) [Borras *et al.*, 2014][Lamsfus *et al.*, 2009] et les contextes non observables (identité d'un membre d'une famille)[Palmisano *et al.*, 2008].

[Adomavicius et al., 2005] ont présenté un SR avec une méthode multidimensionnelle. Des contextes sont ajoutés à la fonction d'évaluation de dimension deux (R : User x Item → Rating). Ainsi on obtient une fonction multidimensionnelle (R : User x Item x Contexte → Rating) qui inclut les informations contextuelles dans la prédiction des préférences des utilisateurs. Pour implémenter la méthode multidimensionnelle et tester sa performance, des données sur des films (notes) et des données contextuelles (localisation, période, compagnie) ont été collectées. Ces données contextuelles ne sont pas disponibles sur la collection de référence Movielens [movielens.umn.edu] généralement utilisé pour évaluer les SR, ni sur les autres données publiques. Par conséquent, un site internet spécifique a été créé et il a été demandé à des utilisateurs d'évaluer les films qu'ils ont vus ainsi que les informations contextuelles pertinentes. Les résultats montrent empiriquement une amélioration de la prédiction des films des systèmes sensibles aux contextes par rapport aux systèmes qui ne les incluent pas.

Selon [Borras et al., 2014] les activités des voyageurs touristiques peuvent être variables en temps réel ; il faut donc adapter les recommandations aux circonstances des voyages (exemple : il pleut ou pas, à l'intérieur ou à l'extérieur d'un musée). Ainsi, dans les applications qui utilisent la mobilité (tourisme, visite de musée, restauration, etc.), les SR sensibles aux contextes améliorent l'expérience utilisateur. L'approche développée par [Lamsfus et al., 2009] utilise les contextes (localisation, période, météo courant) et propose des suggestions à tout instant en fonction des préférences d'activités du touriste. Par exemple, si un client s'attarde sur une activité qu'il rencontre sur la route et que le temps pour les autres activités a du retard, alors les visites suivantes devraient être adaptées au plan initial.

Les informations contextuelles proviennent de plusieurs sources diversifiées. Ainsi, caractériser un contexte de recommandations d'items se différencie par rapport à ses origines. Selon [Adomavicius et Tuzhilin , 2011] il y a trois manières d'obtenir les informations contextuelles :

(1) Explicitement, en posant directement des questions aux utilisateurs (sondages) qui utilisent un site web.

(2) Implicitement, par les informations sur les achats effectués, le nombre de clics, la localisation de l'utilisateur grâce aux smartphones (utilisé en tourisme, restauration).

(3) Par induction, en utilisant des modèles prédictifs (ou des classsifieurs). Par exemple dans un supermarché, il est difficile de connaître explicitement l'identité d'un membre d'une famille, qui réalise des achats ensemble avec une seule carte de paiement ou un même compte pour l'e-commerce. Avec les méthodes d'induction utilisant les classifieurs Naïves Bayes et les

réseaux bayésiens, [Palmisano et al., 2008] ont montré que, des informations contextuelles cachées (ici identité d'un membre) peuvent être induites à partir des données existantes (ici les items achetés).

Un défi se pose sur cette dernière façon d'obtenir les informations contextuelles. Le problème qui se pose est qu'il est difficile de modéliser les contextes à partir des informations contextuelles non observables comme l'intention de l'utilisateur. Une alternative pour le résoudre est la modélisation de thèmes que nous proposons de réaliser via une modélisation LDA.

3. Filtrage collaboratif basé sur LDA

Modéliser les contextes à partir d'informations contextuelles non observables comme l'intention d'achat de l'utilisateur est difficile. Nous avons choisi de capturer l'intention implicite de l'utilisateur par les thèmes associés aux items qu'il a consultés. La méthode LDA (Latent Dirichlet Allocation) permet cette modélisation des thèmes.

Selon [Blei, 2012] les modèles de thèmes sont des techniques d'apprentissage automatique et statistiques qui analysent les mots des textes dans les documents pour découvrir les thèmes qu'ils traitent, comment ces thèmes sont connectés entre eux et comment ils changent au fil du temps.

LDA a été appliqué dans l'analyse de documents [Griffiths et Steyvers, 2004] [Fei-Fei et Perona, 2005], la catégorisation et le regroupement de documents [Wei et Croft ; 2006] [Ramage *et al.*, 2009] et l'ordonnancement de systèmes de recherche d'information [Ionescu *et al.*, 2015]. LDA a été introduit dans les SR afin d'analyser le contexte dans les méthodes basées sur le contenu [Yu *et al.*, 2012]. Dans cet article, nous proposons une autre utilisation du modèle LDA. Le résultat du modèle LDA est intégré dans un système de filtrage collaboratif pour trouver la similarité entre les items consultés par un utilisateur courant.

Nous définissons le profil d'un utilisateur en le représentant par son thème. Notre approche recommande alors des items pour lesquels les distributions de thèmes des titres sont similaires au profil de l'utilisateur courant.

Ainsi, cet article propose un SR sensible aux contextes ; il s'agit d'un modèle de recommandation hybride qui combine la méthode de modélisation de thèmes basée sur LDA et la méthode de filtrage collaboratif. Notre approche est décomposée en trois étapes qui sont décrites dans les sous-sections suivantes.

3.1. Modèle LDA pour la représentation des thèmes des utilisateurs

La première étape implémente le modèle LDA à partir des descriptions des items que les utilisateurs ont consultés. LDA est utilisé pour extraire la structure sémantique cachée dans les descriptions des items que les utilisateurs ont consultés, la distribution des mots sur les thèmes latents et le mélange des distributions des thèmes latents. Cela consiste à estimer la distribution de thèmes latents (noté Θ)

pour chaque item et la distribution de mots (noté φ) pour chaque thème. Ces distributions vont permettre d'identifier la sémantique de l'espace de thèmes latents en les rapportant aux mots et aux items.

Dans la littérature, l'algorithme EM (Expected Maximization) [Blei *et al.*, 2003] et l'algorithme Gibbs sampling [Griffiths et Steyvers, 2004] sont les méthodes les plus utilisées pour l'estimation des paramètres (distributions) Θ et φ du modèle LDA. Cependant, l'algorithme EM est pénalisé par un grand nombre d'opérations à cause du grand nombre de documents ; il est donc plus lent à converger. L'algorithme Gibbs sampling permet de contourner cette difficulté. C'est pour cette raison que nous l'avons utilisé dans ce travail.

Le Collapsed Gibbs Sampling [Griffiths et Steyvers, 2004] est un algorithme d'échantillonnage qui permet l'estimation des paramètres d'un espace discret de grande dimension [Steyvers et al., 2004].

Dans cet article, la méthode de Gibbs sampling est utilisée pour estimer les paramètres de LDA qui itèrent plusieurs fois sur chaque mot v pour extraire un nouveau thème k pour le mot basé sur la probabilité p (z_i=k| v_i, z_{-i}) comme suit :

$$\mathrm{p}(z_i=\mathrm{k}|v_i,z_{-i}) \propto (n_{d,k} + \alpha_k)\frac{n_{k,v}+\beta_v}{\sum_{v\prime} n_{k,v\prime}+\beta_{v\prime}} \qquad (1)$$

Où $n_{k,v}$ calcule le nombre des affectations thème-mot.

$n_{d,k}$ calcule le nombre des affectations document-thème.

z_{-i} désigne toutes les affectations thème-mot et document-thème sauf pour l'affectation courantez_i pour le mot v_i.

α et β sont les paramètres de Dirichlet utilisés comme des paramètres de lissage pour les calculs.

A partir de l'équation (1). Les paramètres θ et φ du modèle LDA sont estimées comme suit [Griffiths et Steyvers, 2004] :

$$\theta_{d,k} = \frac{n_{d,k}+\alpha_k}{\sum_{v\prime} n_{k,v\prime}+\beta_{v\prime}} \qquad (2)$$

$$\phi_{k,v} = \frac{n_{k,v}+\beta_v}{\sum_{v\prime} n_{k,v\prime}+\beta_{v\prime}} \qquad (3)$$

3.2. Similarité entre items

Cette étape intègre les résultats fournis par LDA pour trouver la similarité entre items afin de prédire les préférences de l'utilisateur courant dans le filtrage collaboratif. L'estimation obtenue Θ est la distribution de thèmes latents pour chaque item, vu comme une matrice de similarité d'items par thème, et permet de calculer la similarité entre items. Chaque item possède sa propre distribution à partir de Θ.Pour mesurer la similarité entre deux items, différentes mesures peuvent être utilisées. Nous avons choisi d'utiliser le coefficient de corrélation de Pearson car d'après [Herlocker *et al.*, 2002], en général, les résultats sont meilleurs. Chaque item est représenté comme un vecteur de thèmes et le coefficient de corrélation entre deux items i et j ayant chacune une variance (finie), noté Cor (i, j) est défini par :

$$\mathrm{Cor}\ (i, j) = \frac{Cov(i,j)}{\sigma_i \sigma_j} \qquad (4)$$

Où Cov (i, j) désigne la covariance de deux items i et j, σ_i et σ_j leurs écarts types. Le coefficient de corrélation est symétrique et prend ses valeurs entre -1 et +1

3.3. Prédiction des préférences de l'utilisateur

Notre approche recommande des items pour lesquels les distributions de thèmes des titres sont similaires au profil utilisateur. L'objectif est de prédire les préférences de l'utilisateur courant aux items non consultés. Supposons que nous ayons l'historique des préférences des utilisateurs vus comme une matrice M, qui est la matrice d'évaluation employée dans le filtrage collaboratif. Nous regardons ensuite l'ensemble des items que l'utilisateur courant a déjà consulté et déterminons la similarité avec les autres items que l'utilisateur courant n'a pas encore vus en utilisant la matrice de similarité de l'étape précédente. En effectuant cela, les notes des items pour l'utilisateur courant peut être obtenue et servira à indiquer le degré de préférence de l'utilisateur courant pour les nouveaux items non consultés.

La prédiction des préférences $P_{u,\, i}$ pour un item i, pour l'utilisateur u, est basée sur la moyenne pondérée des préférences et des scores de similarité à partir de tous les autres items qui ont été notés par l'utilisateur u.

La formule est la suivante :

$$\mathrm{Pu,i} = \sum_{j \in J} W_{u,j} * sim(i,j) \qquad (5)$$

Où J est l'ensemble des items les plus similaires à l'item i et que l'utilisateur u a noté ; $W_{u,\, j}$ est le score donné par u pour l'item j ∈ J ; sim (i, j) la similarité entre les items i et j. La somme est calculée à partir de tous les items j ∈ J notés par l'utilisateur u.

Les préférences calculées précédemment sont ordonnées par prédiction de pertinence décroissante et les N premières recommandations non notées par l'utilisateur courant sont recommandées. Notre approche a l'avantage de pouvoir prendre en compte l'oubli en considérant J non pas comme l'ensemble de tous les items déjà consultés mais les k derniers ou l'ensemble des items de la session courante, ou l'ensemble des items consultés dans la semaine courante, l'ensemble des items dans une catégorie donnée, etc. Nous laissons toutefois cette adaptation pour des travaux futurs. Dans la section 4, l'évaluation considère J comme l'ensemble des items déjà consultés par l'utilisateur.

4. Evaluation et résultats

4.1 Cadre d'évaluation : collection, mesures et références

Pour évaluer notre méthode, nous avons utilisé la méthode de validation utilisant 90% des données pour l'entrainement du modèle et 10% de données pour le test. Pour un utilisateur dans les données tests, nous tirons aléatoirement un item supposé être en cours de consultation. A partir de cet item, le système entrainé propose les items recommandés. Si un item recommandé est effectivement noté positivement (la note est supérieure ou égale à 5 dans l'intervalle de 1 à 10) par l'utilisateur dans la collection, la recommandation est considérée comme pertinente. Nous nous appuyons pour l'évaluation sur la collection Book-Crossing et des mesures d'évaluation présentées ci-dessous.

4.1.1 Collection

Source :http://www2.informatik.uni-freiburg.de/~cziegler/BX/

Cette collection a été collectée par Cai-Nicolas Ziegler (via une exploration automatique du web) pendant quatre semaines (en 2004) à partir de la communauté Book-Crossing avec l'autorisation de Ron Hornbaker (Humankind Systems). Elle contient 278 858 utilisateurs (rendus anonymes mais avec des informations démographiques) fournissant 1 149 780 évaluations (explicites / implicites) sur environ 271 379 livres.

La collection **Book-Crossing** est constituée de trois parties :

- **BX-Users**
  contient des informations sur les utilisateurs. Les identifiants des utilisateurs ('ID-Utilisateur') ont été rendus anonymes. Des données démographiques comme ('Localisation','Age') sont parfois fournies.

- **BX-Books**
  Les livres sont identifiés par leur ISBN (International Standard Book Number) ou numéro international standard des livres. Certaines méta-données comme ('Titre du livre', 'Auteur du livre', 'Années de Publication', 'éditeur') sont fournies.

- **BX-Book-Ratings**
  contient les informations de notations du livre. Les notes sont soit explicites, exprimées sur une échelle de 1 à 10 (valeurs plus élevées indiquant une appréciation plus élevée), soit implicites, exprimées par 0.

Dans nos expérimentations, nous nous sommes focalisés sur les données pour lesquelles une notation explicite était présente ; 397 247 notations respectent cette contrainte.

4.1.2 Mesures

En recherche d'information, la précision mesure la proportion de documents pertinents dans l'ensemble de documents restitués. Dans le cas d'un SR, cette mesure peut être adaptée en la proportion d'items pertinents recommandés dans l'ensemble des items recommandés.

$$\textbf{Précision} = \frac{\boldsymbol{RP(i)}}{\boldsymbol{R(i)}} \qquad (6)$$

Où

- RP(i) est le nombre d'items recommandés et pertinents pour l'item i et
- R(i) est le nombre de documents recommandés pour l'item i.

De la même façon, nous pouvons adapter la mesure de précision moyenne pour une requête définie en recherche d'information en la précision moyenne pour un item donné :

$$\text{AP(i)} = \frac{\sum_{r=1}^{R(i)}[P@r(i).rel(r)]}{P(i)} \quad (7)$$

Où :

- P(i) est le nombre d'items recommandés et pertinents pour l'item i ;
- R(i) est le nombre d'items recommandés pour l'item i ;
- r est le rang ;
- P@r(i) est la précision lorsque les r premiers items sont recommandés pour l'item i ;
- rel(r) vaut 1 si la recommandation au rang r est pertinent et 0 sinon.

La moyenne des précisions moyennes (Mean Average Precision ou MAP) est alors la moyenne arithmétique des précisions moyennes sur l'ensemble des items considérées.

$$\text{MAP} = \frac{\sum_{i=1}^{I} AP(I)}{I} \quad (8)$$

Avec I le nombre d'items à partir desquels on recherche les items à recommander.

Ces mesures considèrent deux niveaux de pertinence : un item est soit pertinent, soit non pertinent pour un item de départ donné. Par ailleurs, ces mesures sont orientées vers l'utilisateur qui souhaite d'abord des items pertinents ; le rappel est donc moins important dans les SR.

4.1.3 Méthodes de référence

Nous avons comparé les résultats de notre méthode à 3 modèles de référence.

TFIDF (Term Frequency-Inverse Document Frequency) [Salton, 1989] est une méthode de pondération de termes qui peut être incorporée dans un système de filtrage collaboratif pour trouver la similarité entre items. Il s'agit donc d'une approche hybride.

UBCF (User Based Collaborative Filtering) ou Filtrage Collaborative Basé Utilisateur est une approche qui, à partir d'un utilisateur courant u, recherche les utilisateurs qui sont similaires à cet utilisateur en fonction de la similarité des notes qu'ils ont fournies sur les items. UBCF recommande les items que ces utilisateurs similaires ont aimés [Ekstrand *et al.*, 2011].

IBCF (Item Based Collaborative Filtering) ou Filtrage Collaborative Basé Item est une approche obtenue par la transposition de la matrice de similarité de la méthode UBCF. Alors que UBCF génère des prédictions basées sur les similaritésentre les utilisateurs, IBCF génère des prédictions basées sur les similarités entre les items [Sarwar *et al.*, 2001].

4.2 Résultats et discussions

Le but de l'expérimentation est de comparer notre méthode hybride CBCF avec une autre méthode hybride TFIDF et avec deux autres non hybrides IBCF et UBCF. Nous utilisons la collection Book-Crossing présentée plus haut.

Les figures 1 et 2 montrent une comparaison des performances des différentes méthodes. Nous rapportons la précision et la MAP en fonction du nombre d'items recommandés. Ces résultats correspondent à une moyenne de précision obtenue en prenant 50 items initiaux de différents utilisateurs choisis aléatoirement à partir desquels le système propose des recommandations.

Nous obtenons des résultats MAP ~ 0.5 et précision ~0.6 ce qui correspond à de bons résultats par rapport aux autres méthodes de la littérature.

Dans figure 1, nous obtenons la précision en fonction du nombre d'items recommandés et que nous faisons varier de 5 à 20 items.Pour toutes les méthodes nous obtenons la meilleure performance de précision en utilisant 5 items recommandés et inversement pour 20 items recommandés. Pour 5 items recommandés, notre méthode CBCF enregistre un taux de performance de 58% qui est supérieur à celui de TFIDF de 39%. IBCF et UBCF enregistrent des précisions inférieures de 9% et 6% respectivement.

Dans figure 2, nous comparons la performance de la MAP des différentes approches. Nous observons le même type de résultats que dans la figure 1. CBCF a le meilleur taux avec 47% suivi de TFIDF de 26%. IBCF et UBCF ont des taux de 7% et 5% respectivement.

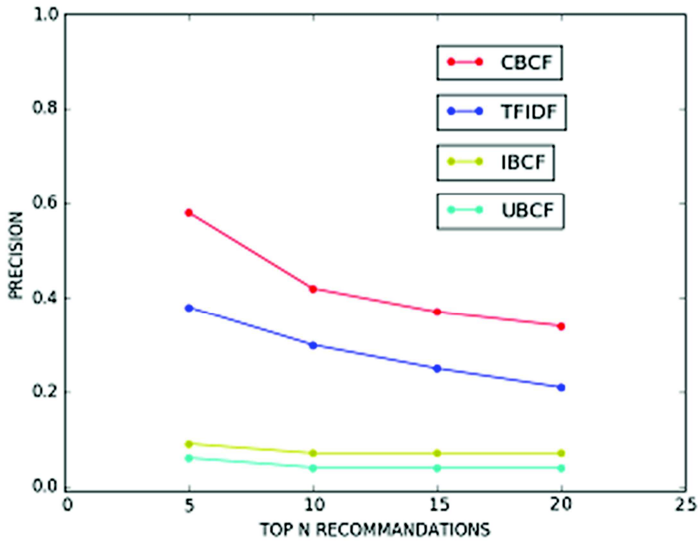

Figure 1. Précision en fonction du nombre d'items recommandés-Moyenne sur 50 items de départ utilisant différentes approches. CBCF correspond à la méthode que nous présentons.

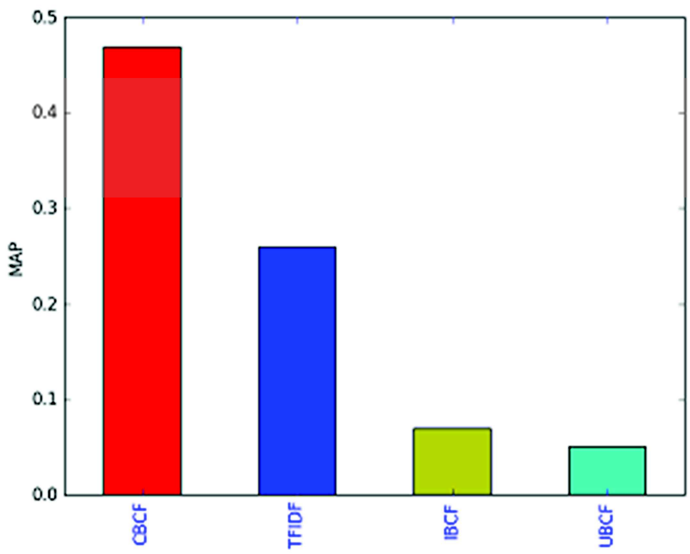

Figure 2. Comparaison de performance de la MAP-Moyennes sur 50 items (Axe des Y) de départ en utilisant les différentes approches, CBCF, TFIDF, IBCF et UBCF (Axe des X).

Dans figures 1 et 2, nous observons que les deux méthodes IBCF et UBCF utilisant des données explicites (notes) sont moins performantes. Ceci est certainement dû au fait que le filtrage collaboratif pur n'arrive pas à gérer le problème de démarrage à froid dans les deux méthodes.

CBCF et TFIDF hybride montrent de meilleure performance comparée à l'approche filtrage collaboratif grâce à leur propriété hybride et l'utilisation des données implicites (titres des livres) pour trouver la similarité entre items. Néanmoins, la méthode que nous proposons a de meilleure performance que TFIDF.

5. Conclusion

Dans cet article, nous avons proposé un SR hybride combinant LDA et la méthode de filtrage collaboratif pour des données implicites. LDA permet de trouver la structure sémantique latente dans les titres des items (ici des livres) consultés, la distribution des mots sur les thèmes latents et le mélange des distributions des thèmes latents. Le résultat provenant de LDA est ensuite intégré dans un système de filtrage collaboratif basé sur la similarité des utilisateurs.

Basée sur le thème de l'intention de l'utilisateur défini comme le profil utilisateur, notre approche recommande des items pour lesquels les distributions de thèmes des titres de l'item courant est similaire au profil utilisateur courant.

Nous avons montré que notre approche donne de meilleurs résultats par rapport au modèle hybride TFIDF. Nous devons maintenant confronter notre modèle à d'autres modèles de la littérature.

Par ailleurs, dans notre implémentation de SR sensible aux contextes, nous avons induit le contexte (intention de l'utilisateur) à partir des titres des livres. Cependant d'autres caractéristiques (auteurs, éditeurs, notes, …) des livres peuvent être utilisées pour induire d'autres contextes non observables et ainsi avoir de meilleure performance. De plus, des travaux additionnels peuvent être effectués en ajoutant des contextes observables comme la localisation, la compagnie, la période, etc.

Bibliographie

Adomavicius, G., Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE transactions on knowledge and data engineering*, *17*(6), 734-749.

Adomavicius, G., Tuzhilin, A. (2011). Contextaware recommender systems. In *Recommender systemshandbook*, pages 217-253. Springer.

Blei, D. M., Ng, A. Y., Jordan, M. I. (2003). *Latent dirichlet allocation. Journal of machine Learning research,* 3(Jan) :993-1022.

Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM, 55(4) :77-84.*

Borras, J., Moreno, A., Valls, A. (2014). Intelligent tourism recommender systems : A survey. Expert Systems with Applications, 41(16) :7370-7389.

Burke, R. (2007). Hybrid web recommender systems. *In The adaptive web, pages 377-408. Springer.*

Candillier L., Chevalier M., Dudognon D., Mothe J. (2012). *Multiple Similarities for Diversity in Recommender Systems.International Journal On Advances in Intelligent Systems*, International Academy, Research and Industry Association, 5(3&4) :234-246.

Candillier L., Chevalier M., Dudognon D., Mothe J. (2011). *Diversity in Recommender Systems: Bridging the gap between users and systems (regular paper).International Conference on Advances in Human-oriented and Personalized Mechanisms, Technologies, and Services (CENTRIC 2011)*, p. 48-58.

Canut M.-F., On-At S., Péninou A., Sèdes F. (2015). *Enrichissement du profil utilisateur à partir de son réseau social dans un contexte dynamique : application d'une méthode de pondération temporelle. INFormatique des Organisations et Systemes d'Information et de Decision (INFORSID 2015)*, p. 15-30.

Chaker H., Chevalier M., Tricot A. (2013). *Une approche de gestion de contextes métiers pour l'accès à l'information (regular paper). INFormatique des Organisations et Systemes d'Information et de Decision (INFORSID 2013)*, p. 115-130.

Chevalier, M., Dudognon, D., Mothe, J. (2016). ADORES : a diversity-oriented online recommender system. *In Proceedings of the 31st Annual ACM Symposium on Applied Computing, pages 1075-1076. ACM.*

Dey, A. K. (2001). Understanding and using context. Personal and ubiquitous computing, 5(1) :4-7.

Ekstrand, M. D., Riedl, J. T., Konstan, J. A., et al. (2011). Collaborative filtering recommender systems. *Foundations and Trends R in Human– Computer Interaction*, 4(2) :81–173.

Fei-Fei, L. & Perona, P. (2005). A bayesian hierarchical model for learning natural scene categories. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), volume 2, pages 524-531. IEEE.

Griffths, T. L. & Steyvers, M. (2004). Finding scientific topics. Proceedings of the National academy of Sciences, 101(suppl 1) :5228-5235.

Herlocker J., Konstan J. A., Riedl J., "An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms," *Information Retrieval*, vol. 5, no. 4, pp. 287–310, 2002.

Ionescu, R. T., Chifu, A. G., Mothe, J. (2015, September). DeShaTo: describing the shape of cumulative topic distributions to rank retrieval systems without relevance judgments. In *International Symposium on String Processing and Information Retrieval* (pp. 75-82). Springer International Publishing.

Lamsfus, C., Alzua-Sorzabal, A., Martin, D., Salvador, Z., and Usandizaga, A. (2009). Human-centric ontology-based context modelling in tourism*. In KEOD, pages 424-434.*

Louëdec, J., Chevalier, M., Mothe, J., Garivier, A., and Gerchinovitz, S. (2015). A multiple-play bandit algorithm applied to recommender systems. *In FLAIRS Conference, pages 67-72.*

Louëdec, J., Chevalier, M., Garivier, A.,Mothe, J. (2015), *Algorithmes de bandits pour la recommandation à tirages multiples.Document numérique*, Hermès, 18(2&3) :59-79.

Palmisano, C., Tuzhilin, A., Gorgoglione, M. (2008). Using context to improve predictive modeling of customers in personalization applications. *IEEE transactions on knowledge and data engineering, 20(11) :1535-1549.*

Pazzani, M. J., Billsus, D. (2007). Content-based recommendation systems. *In The adaptive web, pages 325-341. Springer.*

Ramage, D., Hall, D., Nallapati, R., Manning, C. D. (2009). Labeled lda : A supervised topic model for credit attribution in multi-labeled corpora. *In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing : Volume 1-Volume 1, pages 248-256. Association for Computational Linguistics.*

Rakotonirina A. J. (2017). Filtrage Collaboratif Sensible au Contexte : une approche basée sur LDA, thèse de Master..

Ryan, N., Pascoe, J., Morse, D. (1999). Enhanced reality _eldwork: the context aware archaeological assistant. *Bar International Series, 750 :269-274.*

Salton, G. (1989). Automatic text processing : The transformation, analysis, and retrieval of. Reading : Addison-Wesley.

Sarwar, B., Karypis, G., Konstan, J., Riedl, J. (2001). Itembased collaborative filtering recommendation algorithms. *In Proceedings of the 10th international conference on World Wide Web, pages 285–295. ACM.*

Schein, A. I., Popescul, A., Ungar, L. H., Pennock, D. M. (2002). Methods and metrics for cold-start recommendations. *In Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 253-260). ACM.*

Steyvers, M., Smyth, P., Rosen-Zvi, M., Gri_ths, T. (2004). Probabilistic author-topic models for information discovery. *In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, pages 306-315. ACM.*

Wei, X., Croft, W. B. (2006). Lda-based document models for ad-hoc retrieval. In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, pages 178-185. ACM..

Yu, K., Zhang, B., Zhu, H., Cao, H., Tian, J. (2012). Towards personalizedcontext-aware recommendation by mining context logs through topic models. *In Pacific-Asia Conference on Knowledge Discovery and Data Mining, pages 431-443. Springer.*