

# Apprentissage relationnel pour la recommandation et la prédiction de données manquantes

Damien Sileo<sup>1,2</sup> et Vincent Guigue<sup>1,2</sup>

<sup>1</sup>Université Pierre et Marie Curie, PRES Sorbonne-Universités

<sup>2</sup>Laboratoire d'Informatique de Paris 6, UMR 7606, CNRS

## Résumé

La recommandation joue le rôle primordial de mettre en relation des utilisateurs avec ce qui leur correspond le mieux. Les applications s'étendent du ciblage publicitaire aux catalogues de films et plus généralement, c'est l'ensemble des outils d'accès à l'information en ligne qui se personnalisent. Les historiques de notes ou de navigation sont les principales sources d'information pour produire les recommandations en établissant des profils qui caractérisent des utilisateurs et les contenus. Un des enjeux majeurs du domaine est d'exploiter toutes les informations sur le contenu et/ou l'utilisateur pour améliorer la pertinence des recommandations. Dans cette étude, nous proposons un cadre unifié axé à la fois vers la prédiction de notes et celle des attributs contextuels des items et des utilisateurs (genre des films, sexe des utilisateurs...). Nous utilisons le formalisme de l'apprentissage de représentations pour projeter l'ensemble des concepts dans un espace latent puis nous tirons parti des relations explicites pour prédire les données manquantes ou implicites. Nous évaluons notre approche sur le jeu de données MovieLens et obtenons des résultats qui dépassent l'état de l'art.

**Mots-clé** : Recommandation hybride, apprentissage relationnel

## 1 Introduction

Face au nombre croissant de flux de données, les moteurs de recommandations permettent d'appareiller automatiquement des utilisateurs avec les items qui leur correspondent le mieux. Leur utilisation est vaste et en constante expansion, bien au delà de l'e-commerce ou de la proposition de contenus (Amazon, Netflix...), elle concerne tout l'accès personnalisé à l'information. Nous distinguons trois grandes familles de systèmes de

recommandation [AT05]; les approches basées sur le contenu, qui exploitent les informations connues sur les utilisateurs ou les items et émettent des propositions fondées sur la proximité dudit contenu. Le filtrage collaboratif, à l'inverse, se repose sur des historiques d'interactions entre les utilisateurs et les items (notes, clics...). Depuis quelques années, les modèles basés sur des projections latentes, comme la factorisation matricielle (FM), ont démontré leur intérêt pour construire des propositions personnalisées sur de grandes bases de données [KBV09]. Enfin, les approches hybrides permettent de combiner les informations de contenu et les traces utilisateurs (notamment les notations). Une partie de ces modèles repose également sur la FM [YWG14].

L'apprentissage de représentation consiste à attribuer à chaque profil d'utilisateur ou d'item, un point dans un espace continu; la phase d'inférence exploite ensuite une métrique entre ces profils pour en déduire, par exemple, la note que donnerait un utilisateur à un produit. L'enjeu consiste alors à améliorer ces profils pour rendre les prédictions plus pertinentes. Les travaux récents se focalisent sur la modélisation du contexte [HLGZ14], la prise en compte des phénomènes temporels [Kor09] ou du contexte social de l'utilisateur [MZL<sup>+</sup>11] et démontrent des améliorations concernant la prédiction de note. Notre contribution consiste à basculer dans un formalisme prédictif multi-tâches : nous cherchons à construire un espace latent associé à des métriques capables de prédire n'importe quel élément de contexte d'un utilisateur (classe d'âge, sexe, ...) ou d'un item (genre de film, ...) à partir des autres éléments disponibles. A terme, l'idée est de faire converger les systèmes de recommandation et les systèmes de raisonnements dans les espaces latents tels que [BW11] : l'objectif est donc de prédire des relations diverses entre des entités. Ce problème est aussi connu comme de la construction de base de connaissances. Les Neural

Tensor Networks [SCMN13] fournissent un cadre général pour cet apprentissage relationnel ; les triplets RDF  $(e_i, r_j, e_k)$  [W3C14] correspondent chacun à un niveau de confiance. Les entités  $e$  et relations  $r$  sont projetées dans un espace latent et un produit tensoriel estime ce niveau de confiance. Nous allons adapter ce cadre à la recommandation hybride.

En plus d’améliorer la précision des recommandations, le partage d’un même espace latent permet d’expliquer des recommandations, d’étudier et de visualiser les représentations de concepts hétérogènes dans cet espace. Par ailleurs, le paradigme de prédiction de relations permet d’apprendre la complétion de données manquantes. Nous évaluerons ainsi notre modèle sur différentes tâches comme la prédiction du sexe des utilisateurs. La prise en compte du contexte est largement traitée dans la littérature [BB02] mais l’architecture que nous proposons subsume plusieurs modèles et permet de les combiner élégamment et, *in fine*, de dépasser l’état de l’art en prédiction de notes.

Cet article s’articule d’abord autour de la description des modèles utilisés (section 2), nous démontrons ensuite leur efficacité dans la partie expérimentale (section 3) puis nous revenons sur la littérature du domaine en section 4.

## 2 Modèles d’apprentissage de représentation

Pour valider nos hypothèses, nous avons choisi de travailler sur le jeu de données MovieLens 1M [HK15] constitué d’un million de notes de 1 à 5 que des utilisateurs ont donné à des films. Ces notes sont enrichies par la tranche d’âge, le sexe et l’occupation (métier ou autre) des 6040 utilisateurs, ainsi que du genre des 3952 films. Dans notre formalisme, les concepts *utilisateur*, *film*, *âge*, *genre*, *sexe*, *occupation*, correspondant aux entités, seront projetés dans l’espace latent et nommés respectivement  $\mathcal{U}, \mathcal{V}, \mathcal{A}, \mathcal{G}, \mathcal{S}, \mathcal{O}$ . Nous noterons l’ensemble des concepts  $\mathcal{I}$ . La figure 1 permet de visualiser les liens explicites ou implicites entre les concepts. Ensuite, nous proposons de caractériser toutes les relations sous la forme  $R_{i,j}$  où  $i$  et  $j$  sont des entités. Dans le cas le plus classique,  $i = \text{utilisateur}$ ,  $j = \text{film}$ , et  $R_{i,j}$  représente la note que  $i$  donne à  $j$ . Mais dans notre cadre,  $j$  peut aussi être une classe d’âge ;  $R_{i,j}$  vaut alors 1 si l’utilisateur appartient à cette classe et 0 sinon.

Chaque entité  $e$  correspond à un vecteur  $\mathbf{x}_e \in \mathbb{R}^K$  où  $K$  est un hyper-paramètre. Le problème d’apprentissage de représentation consiste à optimiser les vecteurs

$\{\mathbf{x}_e | \forall e \in \mathcal{I}\}$  de telle manière que  $\mathbf{x}_i \mathbf{x}_j$  soit une bonne approximation de  $R_{i,j}$ .

### 2.1 Factorisation matricielle pour la prédiction de notes

Depuis le challenge Netflix [BL07], la factorisation matricielle (régularisée), abrégée en FM par la suite, fait figure de référence en matière de prédiction personnalisée de note ; elle attribue à chaque entité des facteurs latents qui capturent ses propriétés et l’opérateur de prédiction est un simple produit scalaire entre représentations latentes. Usuellement, la formulation du coût au sens des moindres carrés régularisé pour la prédiction des notes sur l’ensemble de la base de données revient à :

$$L^{FM} = \sum_{i,j \in \mathcal{U}, \mathcal{V}} (R_{i,j} - \mathbf{x}_i \mathbf{x}_j)^2 + \lambda \|\mathbf{X}\|_F^2 \quad (1)$$

où  $\lambda$  est le compromis de régularisation limitant le sur-apprentissage en agissant sur la norme de Frobenius de la matrice  $X$  regroupant l’ensemble des représentations latentes. La minimisation de cette fonction incite à donner des facteurs latents communs élevés pour les utilisateurs et les films reliés par une note élevée. In fine, ces représentations se rapprochent dans l’espace.

### 2.2 Factorisation matricielle contextualisée

Utiliser le contexte des utilisateurs et des items pour améliorer la qualité des représentations revient à déplacer les facteurs latent en introduisant une hypothèse de régularité du type : tous les utilisateurs de la même tranche d’âge partagent certains points communs. Les approches basées sur l’alignement [NZ13, KL14] introduisent une fonction de similarité  $s$  entre deux entités du même genre et bonifient le respect des régularités :

$$L^{FMC} = L^{FM} - \sum_{\substack{i,j \in \mathcal{V}^2 \\ i,j \in \mathcal{U}^2}} s(i,j) \mathbf{x}_i \mathbf{x}_j \quad (2)$$

lorsque les utilisateurs (ou item)  $i$  et  $j$  partagent un concept, la fonction de coût baisse si les représentations latentes partagent des coefficients. Il est possible d’ajouter les mêmes contraintes sur les items.

Cette approche rapproche effectivement les facteurs latents d’utilisateurs similaires, mais sa complexité croît quadratiquement avec le nombre d’utilisateurs/items. De plus, elle dépend fortement de la similarité  $s$  choisie, et l’interprétation des facteurs latents n’est pas directement facilitée.

Un autre paradigme nous est offert dans le cadre de la prédiction de relations [LHYT08]. Les auteurs présentent une variante de factorisation matricielle (Multi-Relational Matrix Factorization - MRMF) pour la prédiction de lien et de notes. Les valeurs de  $R$  sont donc toujours binaires sauf pour les notes et les entités.

$$L^{MRMF} = \sum_{\substack{i,j \\ R_{i,j} \neq 0}} (R_{i,j} - \mathbf{x}_i \mathbf{x}_j)^2 + \lambda \|X\|_F^2 \quad (3)$$

Pour les relations binaires, l'inférence est très simple : tous les produits scalaires entre représentations d'entités supérieurs à un seuil correspondent à l'existence d'un lien.

Dans notre cadre unifié, cette équation est équivalente à la factorisation matricielle de base. Cependant, nous vérifierons dans la partie expérimentale que la prédiction de notes n'est pas optimale dans ce cadre : chaque représentation d'utilisateur et d'item se retrouve alors plus contrainte par son contexte que par la prédiction de notes et les performances ne sont pas intéressantes sur cette dernière tâche.

### 2.3 Factorisation matricielle en contexte pondéré

Nous proposons l'approche Weighted Contextual Matrix Factorization (WCMF) dans laquelle nous différencions les types de liens dans  $\mathcal{T}^2$ . Chaque type de relation observé  $r$  est associé à un coefficient  $\alpha_r$ . Pour plus de clarté, nous notons la relation typée entre  $i$  et  $j$  :  $R_{i,j}^r$  et nous obtenons donc la formulation suivante :

$$L^{WCMF} = \sum_r \alpha_r \sum_{\substack{i,j \\ R_{i,j}^r \neq 0}} (R_{i,j}^r - \mathbf{x}_i \mathbf{x}_j)^2 + \lambda \|X\|_F^2 \quad (4)$$

Dans la pratique le  $\alpha_r$  correspondant à la prédiction de note est unitaire et les coefficients  $\alpha_r$  de contexte ont tous la même valeur qui est déterminée par *grid search* en optimisant la performance sur un ensemble de validation. On nomme  $\alpha_c$  la valeur de  $\alpha_r$  commune aux relations de contexte. Les tests préliminaires portant sur une distinction plus fine des  $\alpha_r$  en fonction notamment du cardinal des classes de contexte n'ont pas été concluants sur ce jeu de données.

La pondération permet d'équilibrer les déplacements des facteurs latents utilisateur et item en fonction de la prédiction de notes et du contexte en modulant les paramètres  $\alpha$ . L'optimisation est réalisée par descente de gradient stochastique, triplet par triplet. Par ailleurs, nous utilisons aussi des biais attachés aux utilisateurs,

items et attributs comme dans [Pat07], même si ceux-ci ne sont pas dans les équations par souci de clarté.

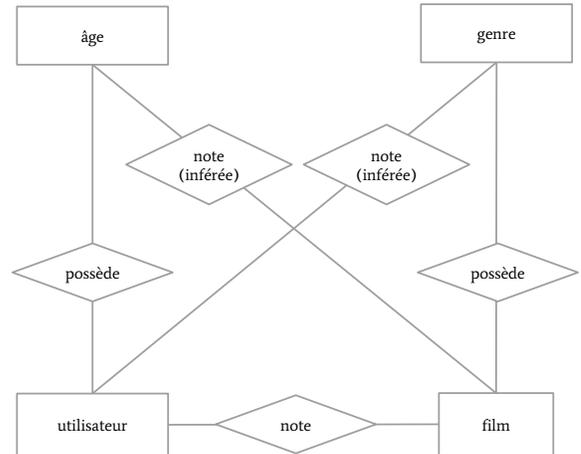


FIGURE 1 – diagramme entité-association des données Moviens 1M. Certains liens sont explicites, d'autres sont implicites (e.g. la note donnée par une classe d'âge à un film).

### 2.4 Utilisation de relations implicites

Lorsqu'un utilisateur note un film, considérer ce que film possède un genre n'est pas suffisant pour modéliser toute l'information liée au genre dans l'interaction. En effet, nous pouvons considérer que l'utilisateur note en partie le genre du film qu'il est aller voir, comme dans [ZWL<sup>+</sup>15]. Afin de rendre notre méthode plus robuste, nous proposons d'optimiser également les relations implicites suivantes (qui concernent toutes la prédiction de notes) :  $r_i = \{\mathcal{V} - \mathcal{A}, \mathcal{V} - \mathcal{S}, \mathcal{V} - \mathcal{O}, \mathcal{U} - \mathcal{G}\}$ , illustrées dans la figure 1 pour l'âge et le genre.

Pour quantifier ces liens implicites, nous utilisons des notes moyennes. Par exemple, la valeur de la relation entre un utilisateur  $i$  et un genre  $j$ ,  $R_{i,j}$ , est la note moyenne qu'il a donné aux films de genre  $j$  dans le passé. L'utilisation de ces relations permet d'apprendre à prédire explicitement les goûts des utilisateurs pour des caractéristiques, et pour un film, les goûts du public selon les attributs des utilisateurs.

## 3 Expériences

Nous travaillons sur la recommandation de films, qui propose un cadre et des données riches permettant

de tester nos hypothèses. L’originalité de nos travaux consiste à évaluer non seulement la prédiction de notes (au sens des moindres carrés) mais aussi la prédiction de données manquantes (comme le sexe des utilisateurs). Cette dernière tâche est évaluée en pourcentage de bonne classification.

### 3.1 Données

On utilise jeu de données Movielens 1M [HK15] constitué d’un million de notes sur une échelle de 1 à 5 données à des films par des utilisateurs. Ces notes sont enrichies par la tranche d’âge, le sexe et l’occupation (métier ou autre) des 6040 utilisateurs ainsi que du genre des 3952 films.

### 3.2 Protocole et paramètres

Nous nous comparons à des versions ré-implémentées de MF et MRMF. Pour la prédiction de notes, les performances sont données en validation croisée (10 permutations). Les représentations des différents concepts sont donc apprises sur 90% des notes. Pour la prédiction d’attributs manquants, nous avons basculé sur une validation croisée à 4 permutations, où seuls 75% des attributs sont utilisées pour l’apprentissage des représentations latentes.

Nous utilisons les paramètres suivants :  $K = 96$  (nombre de facteurs latents, choisi pour correspondre à [LHYT08])  $\epsilon = 0.01$  (pas du gradient pour la mise à jour),  $\lambda = 0.05$  (compromis de régularisation), Nous posons  $\alpha_c = 0.015$  pour la prédiction de notes puis  $\alpha_c = 0.2$  pour la prédiction de données manquantes et la visualisation : la prédiction de notes requiert une influence légère des attributs alors que la prédiction de données manquantes demande (logiquement) plus d’impact pour les attributs.

Le nombre d’itérations lors de l’apprentissage est déterminé par validation croisée pour chaque expérience.

### 3.3 Prédiction de notes

Commençons par la tâche la plus classique pour les systèmes de recommandation. Nous évaluons les performances en terme de RMSE (Root Mean Square Error) et MAE (Mean Absolute Error).

Le tableau 1 présente les résultats obtenus. Nous observons tout d’abord que notre approche  $WC MF_r$  (sur les relations explicites) obtient de meilleures performances que  $MRMF$  à partir des mêmes relations et

	RMSE	MAE
$FM$	$0.8424 \pm 0.0017$	$0.6624 \pm 0.0014$
$MRMF$	$0.8425 \pm 0.0016$	$0.6622 \pm 0.0012$
$WC MF_r$	$0.8398 \pm 0.0015$	$0.6599 \pm 0.0013$
$WC MF_{r_i}$	$0.8414 \pm 0.0013$	$0.6610 \pm 0.0013$
$WC MF_{r \cup r_i}$	<b><math>0.8389 \pm 0.0015</math></b>	<b><math>0.6588 \pm 0.0012</math></b>

TABLE 1 – RMSE et MAE avec validation croisée en 10 échantillons avec intervalles de confiance à 95%

d’un formalisme très proche<sup>1</sup>. Cela démontre l’intérêt d’optimiser des coûts pondérés selon les entités considérées.

Nous avons ensuite introduit l’optimisation des relations implicites ( $r_i$ ) dans notre algorithme (cf section 2.4). Seules, ces relations ne sont pas aussi intéressantes que les relations explicites ( $WC MF_{r_i}$ ); cependant, prendre en compte l’ensemble des relations permet de robustifier l’optimisation et améliore légèrement les performances de prédiction ( $WC MF_{r \cup r_i}$ ).

La figure 2 montre l’influence du paramètre  $\alpha_c$  en fonction du nombre de notes disponibles dans le jeu de données (sous échantillonné pour l’occasion), en utilisant toutes les relations. L’amplitude des variations de performances étant assez faible, nous avons normalisé les résultats par rapport à une base 0 correspondant à la factorisation matricielle; les courbes sont donc des écarts de performances RMSE (les courbes basses sont les meilleures). Le gain le plus important est observé pour un petit jeu de données, ce qui est intuitif : plus il y a de données, plus les profils sont riches et moins il y a à gagner. Par ailleurs, quand le nombre de notes augmente,  $\alpha_c$  gagne à être petit pour moins perturber les mises à jour venant des votes, d’une qualité croissante. Le réglage des  $\alpha_c$  est sensible, mais pour une valeur optimale, nous sommes meilleurs que la référence.

### 3.4 Prédiction d’attributs manquants

Notre formalisme permet de prédire directement les attributs manquants en comparant les sorties des produits scalaires entre entités. Cependant, il est plus efficace d’apprendre une régression logistique pour chaque attribut afin de prédire une valeur directement à partir de la représentation latente de l’item ou de l’utilisateur. Comme modèles de référence, nous utilisons les approches suivantes :

1. Notons que nous n’avons pas réussi à répliquer les résultats exacts de [LHYT08] (ni avec nos paramètres, ni avec les leur). L’implémentation qu’ils utilisent est inspirée de [Tik07] qui arrondit les prédictions, ce qui peut expliquer des différences de résultats.

	age	occupation	sexe	genre
<i>freq</i>	$34.7 \pm 0.9$	$12.6 \pm 0.5$	$71.7 \pm 0.8$	$25.4 \pm 1.0$
<i>notes</i>	<b><math>44.5 \pm 0.5</math></b>	$14.0 \pm 0.2$	$76.4 \pm 0.8$	$37.4 \pm 2.2$
<i>FM</i>	$40.3 \pm 0.8$	$17.5 \pm 0.4$	$76.3 \pm 0.3$	$38.2 \pm 0.5$
<i>WCMF<sub>r</sub></i>	$40.9 \pm 0.7$	<b><math>18.1 \pm 0.3</math></b>	<b><math>76.7 \pm 0.4</math></b>	<b><math>38.6 \pm 0.5</math></b>

TABLE 2 – Précision de la prédiction de données de manquantes avec intervalles de confiance à 95%

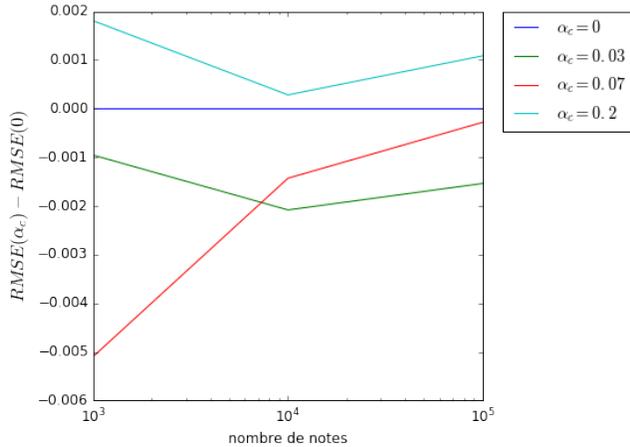


FIGURE 2 – Gains de RMSE en fonction du nombre de notes et du coefficient de pondération du contexte  $\alpha_c$ .

**Fréquence** : méthode naïve de prédiction de l'étiquette la plus fréquente.

**Notes** : correspond à une régression logistique apprise sur la matrice brute de notation. Pour un utilisateur, la prédiction du sexe ou de la classe d'âge se fait donc par pondération linéaire des notes données dans l'ensemble d'apprentissage. Pour un film, la prédiction du genre dépend des notes données par l'ensemble des utilisateurs.

**FM** : la régression logistique est cette fois apprise sur les représentations latentes issues de la factorisation matricielle.

Nous observons que le modèle WCMF est systématiquement au-dessus de la factorisation matricielle classique, ce qui est rassurant étant donné les contraintes structurantes ajoutées dans l'apprentissage des représentations WCMF. Globalement, les performances de notre modèle sont intéressantes. La seule exception concerne la classe d'âge : certains films sont très caractéristiques de certaines classes et il nous semble que les approches factorisées régularisent trop les sorties dans ce cas précis.

### 3.5 Visualisations

Toutes les entités,  $\mathcal{U}, \mathcal{V}, \mathcal{A}, \mathcal{G}, \mathcal{S}, \mathcal{O}$ , sont projetées dans le même espace latent, qui induit entre elles une similarité. Il est possible de visualiser ces similarités en utilisant l'algorithme TSNE [VH08] qui produit une sortie 2D où la distance euclidienne entre deux entités est d'autant plus faible que les entités sont similaires d'après une métrique spécifiée -ici la similarité cosinus, elle-même basée sur le produit scalaire que nous utilisons dans notre phase d'apprentissage-. La figure 3 montre les 100 films les plus populaires (en nombre de notes) et toutes les entités dans un même espace. La dispersion des représentations des attributs démontre leur capacité à structurer l'espace latent. Nous observons que les relations entre attributs sont conservées. Ainsi, l'attribut *K-12* qui correspond aux niveaux d'études primaire et secondaire, correspond bien avec la tranche d'âge *-18* (groupe compact au centre de la figure). Ces classes identifiant un public jeune sont elles-mêmes proches des genres *fantasy*, *film pour enfants* et *animation*, ce qui semble cohérent.

Quelques autres stéréotypes apparaissent dans la figure : les hommes sont plus proches de la science-fiction et des thrillers tandis que les femmes tendent vers les films de la classe *romance* ; les personnes se décrivant comme artistes ou écrivains sont proches des documentaires. Les films d'action et d'aventure sont nombreux parmi les plus commentés et occupent tout le haut de la figure.

## 4 Travaux connexes

La prise en compte du contexte est largement traitée dans la littérature, que ce soit avec des systèmes basés sur le contenu ou des systèmes hybrides mêlant factorisation matricielle et optimisations spécifiques [BB02]. Plusieurs philosophies se distinguent dans l'optimisation de représentations latentes. En misant sur une régularisation mutuelle, les approches basées sur l'alignement [NZ13, KL14] rapprochent les facteurs latents d'utilisateurs ou d'items similaires d'après leurs attributs.



Les attributs (items ou utilisateurs) peuvent également être vus en tant que translations dans l'espace latent, comme dans le modèle SVDFeature [CZL<sup>+</sup>12]. La représentation d'un film utilisée en inférence est alors la somme de celle qui lui est propre et de celle de son genre.

Les modèles latents pour l'apprentissage relationnel, introduits par [BW11] et étendus par [SCMN13] avec les Neural Tensor Network offrent un nouveau cadre à la recommandation hybride. L'idée est de régulariser les représentations en postulant qu'une même représentation doit pouvoir prédire plusieurs relations. Ces principes ont été utilisés par [LHYT08] pour la recommandation, mais en donnant le même poids à toutes les relations ce qui affecte négativement les résultats dans nos expériences. [BGY13] applique également ce principe dans une formulation convexe qui minimise la norme nucléaire pour régulariser les factorisations. Cette factorisation collective peut être ramenée à la factorisation d'une seule matrice, comme dans [Spi09] en plaçant des caractéristiques utilisateurs aux cotés des items, et des caractéristiques utilisateurs aux cotés des utilisateurs. Les relations sont traitées comme des notes : une valeur absente si elles sont fausses, et une constante non-nulle sinon. Cette constante module l'importance des relations mais d'une manière différente de notre approche. Sans cela, le principe de cette méthode se ramène à [LHYT08] Nous pouvons également citer la factorisation tensorielle [KABO10] qui prédit les notes à partir de produits tensoriels mais dont les performances n'ont pas été montrées en exploitant un grand nombre de relations.

D'autres approches se concentrent sur une modélisation probabiliste de relations binaires. La factorisation collective [GS15] apprend à prédire des attributs dans un cadre probabiliste. Mais cela contraint à discrétiser les notes et donc perdre de l'information. De plus, l'inférence d'attributs s'effectue par des produits scalaires des représentations ce qui dans notre approche produit des résultats inférieurs à ceux d'une régression logistique.

Dans un paradigme encore différent, le raisonnement dans les espaces latent pour la recommandation hybride est utilisé par [KFF<sup>+</sup>15] qui modélise une bonne recommandation à l'aide d'un ensemble de règles et utilise un formalisme de logique probabiliste [BBHG15] pour les spécifier.

Enfin, les relations que nous nommons implicites ont été utilisées par [ZWL<sup>+</sup>15] qui présente une approche centrée sur les caractéristiques des items. La fonction de coût de leur modèle peut être vue comme un cas particulier de la notre en ne considérant pas les liens di-

rects (utilisateur à item, item à attributs). Pour prédire les notes données à un item par un utilisateur, la méthode proposée est de prédire pour chacune des caractéristiques de l'item la note que l'utilisateur lui donne, et retrouver la note globale par combinaison linéaire. Cette méthode semble peu flexible et ne permet pas de modéliser d'autres caractéristiques que celles prévues au départ.

## 5 Conclusion

Nous avons décliné un cadre d'apprentissage relationnel à la recommandation en projetant l'ensemble des concepts de la base de données dans un espace latent commun. Nous avons montré l'intérêt d'utiliser des liens explicites mais aussi implicites entre des entités, en affichant des résultats compétitifs avec l'état de l'art pour la prédiction de notes. Notre espace semble également mieux structuré pour la prédiction des attributs et la flexibilité de ce cadre permet d'envisager l'incorporation de tags, de liens sociaux ou d'indicateurs temporels ou avec des entités plus nombreuses et plus connectées.

Cependant, la détermination automatique des coefficients de pondération reste une question ouverte. A terme, l'objectif est de faire converger l'apprentissage relationnel et la recommandation hybride ; la pondération des relations pourrait par exemple probablement améliorer les Neural Tensor Network [SCMN13] dans la construction de base de connaissances.

## 6 Remerciements

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

## Références

- [AT05] Gediminas Adomavicius and Alexander Tuzhilin. 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, 2005.
- [BB02] Robin Burke and Robin Burke. Hybrid Recommender Systems : Survey and Experiments 1. *User Modeling and User-Adapted Interaction*, pages 1–29, 2002.

- [BBHG15] Stephen H Bach, Matthias Broecheler, Bert Huang, and Lise Getoor. Hinge-Loss Markov Random Fields and Probabilistic Soft Logic. pages 1–46, 2015.
- [BGY13] Guillaume Bouchard, Shengbo Guo, and Dawei Yin. Convex Collective Matrix Factorization. *AISTATS*, 31 :144–152, 2013.
- [BL07] James Bennett and Stan Lanning. The Netflix Prize. *KDD Cup and Workshop*, pages 3–6, 2007.
- [BW11] Antoine Bordes and Jason Weston. Learning Structured Embeddings of Knowledge Bases. *Artificial Intelligence*, (Ben-gio) :301–306, 2011.
- [CZL<sup>+</sup>12] Tianci Chen, Weinan Zhang, Qiuxia Lu, Kailong Chen, Zhao Zheng, and Yong Yu. SVDFeature : A toolkit for feature-based collaborative filtering. *Journal of Machine Learning Research*, 13 :3619–3622, 2012.
- [GS15] Nitish Gupta and Sameer Singh. Collectively Embedding Multi-Relational Data for Predicting User Preferences. *arXiv :1504.06165 [cs]*, (2), 2015.
- [HK15] F Maxwell Harper and Joseph a Konstan. The MovieLens Datasets : History and Context. *ACM SIGMIS Database*, 46(4) :1–19, 2015.
- [HLGZ14] Tim Hussein, Timm Linder, Werner Gaulke, and Jürgen Ziegler. Hybreed : A software framework for developing context-aware hybrid recommender systems. *User Modeling and User-Adapted Interaction*, 24(1-2) :121–174, 2014.
- [KABO10] Alexandros Karatzoglou, Xavier Amatriain, Linas Baltrunas, and Nuria Oliver. Multiverse recommendation : n-dimensional tensor factorization for context-aware collaborative filtering. *Proceedings of the fourth ACM conference on Recommender systems - RecSys '10*, page 79, 2010.
- [KBV09] Y. Koren, R. Bell, and C. Volinsky. Matrix Factorization Techniques for Recommender Systems. *Computer*, 42(8) :42–49, 2009.
- [KFF<sup>+</sup>15] Pigi Kouki, Shobeir Fakhraei, James Foulds, Magdalini Eirinaki, and Lise Getoor. HyPER : A Flexible and Extensible Probabilistic Framework for Hybrid Recommender Systems. *Proceedings of the 9th ACM Conference on Recommender Systems*, pages 99–106, 2015.
- [KL14] Oleksandr Krasnoshchok and Yngve Lamo. Extended content-boosted matrix factorization algorithm for recommender systems. *Procedia Computer Science*, 35(C) :417–426, 2014.
- [Kor09] Yehuda Koren. Collaborative filtering with temporal dynamics. *Proc. of KDD '09*, pages 447–456, 2009.
- [LHYT08] Christoph Lippert, Stefan Hagen Weber, Huang Yi, and Volker Tresp. Relation Prediction in Multi-Relational Domains using Matrix Factorization. In *NIPS workshop on Structured Input, Structured Output (SISO)*, 2008.
- [MZL<sup>+</sup>11] Hao Ma, Dengyong Zhou, Chao Liu, Michael R Lyu, and Irwin King. Recommender Systems with Social Regularization Categories and Subject Descriptors. *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 287–296, 2011.
- [NZ13] Jennifer Nguyen and Mu Zhu. Content-boosted matrix factorization techniques for recommender systems. *Statistical Analysis and Data Mining*, 6(4) :286–301, 2013.
- [Pat07] Arkadiusz Paterek. Improving regularized singular value decomposition for collaborative filtering. *Categories and Subject Descriptors. KDD Cup. 07 August 12*, pages 39–42, 2007.
- [SCMN13] Richard Socher, Danqi Chen, Christopher D. Manning, and Andrew Y. Ng. Reasoning With Neural Tensor Networks for Knowledge Base Completion. *Proceedings of the Advances in Neural Information Processing Systems 26 (NIPS 2013)*, pages 1–10, 2013.
- [Spi09] Stephan Spiegel. A Hybrid Approach to Recommender Systems based on Matrix Factorization presented by. *Matrix*, 2009.
- [Tik07] Domonkos Tikk. On the Gravity Recommendation System. *Processing*, 2007 :22–30, 2007.
- [VH08] L J P Van Der Maaten and G E Hinton. Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 9 :2579–2605, 2008.

- [W3C14] W3C. RDF 1.1 Concepts and Abstract Syntax, 2014.
- [YWG14] Yonghong Yu, Can Wang, and Yang Gao. Attributes Coupling based Item Enhanced Matrix Factorization Technique for Recommender Systems. *XX(Xx)* :1–15, 2014.
- [ZWL<sup>+</sup>15] Chenyi Zhang, Ke Wang, Ee-peng Lim, Qinneng Xu, Jianling Sun, and Hongkun Yu. Are Features Equally Representative? A Feature-Centric Recommendation. *AAAI 2015 : Proceedings of the Twenty-ninth AAAI Conference on Artificial Intelligence*, pages 389–395, 2015.