

Renforcement des modèles probabilistes en utilisant l'Information Mutuelle pour des Recommandations contextualisées

Geoffray BONNIN

Armelle BRUN

Anne BOYER

geoffray.bonnin@loria.fr

armelle.brun@loria.fr

anne.boyer@loria.fr

LORIA – Université Nancy2

RÉSUMÉ. Les systèmes de recommandation ont pour but d'aider l'utilisateur dans ses recherches. La plupart des systèmes de recommandation exploitent les similitudes entre les utilisateurs, en particulier le filtrage collaboratif, qui consiste à considérer les votes des utilisateurs pour fournir les recommandations, sans regarder le contenu des ressources considérées. Or, les votes constituent le seul critère de recommandation, et il semblerait intéressant d'étendre le concept en y incluant la notion de contexte temporel.

Dans cet article, nous présentons un modèle qui fournit des recommandations personnalisées, basé à la fois sur la corrélation et le contexte temporel, en prenant en compte un historique de taille fixe. Nous proposons d'utiliser deux modèles ayant des propriétés complémentaires et de les combiner afin de fournir des listes de recommandation plus précises. Le premier est basé sur les modèles de Markov et le second est issu du domaine des modèles statistiques de langage, où il est connu sous le nom de modèle trigger.

Nous avons testé notre modèle contextuel sur un corpus de navigation sur l'Intranet bancaire du Crédit Agricole S.A. Nous montrons la complémentarité des deux modèles qui se concrétise par une nette amélioration des résultats

MOTS-CLÉS : Systèmes de recommandation, modèles de Markov, triggers, information mutuelle.

ABSTRACT. Recommender Systems aim at providing help for users to find pertinent resources. Most of recommender systems take advantage of the similarities between users, in particular collaborative filtering which consists in providing recommendations based on the users' ratings, without taking into account the content of the resources. However, ratings are the only criterion taken into account, and it seems interesting to extend the concept by including the notion of temporal context.

In this paper, we present a model able to provide personalised recommendations, based both on the correlation and the temporal context by taking into account a fixed size history. We propose to use two models that have complementary features and to combine them in order to provide more accurate recommendation lists. The first one is based on Markov models and the second is extracted from the natural language domain and is known as trigger model.

Our context sensitive model has been tested on an Intranet browsing dataset of the Crédit Agricole S.A. Banc. We show the complementarity of both models which is materialised by a significant improvement.

KEYWORDS: Recommender systems, Markov models, triggers, mutual information.

1. Introduction

De nos jours, le Web fournit un espace de ressources presque illimité. C'est pourquoi les utilisateurs doivent recourir à des moteurs de recherche et des systèmes de recommandation afin de trouver des ressources correspondant à leurs besoins. De nombreuses approches ont été utilisées, parmi lesquelles le filtrage collaboratif, les recommandations basées sur le contenu, les raisonnements à partir de cas, ou encore des recommandations basées sur la connaissance de l'utilisateur ou du domaine.

L'une des approches principales est le Filtrage Collaboratif (FC) [Goldberg 1992, Das 2007], qui

considère les appréciations des utilisateurs pour fournir les recommandations. Le principe est d'estimer l'appréciation d'un utilisateur actif, en fonction de l'avis d'autres utilisateurs ayant des goûts similaires. Le point intéressant est que l'algorithme n'a pas besoin de considérer le contenu des ressources ; cependant, il prend en compte toutes les évaluations précédentes de l'utilisateur sous forme d'un ensemble de votes. Par conséquent il ne fournit pas de recommandations contextualisées.

Les intérêts des utilisateurs peuvent évoluer en fonction de la période de la journée, l'humeur ou les actions passées. Le contexte a donc une importance majeure lors de la recommandation de ressources à un utilisateur. Parmi les différents contextes possibles, nous pouvons considérer la prise en compte des actions récentes, qui a montré son efficacité dans le cadre de la navigation Web [Deshpande 2004] ; cependant cela implique généralement un modèle complexe, et soulève la question de la couverture. Le défi est donc de trouver un bon compromis entre la complexité, la précision et la couverture.

Dans ce papier, nous présentons un modèle de recommandations contextualisées de faible complexité et ayant une bonne précision, en tenant compte du contexte. Celui-ci combine deux sous-modèles complémentaires : le premier est un modèle de Markov et le second est basé sur une mesure normalisée d'information mutuelle (pour le calcul des corrélations entre les ressources) qui possède des caractéristiques complémentaires aux probabilités utilisées dans les modèles de Markov. Ce second modèle a aussi l'avantage de prendre en compte des ressources plus éloignées que les modèles de Markov. Ainsi, la combinaison de ces deux modèles augmente la précision des recommandations.

L'article s'articule autour des parties suivantes. Une première section introduit la problématique. Puis nous présentons les travaux connexes. La section 3 définit le modèle contextuel que nous proposons. Dans la section 4, nous présentons notre protocole d'évaluation et les résultats obtenus. La conclusion et les perspectives sont enfin mises en avant dans la dernière section.

2. Travaux connexes

Deux approches principales de la recommandation prenant en compte un passé récent sont utilisées dans la littérature. La première considère le contexte comme un ensemble de ressources non ordonnées, comme le font les règles d'association [Agrawal 1993, Fu 2000] ou quelques modèles de filtrage collaboratif [Zimdars 2001], alors que la seconde approche représente le contexte comme un ensemble ordonné de ressources comme le font les motifs séquentiels [Agrawal 1995, Lu 2005] et les modèles de Markov [Zimdars 2001, Mobasher 2007].

Ces modèles sont des modèles statistiques et sont construits à partir de corpus d'apprentissage. Ces corpus sont constitués des sessions de navigation des différents utilisateurs. Le système exploite les traces des utilisateurs de la communauté pour construire le modèle et recherche les ensembles ou séquences de ressources correspondants aux traces de l'utilisateur courant pour effectuer les recommandations, ce qui constitue une forme de collaboration passive.

Considérant que la navigation Web est structurée en liens hypertextes, la prise en compte d'un contexte séquentiel semble appropriée pour fournir des recommandations. Dans le reste de cet article, nous nous concentrerons donc sur la prise en compte de ce contexte.

2.1 Modèles de Markov

Un des modèles les plus populaires est le modèle de Markov. Un modèle de Markov d'ordre k

réduit le nombre de ressources à prendre en compte dans l'historique aux k dernières. Un modèle de Markov d'ordre 1 ne considère donc que la ressource précédente, ce qui ne modélise pas très précisément le comportement d'un utilisateur navigant sur le Web. En effet, non seulement ces modèles ne considèrent pas suffisamment de ressources, mais en plus ils ne sont pas suffisamment éloignés dans l'historique pour discriminer différents historiques [Deshpande 2004].

Cependant, en pratique, seuls des modèles d'ordre 1 ou 2 sont utilisés, plus rarement 3. Au-delà de ces valeurs, le nombre de possibilités devient très grand et il est très difficile de trouver des données d'apprentissage capables de fournir une modélisation représentative, et même si de tels corpus sont disponibles, le modèle obtenu est trop lourd à manipuler. Pour les mêmes raisons, l'utilisation de modèles de Markov d'ordres supérieurs implique une réduction de la couverture. En effet, le modèle issu du corpus d'apprentissage ne permet pas toujours de trouver des historiques correspondant à l'historique de l'utilisateur actif.

Un moyen de fournir une couverture quasi totale tout en conservant la précision des modèles de Markov d'ordres supérieurs est d'utiliser des modèles de Markov de différents ordres variant de k à 1. Quand aucun historique de taille k ne correspond à l'historique de l'utilisateur actif, un historique de taille $k - 1$ est utilisé. Le modèle obtenu est appelé le all- k^{th} -order Markov model [Pitkow 1999].

Un modèle de Markov a également l'inconvénient de ne prendre en compte que des séquences strictement contiguës. Si un utilisateur effectue des navigations parallèles ou accède à une ressource indésirable (bruit), le modèle ne sera pas en mesure de considérer le vrai chemin, et réduira l'ordre du modèle. En outre, lorsque le modèle ne correspond pas à l'historique, les ressources les plus éloignées sont ignorées, et les ressources les plus proches sont toujours considérées alors que certaines d'entre elles pourraient être des erreurs de navigation et devraient être ignorées.

Enfin, les ressources sont recommandées en fonction de leurs probabilités conditionnelles étant donné l'historique. Dans une telle configuration, les ressources les plus fréquentes sont plus susceptibles d'être recommandées, même si elles peuvent ne pas être d'une grande utilité pour un utilisateur. Par exemple, la page d'accueil d'un site Web est généralement la plus visitée, mais pourrait ne pas être la plus intéressante à recommander à un utilisateur.

2.2 Modèle de triggers multi-mots

Les modèles de triggers [Rosenfeld 1992] sont utilisés dans la modélisation statistique du langage naturel pour améliorer les modèles n -grammes (modèle de Markov d'ordre $n - 1$). Ils sont composés de paires ordonnées de mots fortement corrélés, le premier mot A étant le déclencheur et le second mot C le mot déclenché. Comme les mots sont extraits à l'intérieur d'une fenêtre glissante de taille fixe, des ressources distantes sont considérées. Le modèle est par conséquent robuste au bruit.

Une amélioration des modèles de trigger appelée modèle de triggers multi-mots a été expérimentée dans [Chen 2003]. Ce modèle consiste à considérer plus d'un mot comme élément déclencheur.

La corrélation entre les mots est habituellement mesurée en utilisant l'Information Mutuelle (IM), qui mesure la quantité d'information apportée en moyenne par les déclencheurs sur la probabilité du mot déclenché. Si le nombre de déclencheurs est égal à 2, si A et B sont ces déclencheurs et si C est l'élément déclenché, l'information mutuelle est donnée par la formule suivante :

$$\begin{aligned}
\text{MI}(A, B, C) = & p(A, B, C) \log \frac{p(C|A, B)}{p(C)} + p(\overline{A}, \overline{B}, C) \log \frac{p(C|\overline{A}, \overline{B})}{p(C)} \\
& + p(A, B, \overline{C}) \log \frac{p(\overline{C}|A, B)}{p(\overline{C})} + p(\overline{A}, \overline{B}, \overline{C}) \log \frac{p(\overline{C}|\overline{A}, \overline{B})}{p(\overline{C})}
\end{aligned} \quad [1]$$

où A désigne la présence de A et \overline{A} l'absence de A , $p(A, B, C)$ la probabilité que A , B , et C soient dans une même fenêtre de taille D et $p(C|A, B)$ la probabilité conditionnelle de C sachant que A et B sont à l'intérieur de la fenêtre. Cette formule peut être adaptée à un plus grand nombre de déclencheurs. Utiliser cette mesure permet une modélisation différente de celle utilisant des probabilités conditionnelles, car elle mesure les corrélations entre les deux parties.

L'IM constitue une mesure complémentaire et est habituellement combinée avec des modèles n -grammes. De plus, les modèles de triggers sont capables de manipuler des relations entre des ressources éloignées, ce qu'un modèle de Markov ne peut pas faire. Dans la suite, le modèle de triggers multi-mots est appliqué à la navigation Web, et est appelé **modèle de triggers multi-pages**.

3. Un système de recommandation contextualisé

Le modèle contextualisé que nous proposons ici consiste en une combinaison des modèles de Markov avec un modèle de triggers multi-pages. Comme un modèle de Markov simple ne fournit pas à la fois une couverture totale et une bonne précision nous utilisons un *all- k^{th} -order Markov model*.

Dans cette section, nous décrivons d'abord le modèle de triggers multi-pages avec un nombre de déclencheurs de 2 pour des raisons de simplicité, mais il peut facilement être transposé à n'importe quel nombre de déclencheurs. Nous décrivons ensuite comment nous combinons nos deux modèles.

3.1 Le modèle de triggers multi-pages

3.1.1 Information Mutuelle normalisée

Les valeurs de la mesure d'information mutuelle de la formule (1) sont très hétérogènes. Afin de permettre une combinaison avec le modèle probabiliste, nous avons appliqué une forme normalisée extraite de [Fred 2003]. L'information mutuelle normalisée (NMI) est donnée par la formule suivante :

$$\text{NMI}(A, B, C) = \frac{\text{MI}(A, B, C)}{H(A) + H(B) + H(C)} \quad [2]$$

où $H(A)$, $H(B)$ et $H(C)$ sont les entropies de A , B et C , et on a $0 \leq \text{NMI}(A, B, C) \leq 1$.

3.1.2 Pondération en fonction de la distance

Les pages considérées pendant la phase d'apprentissage et l'application du modèle de triggers multi-pages sont prises à l'intérieur d'une fenêtre glissante de taille fixe D . La distance d_1 entre le premier déclencheur et la page déclenchée et la distance d_2 entre le second déclencheur et la page déclenchée peuvent avoir n'importe quelles valeurs dans $\{0, \dots, D-2\}$ telles que $d_1 \leq D-2$ et $d_2 < d_1$.

Nous partons de l'hypothèse que les pages les plus distantes devraient avoir un impact moindre sur

les prédictions, et proposons donc de pondérer les valeurs des NMI en fonction de d_1 et d_2 . Cette pondération est calculée selon un schéma de décroissance exponentielle extraite de [Bonnin 2008].

3.1.3 Apprentissage

Les probabilités utilisées pour le calcul des valeurs d'information mutuelle normalisée sont calculées en fonction des occurrences de chaque triplet possible à l'intérieur de la fenêtre glissante sur un corpus d'apprentissage. On calcule pour chaque triplet considéré un nombre d'occurrences pondéré. Avant de calculer les valeurs finales des probabilités, les triplets sont filtrés en fonction de leur nombre d'occurrences afin de supprimer les triplets les moins représentatifs.

3.1.4 Calcul du score pour le modèle de triggers multi-pages

Étant donné un historique h de taille $D-1$, un score est attribué à chaque ressource r candidate pour la recommandation. Ce score est une combinaison des NMI calculés en fonction de tous les déclencheurs dans la limite de l'historique. Il sera utilisé pour calculer les listes de recommandation et est appelé Combined Normalized Mutual Information (CNMI).

3.2 Combinaison des modèles

La combinaison *du all- k^{th} -order Markov model* et du modèle de triggers multi-pages est effectuée selon une interpolation linéaire. Le score final utilisé pour trier les pages dans les listes de recommandation est calculé en fonction de la formule suivante :

$$\text{score}(r, h) = \alpha p(r | h) + (1 - \alpha) \text{CNMI}(r, h) \quad [3]$$

où $0 \leq \alpha \leq 1$, $p(r | h)$ est la probabilité conditionnelle de r sachant h fournie par le all- k^{th} -order Markov model. Plusieurs valeur de α , $0 \leq \alpha \leq 1$, sont testées dans les expérimentations.

4. Expérimentations et Résultats

4.1 Protocole d'évaluation

4.1.1 Corpus

Les données que nous avons utilisées ont été fournies par le Crédit Agricole S.A. Les employés de cette banque utilisent une interface contenant des environnements de travail, des news, des articles, etc. La banque nous a fourni des logs de navigation anonymisées contenant 3391 ressources (pages) distinctes sur lesquelles 815 employés de banque ont navigué. A partir de ces logs, nous avons pu extraire un corpus de 123470 consultations. Afin de le rendre exploitables, nous l'avons divisé en un corpus d'apprentissage qui contient 90% des sessions et un corpus de test qui contient les 10% des sessions restantes.

4.1.2 Mesure d'évaluation

Pour évaluer notre modèle, nous avons utilisé le Recommendation Score (RS). Cette mesure évalue la pertinence moyenne des listes de recommandation. Pour chaque historique du corpus de test, une liste de recommandations de taille m est calculée, contenant les m ressources les plus probables selon

le modèle. Si la ressource observée est dans la liste, la recommandation est pertinente. Cette mesure calcule donc le pourcentage de recommandations pertinentes.

4.2 Résultats

Cette section est consacrée à l'étude des performances de notre modèle contextualisé. Les tests sont effectués en utilisant deux déclencheurs et des listes de recommandation de taille 10.

Nous nous intéressons d'abord aux performances du *all-kth-order Markov model* quand il est utilisé seul. Le Tableau I montre les RS obtenus en fonction d'une valeur de k croissante. Comme on peut le voir, au-delà d'une valeur de $k = 2$, le RS diminue. Cela est dû à la taille du corpus qui n'est pas d'ordre assez grand pour obtenir un modèle de Markov d'ordre plus grand qui soit fiable. Nous avons donc choisi une valeur de $k = 2$ pour la combinaison avec le modèle de triggers multi-pages.

Tableau I : Résultats du *All-kth-order Markov model* utilisé seul

k	0	1	2	3	4
RS	31,88	64,83	65,16	61,34	60,51

La Figure I montre les RS obtenus quand l'interpolation linéaire est utilisée, en fonction d'une valeur de α croissante. Le RS pour $\alpha = 0$ correspond au modèle de triggers multi-pages utilisé seul et atteint 63,47 et le RS pour $\alpha = 1$ au résultat présenté dans le Tableau I. La meilleure valeur est obtenue pour $\alpha = 0,6$ et atteint un RS de 68,78 ce qui représente une amélioration de 8,37% en comparaison du *all-kth-order Markov model* utilisé seul. Ces résultats mettent clairement en avant la complémentarité des deux modèles, due à l'utilisation de probabilités et de l'information mutuelle et de l'intégration de ressources distantes. Le modèle qui en résulte constitue un modèle encore plus précis que le *all-kth-order Markov model* tout en conservant une couverture totale.

Figure I : Interpolation linéaire du all-kth-order Markov model avec le modèle de triggers multi-pages

5. Conclusion

Dans cet article nous mettons en avant la prise en compte du contexte temporel pour la recommandation d'information. Nous décrivons tout d'abord les travaux connexes et présentons deux modèles, le premier étant un modèle performant de l'état de l'art et le second étant basé sur l'information mutuelle et ayant des propriétés complémentaires. Nous tirons profit des deux modèles en les combinant à l'aide d'une interpolation linéaire. Le modèle obtenu a été testé sur un corpus de navigation sur un Intranet, concrétisant la complémentarité entre les deux modèles par une nette amélioration de 8,37%.

Dans un travail futur, nous étudierons le moyen d'intégrer des modèles contextualisés ne prenant pas en compte l'ordre. Un moyen d'accomplir cela serait de regrouper les éléments du *all-kth-order Markov model* ayant une fréquence similaire.

6. Références

- [Agrawal 1993] R. Agrawal, T. Imielinski, et A. Swami, Mining Association Rules Between sets of Items in Large Databases, *ACM SIGMOD international conference on Management of data*, 1993, p. 207-216,.
- [Agrawal 1995] R. Agrawal et R. Srikant, Mining Sequential Patterns, *International Conference on Data Engineering*, 1995, p 3-14
- [Bonnin 2008] G. Bonnin, A. Brun et A. Boyer, Using Skipping for Sequence-Based Collaborative Filtering, *International Conference on Web Intelligence (WI)*, 2008, p. 775-779
- [Chen 2003] Y. Chen et K. Chan, Extended Multi-word Trigger Pair Language Model using Data Mining Technique, *IEEE International Conference on Systems, Man and Cybernetics*, 2003, p. 262-267
- [Das 2007] A. Das, M. Datar, A. Garg et S. Rajaram, Google News Personalization: Scalable Online Collaborative Filtering, *International Conference on World Wide Web*, 2007, p. 271-280
- [Deshpande 2004] M. Deshpande et G. Karypis, Selective Markov Models for Predicting Web Page Accesses, *ACM Trans. Internet Technol.*, 2004, p. 163-184,
- [Fred 2003] A. Fred et A. Jain, Robust Data Clustering, *IEEE Computer Society Conference on Vision and Pattern Recognition*, 2003.
- [Fu 2000] X. Fu, J. Budzik et K. Hammond, Mining Navigation History for Recommendation, *International Conference on Intelligent User Interfaces*, 2000, p. 106-112
- [Goldberg 1992] D. Goldberg, D. Nichols, B. Oki et D. Terry, Using Collaborative Filtering to Weave an Information Tapestry, *Communications of the ACM*, 1992, p. 61-70
- [Lu 2005] L. Lu, M. Dunham, et Y. Meng, Mining Significant Usage Patterns from Clickstream Data, *International Workshop on Knowledge Discovery on the Web*, 2005, p. 1-17
- [Mobasher 2007] B. Mobasher, Data Mining for Web Personalization, *chapter 3*, 2007
- [Pitkow 1999] J. Pitkow et P. Pirolli, Mining Longest Repeating Subsequences to Predict World Wide Web Surfing, *2nd conference on USENIX Symposium on Internet Technologies and Systems*, 1999, p 139-150
- [Rosenfeld 1992] R. Rosenfeld et X. Huang, Improvement in Stochastic Language Modeling, *DARPA Workshop on Speech and Natural Language*, 1992, p. 107-111
- [Zimdars 2001] A. Zimdars, D. Chickering et C. Meek, Using Temporal Data for Making Recommendations, *17th Conference in Uncertainty in Artificial Intelligence*, 2001.