



# Recommandation diversifiée via des processus ponctuels déterminantaux sur des graphes de connaissances

Lu Gan, Diana Nurbakova, Léa Laporte, Sylvie Calabretto

## ► To cite this version:

Lu Gan, Diana Nurbakova, Léa Laporte, Sylvie Calabretto. Recommandation diversifiée via des processus ponctuels déterminantaux sur des graphes de connaissances. *Extraction et Gestion des connaissances*, 2021, Jan 2021, Montpellier, France. pp.365-372. hal-03121974

**HAL Id: hal-03121974**

**<https://hal.archives-ouvertes.fr/hal-03121974>**

Submitted on 26 Jan 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Recommandation diversifiée via des processus ponctuels déterminantaux sur des graphes de connaissances

Lu Gan, Diana Nurbakova, Léa Laporte, Sylvie Calabretto

LIRIS, INSA Lyon, 7 Avenue Jean Cappelle, 69621 Villeurbanne Cedex  
prenom.nom@insa-lyon.fr

**Résumé.** Les systèmes de recommandation top- $N$  sont utilisés dans de nombreux domaines et intéressent autant les industriels que les chercheurs. Alors que l'amélioration de la prédiction de la pertinence des recommandations a fait l'objet de nombreuses recherches, d'autres facettes du problème comme la diversité ou l'explicabilité ont été moins abordées. Dans cet article, nous proposons une approche permettant d'améliorer la diversité tout en assurant un compromis avec la pertinence. Notre approche, DivKG, combine des graphes de connaissances et des Processus Ponctuels Déterminantaux (DPP). Les relations entre utilisateurs, items et autres entités, capturées par un graphe de connaissances, sont représentées via des plongements de graphes. Les vecteurs obtenus sont utilisés pour construire la matrice noyau des DPP, qui génèrent des recommandations diversifiées. Notre approche obtient de meilleurs résultats que l'état de l'art sur un jeu de données de recommandation de films. Cet article est un résumé en français de Gan et al. (2020), article court de la conférence SIGIR 2020.

## 1 Introduction

Les systèmes de recommandation *Top-N* visent à sélectionner l'ensemble des  $N$  items de plus grand intérêt pour un utilisateur. Largement utilisés par les moteurs commerciaux, ils ont fait l'objet de nombreuses études au cours des dernières décennies. Les approches de filtrage collaboratif (CF), utilisant les interactions entre utilisateurs et items, sont plébiscitées, notamment celles basées sur la factorisation de matrice (He et al., 2016; Salakhutdinov et Mnih, 2007). Elles fournissent généralement de très bons résultats en matière de pertinence de prédiction. Néanmoins, d'autres types de relations sémantiques explicites entre items et entités (autres que l'utilisateur) peuvent être intégrées pour améliorer la recommandation. Ces relations peuvent être représentées sous la forme d'un graphe de connaissances. Un exemple de graphe de connaissances pour la recommandation de films est présenté à la figure 1.

Les approches de plongements de graphes de connaissances (Bordes et al., 2013; Lin et al., 2015; Wang et al., 2014; Ji et al., 2015), qui capturent différents types de relations entre des entités variées (utilisateurs, items, etc) ont récemment été considérées afin de modéliser les graphes de connaissances et de les intégrer au processus de recommandation. Dans ce contexte, Zhang et al. (2016) ont utilisé une approche de plongements basée translation pour la recommandation. Xin et al. (2019) ont proposé RCF, une approche de filtrage collaboratif utilisant deux

niveaux de relations et exploitant les plongements de graphes de connaissances pour la recommandation top- $N$ . Ces approches améliorent la pertinence des recommandations en exploitant l'information structurée issue des graphes de connaissances.

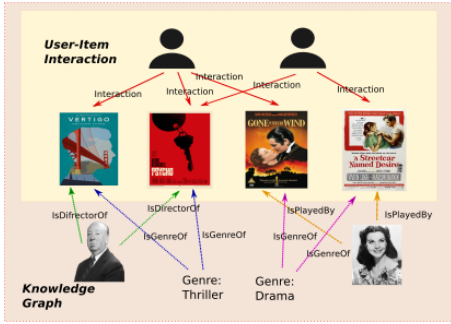


FIG. 1: Exemple de graphe de connaissances pour la recommandation de films, comprenant plusieurs types de relation (*Interaction*, *IsGenreOf*, etc.) entre différents types d'entités (user, item, genre, etc.).

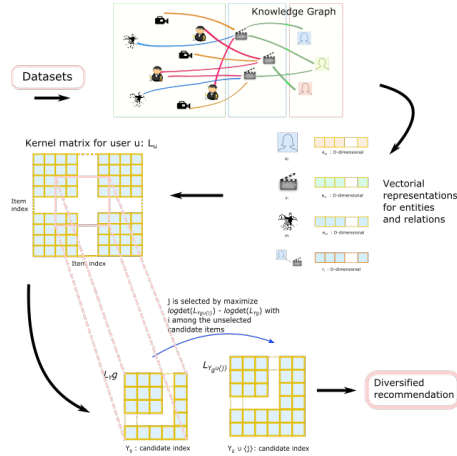


FIG. 2: Schéma général de **DivKG**.

Néanmoins, l'amélioration de la performance sans prise en compte de la diversité des recommandations peut conduire à proposer des items trop similaires aux utilisateurs. Par exemple, un système de recommandation sur un site de e-commerce qui, après avoir détecté l'intérêt d'un utilisateur pour un ordinateur portable, ne retournerait que des recommandations de PC portables, serait inefficace. En effet, il est peu probable qu'un utilisateur achète plus d'un PC à un moment donné. De façon générale, il est crucial de proposer des recommandations diversifiées aux utilisateurs. Dans ce cadre, He et al. (2016) ont proposé une approche de diversification nommée MMR, permettant de ré-ordonner les items à recommander via un processus itératif consistant à ajouter à chaque étape l'item pertinent le plus dissemblable à ceux déjà présents dans la liste. Borodin et al. (2017) ont proposé une extension de MMR garantissant l'existence d'une solution théorique. Néanmoins, ces solutions de diversification considèrent des similarités entre paires d'items, non sur toute la liste, et peuvent être sous-optimales.

Récemment, les Processus Ponctuels Déterminants (DPP) ont démontré leur efficacité pour l'amélioration de la diversité de plusieurs problèmes d'apprentissage automatique (Kulesza et Taskar, 2012; Gartrell et al., 2016; Warlop et al., 2019). Les DPPs sont une classe de modèles probabilistes sur des ensembles paramétrés avec une matrice semi-définie positive. Ils permettent de caractériser la pertinence de chaque élément de l'ensemble vis-à-vis d'un utilisateur ou d'une requête ainsi que la dissimilarité entre sous-ensembles d'items. Les DPP capturent la similarité entre items dans un espace unifié et proposent des listes d'items dissimilaires. L'utilisation des DPPs en recommandation peut améliorer les capacités de diversification des approches. L'enjeu est alors de définir une méthode de construction de la matrice noyau semi-définie positive, afin d'obtenir le meilleur compromis entre pertinence et diversité.

Dans cet article, nous considérons le problème du compromis entre pertinence et diversité pour la recommandation top- $N$ . Nous proposons une approche nommée **DivKG** combinant

plongements de graphe de connaissances et DPP, via une méthode de construction de la matrice noyau des DPP basée sur les plongements de graphe. A notre connaissance, **DivKG** est la première approche à combiner des plongements de graphes de connaissances et des DPP inférés par Maximum A Priori pour la recommandation top- $N$  diversifiée. Nos expérimentations sur le jeu de données MovieLens augmenté de données issues d’IMDb montrent que **DivKG** est plus performante que l’état de l’art en matière de pertinence et de diversité.

## 2 Recommandation diversifiée sur des graphes de connaissances

Nous présentons **DivKG**, l’approche pour la recommandation diversifiée sur des graphes de connaissances. Elle combine des plongements de graphes de connaissances et des processus ponctuels déterminantaux. Son schéma général est donné à la figure 2.

### 2.1 Plongements de graphe de connaissances pour la représentation d’entités et de relations

Incorporer des informations additionnelles dans les méthodes de filtrage collaboratif peut permettre d’améliorer la qualité des recommandations. Les graphes de connaissances ont démontré leur efficacité pour cette tâche (Zhang et al., 2016; Xin et al., 2019). Dans **DivKG**, nous modélisons l’ensemble des relations dans une seule structure de données à l’aide d’un graphe de connaissances. Contrairement aux travaux de Xin et al. (2019), où seuls les items constituent les sommets du graphe, nous proposons d’utiliser toutes les entités : utilisateurs, items et d’autres entités additionnelles (*e.g.* pour la recommandation de films : genre, acteur, etc ). Les interactions utilisateur-item utilisées en filtrage collaboratif sont vues uniquement comme une relation spécifique au sein du graphe de connaissances.

Formellement, nous représentons chaque relation comme un triplet  $(h, r, t)$  ayant une interprétation sémantique, où  $h$  et  $t$  sont respectivement deux entités liées par une relation  $r$ . Pour appliquer les plongements sur le graphe de connaissances, nous représentons  $h, r, t$  par les vecteurs  $\mathbf{v}_h, \mathbf{v}_r, \mathbf{v}_t$  respectivement. Nous utilisons des approches de plongements orientés translation (Bordes et al., 2013; Wang et al., 2014) telles que  $translation(\mathbf{v}_h, \mathbf{v}_r) \approx \mathbf{v}_t$ . Nous considérons la fonction de perte de marge  $\gamma$  suivante pour l’optimisation :  $Loss_{KGE} = \sum_{(h,r,t)} \sum_{(h',r',t')} [f_r(h, t) + \gamma - f_{r'}(h', t')]_+$ . Les triplets  $(h', r', t')$  sont dérivés des triplets de référence  $(h, r, t)$  en conservant la relation inchangée,  $r = r'$ , et en conservant une entité inchangée tandis que la deuxième est sélectionnée aléatoirement, i.e.  $h' = h, t' \neq t$ , ou  $t' = t, h \neq h'$ .  $f_r$  est la fonction de translation. Dans ces travaux, nous considérons deux fonctions de translation : TransE (Bordes et al., 2013) et TransH (Wang et al., 2014). TransE est définie par  $f_r(h, t) = \|\mathbf{v}_h + \mathbf{v}_r - \mathbf{v}_t\|_2$ , TransH par  $f_r(h, t) = \|(\mathbf{v}_h - \mathbf{w}_r^\top \mathbf{v}_h \mathbf{w}_r) + \mathbf{v}_r - (\mathbf{v}_t - \mathbf{w}_r^\top \mathbf{v}_t \mathbf{w}_r)\|_2$ , où  $\mathbf{w}_r$  est un vecteur de projection.

### 2.2 DPP pour la diversification des recommandations

Nous proposons d’exploiter les processus ponctuels déterminantaux (DPP) pour améliorer la diversité des recommandations. Selon Kulesza et Taskar (2012), un DPP sur un ensemble

de  $N$  éléments peut être interprété comme une modélisation sous forme d'un vecteur de présence/absence des éléments. Les variables de ce vecteur sont corrélées négativement : l'inclusion d'un élément diminue la probabilité d'inclusion des autres. Les valeurs de ces corrélations négatives sont dérivées de la matrice noyau qui détermine la similarité globale entre les paires d'éléments de telle façon que les éléments plus similaires ont moins de chances d'apparaître ensemble. Ainsi, les DPPs (Kulesza et Taskar, 2012) sont des modèles probabilistes qui décrivent la distribution d'une liste  $X$  dans un ensemble  $Y, Y \subseteq 2^X$  telle que la probabilité de sélection de sous-ensemble  $S, S \in Y$  est proportionnelle au déterminant de la matrice noyau du DPP. La matrice noyau d'un DPP est semi-définie positive, de taille  $|Y| \times |Y|$ . Les éléments diagonaux représentent l'affinité inhérente d'un item (pour un utilisateur donné par exemple), tandis que les éléments non diagonaux représentent la similarité entre items pour chaque paire d'items  $(i, j)_{i \neq j}$ . Nous proposons de combiner les DPP aux représentations obtenues par les plongements de graphes de connaissances (cf. section 2.1). Pour cela, nous devons d'une part construire la matrice noyau et d'autre part prédire les items à recommander.

### 2.2.1 Construction de la matrice noyau des DPP

Pour construire la matrice noyau  $\mathbf{L}_u$  pour chaque utilisateur  $u$ , nous définissons les matrices auxiliaires  $\mathbf{A}_u$  et  $\mathbf{D}_u$ .  $\mathbf{A}_u$  est une matrice diagonale représentant l'affinité de l'utilisateur  $u$  pour chaque item candidat, telle que  $\mathbf{A}_u = \text{diag}(a_1, \dots, a_m)$ , avec  $a_i = \frac{e^{-(f_r(u,i)-\delta)}}{\sum_{j \in X, j \neq i} e^{-(f_r(u,j)-\delta)}}$ , où  $m$  est le nombre d'items candidats,  $f_r(u, i)$  est la fonction de translation retenue pour les plongements de graphes de connaissances et  $\delta$  est la moyenne des  $f_r(u, i)$  pour l'utilisateur  $u$ . Nous utilisons la fonction softmax pour la normalisation. La matrice  $\mathbf{D}_u$  représente les similarités item-item, telles que  $\mathbf{D}_u = [d_{ij}]^{m \times m}$ , avec  $d_{ij} = \frac{e^{-f_{r_0}(i,j)}}{\sum_{k \in X, k \neq i} e^{-f_{r_0}(i,k)}}$ , et  $d_{ii} = 0$ .  $f_{r_0}(i, j)$  est le plongement des items  $i$  et  $j$  et de la relation  $r_0$ , tel que  $v_{r_0} = \vec{0}$  si  $i$  et  $j$  appartiennent à la même *catégorie* (relation). La matrice noyau  $\mathbf{L}_u$  d'un utilisateur  $u$  est alors définie comme :  $\mathbf{L}_u = \alpha \mathbf{A}_u + \mathbf{D}_u$ , où  $\alpha$  permet d'ajuster le compromis entre affinité de l'utilisateur pour les items et similarité entre les items, donc entre qualité de prédiction et diversité.

### 2.2.2 Prédiction par inférence Maximum A Priori (MAP)

Une fois la matrice noyau construite pour chaque utilisateur  $u$ , nous sélectionnons une liste  $S$  de  $N$  items parmi tous les items candidats, telle que :  $S_{map} = \underset{S \in Y, |S|=N}{\text{argmax}} \log \det(\mathbf{L}_S)$  où  $\mathbf{L}_S$  est la matrice noyau  $\mathbf{L}_u$  indexée par les items de la liste  $S$ . Pour rappel, la probabilité de sélectionner un sous-ensemble donné  $S$  est proportionnelle au déterminant de la matrice noyau indexée. DPP garantit une diversité des items sélectionnés par définition. Ainsi, les items sélectionnés avec la plus grande valeur du logarithme du déterminant sont théoriquement les plus divers et les plus pertinents pour un utilisateur  $u$ . Le problème d'optimisation associé est de difficulté NP. Pour le résoudre, nous utilisons un algorithme glouton proposé par Chen et al. (2018) permettant d'inférer une liste approximée par Maximum A Priori (MAP). Dans la suite, nous noterons FastDPP notre approche combinant DPP et inférence par MAP.

### 3 Evaluation

#### 3.1 Protocole expérimental

**Jeux de données.** Nous avons construit un jeu de données comportant de multiples relations à partir de deux jeux de données reconnus dans la communauté : MovieLens-100K<sup>1</sup> (noté ML-100K) et IMDb. ML-100K comporte 100 000 notes comprises entre 1 et 5 fournies par 943 utilisateurs pour 1 682 films. Chaque utilisateur a noté 20 films a minima. La matrice des notes est extrêmement creuse (93.70% des couples utilisateur-film sans note). Nous avons considéré comme retour implicite positif les scores supérieurs ou égaux à 4. Le jeu de données IMDb<sup>2</sup> contient des informations auxiliaires pour plus de 947 000 films. Nous en avons extrait 13 catégories d’information (genre, acteur, compositeur, réalisateur, etc), que nous avons combiné au jeu de données ML-100K afin de définir les relations au sein du graphe de connaissances.

**Protocole d’évaluation.** Nous évaluons **DivKG** selon la qualité de prédiction des recommandations et leur diversité à deux étapes : l’étude de la qualité des plongements pour la recommandation non diversifiée et l’évaluation de la recommandation diversifiée via les DPP.

*Qualité de prédiction des recommandations.* Nous considérons deux mesures usuelles : la *normalized discounted cumulative gain*  $NDCG@N$  et le *hit ratio*  $hit@N$ . Calculées pour chaque utilisateur test, elles sont ensuite moyennées sur l’ensemble des utilisateurs.

*Diversité des recommandations.* Nous considérons deux mesures de diversité définies sur la liste d’items (Chen et al., 2018) :  $ILAD = \text{moyenne}_{u \in U} \text{moyenne}_{i, j \in R_u, i \neq j} (1 - S_{i, j})$  et  $ILMD = \text{moyenne}_{u \in U} \min_{i, j \in R_u, i \neq j} (1 - S_{i, j})$ , avec  $U$  l’ensemble des utilisateurs,  $R_u$  la liste des recommandations pour l’utilisateur  $u$  et  $S_{ij}$  la similarité entre les items  $i$  et  $j$ . Ces mesures sont calculées pour chaque liste, puis moyennées sur l’ensemble des listes.

*Evaluation.* Nous adoptons une stratégie de type *leave-one-out*, usuelle dans la littérature (Xin et al., 2019) pour les plongements de graphes de connaissances et les DPP. Pour évaluer la qualité des plongements de graphes de connaissances (section 2.1), nous procédons de la façon suivante. Pour chaque utilisateur, une interaction utilisateur-item est sélectionnée aléatoirement comme jeu de test, tandis que les interactions restantes sont aléatoirement réparties entre jeu d’apprentissage et jeu de validation avec un ratio de 80% : 20% respectivement. Pour évaluer la recommandation diversifiée via les DPP (Section 2.2), nous (1) tirons aléatoirement un item pour chaque utilisateur à partir de ses interactions historiques, (2) pour cet item, nous choisissons les 50 items les plus similaires en terme de résultats de leurs plongements de graphes de connaissances, (3) nous prenons l’union de l’item sélectionné et de ces 50 items pour construire la matrice noyau de DPP et ensuite faire la prédiction avec MAP.

**Algorithmes de référence.** Nous utilisons trois algorithmes de référence pour évaluer la recommandation non diversifiée via les plongements de graphes de connaissances : BPRMF, RC et FISM. BPRMF (Rendle et al., 2009), basé sur la factorisation de matrice, apprend sur des retours de pertinence implicites sans considérer les relations entre entités. RCF (Xin et al., 2019), basé sur des graphes de connaissances, considère les interactions utilisateur-item ainsi

1. <https://grouplens.org/datasets/movielens/100k/>

2. Lien au 26/11/2020 : <https://datasets.imdbws.com/>

que d’autres relations entre items. FISM (Kabbur et al., 2013) est une méthode de filtrage collaboratif centrée item. Pour garantir une comparaison équitable en matière de diversité, nous combinons chaque approche non diversifiée avec deux modèles de diversification : l’approche FastDPP que nous proposons (section 2.2) et la méthode de référence *Maximal Marginal Relevance* (MRR) (He et al., 2016). MRR est une approche de ré-ordonnement utilisée pour la génération de résumés, permettant de réduire la redondance en conservant la pertinence.

### 3.2 Résultats

**Plongements de graphes de connaissances.** Le tableau 1 montre une amélioration de la qualité de prédiction, quelle que soit la mesure considérée, par l’utilisation des plongements de graphes de connaissances orientés translation. Ainsi, TransE et TransH, les deux méthodes de plongements orientés translation que nous avons considérées, obtiennent de meilleurs résultats que l’ensemble des approches de l’état de l’art, y compris la méthode RCF utilisant aussi des relations entre entités.<sup>3</sup> Par construction, le graphe de connaissances que nous avons proposé tient compte de multiples relations entre les entités (primaires et auxiliaires). Ainsi, une interaction utilisateur-élément (film) devient plus nuancée que la simple considération d’une note ou du nombre de relations comme dans le cas de RCF. Ces nuances sont captées par les méthodes de plongement considérées, TransE et TransH, ce qui se traduit par l’amélioration de résultats de recommandation par rapport aux autres algorithmes de l’état de l’art.

TAB. 1: Résultats avant diversification (dimension = 75, learning rate = 0.001)

Metric	Hit@5	Hit@10	Hit@20	NDCG@5	NDCG@10	NDCG@20
BPRMF	0.1394	0.2200	0.3240	0.0888	0.1150	0.1412
FISM	0.1182	0.2041	0.3160	0.0746	0.1023	0.1304
RCF	0.1442	0.2179	0.3261	0.0888	0.1123	0.1393
TransE	0.1879	0.2842	0.4087	0.1253	<b>0.1561</b>	0.1876
TransH	<b>0.1917</b>	<b>0.2861</b>	<b>0.4123</b>	<b>0.1257</b>	<b>0.1561</b>	<b>0.1878</b>

**Recommandation diversifiée.** Le tableau 2 présente la performance des méthodes de recommandation diversifiée en matière de qualité de prédiction et de diversité. Nos méthodes, combinant FastDPP avec TransE et TransH, sont respectivement notées  $\text{DivKG}_E$  and  $\text{DivKG}_H$ . Nous constatons qu’elles obtiennent systématiquement de meilleurs résultats que les approches de l’état de l’art, pour l’ensemble des mesures considérées. De plus, la combinaison des méthodes de l’état de l’art avec FastDPP obtient généralement de meilleurs résultats qu’avec MRR.

## 4 Conclusions

Dans cet article, nous avons proposé **DivKG**, une approche de recommandation top- $N$  diversifiée, combinant des plongements de graphes de connaissances et des processus ponc-

<sup>3</sup>. Nous avons testé plusieurs configurations de paramètres. Ici, nous décrivons les résultats obtenus pour la meilleure configuration trouvée.

TAB. 2: Résultats pour la recommandation diversifiée avec  $\alpha=0.9$ 

	BPRMF +MMR	BPRMF +FastDPP	RCF +MMR	RCF +FastDPP	TransE +MMR	DivKG <sub>E</sub>	TransH +MMR	DivKG <sub>H</sub>
Hit@10	0.2821	0.3065	0.2842	0.3001	0.2768	0.3160	0.2693	<b>0.3175</b>
NDCG@10	0.0789	0.0848	0.0785	0.0798	0.0774	0.1176	0.0705	<b>0.1178</b>
<i>ILAD</i>	0.9683	0.9922	0.9698	0.9923	0.9911	<b>0.9959</b>	0.9898	0.9956
<i>ILMD</i>	0.8867	0.9726	0.8886	0.9729	0.9183	<b>0.9768</b>	0.8951	0.9690

tuels déterminantaux. Les graphes de connaissances permettent de capturer différentes relations entre items, utilisateurs, et d’autres entités auxiliaires. La diversité des recommandations est assurée par la méthode de construction des noyaux des DPP sur les graphes de connaissances générés. **DivKG** garantit un compromis entre diversité et pertinence. Elle obtient expérimentalement de meilleurs résultats que l’état de l’art en matière de pertinence et de diversité.

**Remerciements.** Lu Gan bénéficie d’une bourse de doctorat du *China Scholarship Council*, no. 201801810094, pour effectuer sa thèse de doctorat à l’INSA Lyon, France.

## Références

- Bordes, A., N. Usunier, A. Garcia-Durán, J. Weston, et O. Yakhnenko (2013). Translating embeddings for modeling multi-relational data. In *Proc. of the 26th International Conference on Neural Information Processing Systems - Volume 2, NIPS’13*, pp. 2787–2795.
- Borodin, A., A. Jain, H. C. Lee, et Y. Ye (2017). Max-sum diversification, monotone submodular functions, and dynamic updates. *ACM Trans. Algorithms* 13(3).
- Chen, L., G. Zhang, et H. Zhou (2018). Fast greedy map inference for determinantal point process to improve recommendation diversity. In *Proc. of the 32Nd International Conference on Neural Information Processing Systems, NIPS’18*, pp. 5627–5638.
- Gan, L., D. Nurbakova, L. Laporte, et S. Calabretto (2020). Enhancing recommendation diversity using determinantal point processes on knowledge graphs. In *Proc. of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pp. 2001–2004.
- Gartrell, M., U. Paquet, et N. Koenigstein (2016). Bayesian low-rank determinantal point processes. In *Proc. of the 10th ACM Conference on Recommender Systems*, pp. 349–356.
- He, X., H. Zhang, M.-Y. Kan, et T.-S. Chua (2016). Fast matrix factorization for online recommendation with implicit feedback. In *Proc. of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR ’16*, pp. 549–558.
- Ji, G., S. He, L. Xu, K. Liu, et J. Zhao (2015). Knowledge graph embedding via dynamic mapping matrix. In *Proc. of the 53rd Annual Meeting of the Association for Computational*



- Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers)*, pp. 687–696.
- Kabbur, S., X. Ning, et G. Karypis (2013). Fism : Factored item similarity models for top-n recommender systems. In *Proc. of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 659–667.
- Kulesza, A. et B. Taskar (2012). Determinantal point processes for machine learning. *Found. Trends Mach. Learn.* 5(2-3), 123–286.
- Lin, Y., Z. Liu, M. Sun, Y. Liu, et X. Zhu (2015). Learning entity and relation embeddings for knowledge graph completion. In *Proc. of the 29th AAAI Conference on Artificial Intelligence*, AAAI'15, pp. 2181–2187.
- Rendle, S., C. Freudenthaler, Z. Gantner, et L. Schmidt-Thieme (2009). Bpr : Bayesian personalized ranking from implicit feedback. In *Proc. of the 25th Conference on Uncertainty in Artificial Intelligence*, UAI '09, pp. 452–461.
- Salakhutdinov, R. et A. Mnih (2007). Probabilistic matrix factorization. In *Proc. of the 20th International Conference on Neural Information Processing Systems*, pp. 1257–1264.
- Wang, Z., J. Zhang, J. Feng, et Z. Chen (2014). Knowledge graph embedding by translating on hyperplanes. In *Proc. of the 28th AAAI Conference on Artificial Intelligence*, pp. 1112–1119.
- Warlop, R., J. Mary, et M. Gartrell (2019). Tensorized determinantal point processes for recommendation. In *Proc. of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, pp. 1605–1615.
- Xin, X., X. He, Y. Zhang, Y. Zhang, et J. Jose (2019). Relational collaborative filtering : Modeling multiple item relations for recommendation. In *Proc. of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, pp. 125–134.
- Zhang, F., N. J. Yuan, D. Lian, X. Xie, et W.-Y. Ma (2016). Collaborative knowledge base embedding for recommender systems. In *Proc. of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pp. 353–362.

## Summary

Top- $N$  recommendations are applied in various real life domains and keep attracting intense attention from researchers and industry. While *accuracy* has been the prevailing issue of the recommendation problem for the last decades, other facets of the problem, namely *diversity* and *explainability*, have received much less attention. In this paper, we focus on enhancing diversity of top- $N$  recommendation, while ensuring the trade-off between accuracy and diversity. We propose an effective framework **DivKG** leveraging knowledge graph embedding and determinantal point processes (DPP). First, we capture different kinds of relations among users, items and additional entities through a knowledge graph structure. Then, we represent both entities and relations through graph embedding, using all kind of historical interaction. We use these representations to construct kernel matrices of DPP in order to make top- $N$  diversified predictions. Our empirical results show substantial improvement over the state-of-the-art regarding accuracy and diversity metrics. This paper is a french translation of Gan et al. (2020).