



HAL
open science

Apprentissage multi-vues pour la recommandation dans le domaine du pneumatique

Thomas Ranvier, Khalid Benabdeslem, Kilian Bourhis, Bruno Canitia

► To cite this version:

Thomas Ranvier, Khalid Benabdeslem, Kilian Bourhis, Bruno Canitia. Apprentissage multi-vues pour la recommandation dans le domaine du pneumatique. Extraction et Gestion des Connaissances (EGC) 2021, Jan 2021, Montpellier, France. pp.261-268. hal-04230586

HAL Id: hal-04230586

<https://hal.science/hal-04230586v1>

Submitted on 6 Oct 2023

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.

Apprentissage multi-vues pour la recommandation dans le domaine du pneumatique

Thomas Ranvier*, Khalid Benabdeslem*, Kilian Bourhis**, Bruno Canitia**

*Université Lyon 1, 43 Boulevard du 11 novembre 1918, Villeurbanne cedex 69622
ranvier.thomas.pro@gmail.com,
khalid.benabdeslem@univ-lyon1.fr

**Lizeo IT, 42 Quai Rambaud, 69002 Lyon
kilian.bourhis,bruno.canitia@lizeo-group.com,

Résumé. Nous utilisons constamment des systèmes de recommandation, souvent sans même nous en apercevoir. Ils construisent un profil nous correspondant afin de nous recommander le contenu le plus susceptible de nous intéresser. Les données représentant les utilisateurs, leurs interactions avec le système ou encore les produits peuvent provenir de sources différentes et être de natures diverses. Notre objectif est d'appliquer les approches d'apprentissage multi-vues pour améliorer notre système de recommandation dans le domaine du pneumatique et lui permettre de gérer au mieux des données organisées en de multiples vues. Nous présentons une étude comparative entre plusieurs modèles multi-vues de l'état de l'art appliqués à nos données industrielles. Celle-ci nous a permis de démontrer la pertinence de l'utilisation de l'apprentissage multi-vues au sein de systèmes de recommandation.

1 Introduction

L'objectif d'un système de recommandation est d'estimer les produits les plus susceptibles d'intéresser un utilisateur (Betru et al. (2017)). Il existe deux principales tâches de recommandation : la prédiction d'une note et l'établissement d'un classement. La prédiction d'une note vise à estimer la note que l'utilisateur pourrait donner à un produit et s'appuie généralement sur les notes attribuées à d'autres produits par ce même utilisateur. L'établissement d'un classement vise à fournir une liste classée de n produits dans l'ordre estimé de préférence de l'utilisateur, généralement appelé un "top- n " (Zhang et al. (2019)). Il existe trois principales catégories de systèmes de recommandations (Adomavicius et Tuzhilin (2005)). Les *Content-based recommendations*, où l'utilisateur se voit recommander des produits similaires à ceux qu'il a déjà consultés et appréciés. Le *Collaborative Filtering*, où l'utilisateur se voit recommandé des produits que des personnes avec des goûts similaires ont apprécié. Les *hybrid approaches*, qui combinent les deux méthodes décrites précédemment. Une quatrième catégorie moins courante sont les systèmes de recommandations basés sur les sessions des utilisateurs. Les recommandations se font alors sur les similarités entre les habitudes de navigation des utilisateurs et leurs relations aux produits consultés.

Beaucoup de données sont collectées depuis des sources distinctes, exploiter intelligemment ces différentes vues peut permettre d'améliorer les performances d'un système de classification. Pour que l'apprentissage multi-vues soit efficace il est nécessaire que les vues respectent certains principes, sans quoi l'utilisation de plusieurs vues pourrait entraîner une dégradation des performances. Les deux grands principes de l'apprentissage multi-vues sont le principe du consensus et celui de la complémentarité (Xu et al. (2013)). Dans le cadre d'un apprentissage supervisé, le principe du consensus définit qu'en minimisant les désaccords entre chaque vue, le taux d'erreur sur chaque vue en sera également minimisé. C'est-à-dire qu'un système capable de faire corrélérer les résultats obtenus à partir de différentes vues minimisera par la même occasion les erreurs de classification sur chacune de ces vues. Le principe de complémentarité définit que si chaque vue contient des connaissances que les autres ne possèdent pas, alors ces multiples vues peuvent être exploitées ensemble pour décrire les données de manière compréhensive et précise et mener à de meilleures performances d'apprentissage..

Un système de recommandation peut bénéficier de l'apprentissage multi-vues à la condition que les données exploitées permettent l'application des principes du consensus et de la complémentarité. Les applications concrètes de l'approche multi-vues dans le domaine de la recommandation utilisent généralement des modèles composites issus de l'apprentissage profond. (Elkahky et al. (2015)) introduisent un modèle multi-vues profond, par la suite (Song et al. (2016)) enrichissent ce modèle en y intégrant un sous-modèle permettant la gestion d'une ou plusieurs vue(s) temporelle(s). Plus tard le concept de mécanismes d'attention a été introduit dans ce domaine avec le modèle de (Liang et al. (2020)).

Au sein de cet article, nous décrivons dans un premier temps les modèles issus de l'état de l'art qui seront évalués sur nos données industrielles par une étude comparative. Nous détaillerons ensuite les données utilisées ainsi que le protocole expérimental utilisé pour l'étude. Enfin, nous présenterons les résultats obtenus par notre étude comparative.

2 Apprentissage multi-vues pour la recommandation

2.1 Vues disponibles

Rezulteo est un comparateur en ligne au sein duquel se trouve un système de recommandation, ce système répond à des contraintes telles que le problème du "cold-start" (Schein et al. (2002)), des interactions implicites entre l'utilisateur et les produits et aucune confirmation d'achat après recommandation (redirection vers sites revendeurs). Cette section présente les cinq vues disponibles pour le projet Rezulteo et leur évaluation individuelle.

1. **Données de session des utilisateurs :** Ces données séquentielles sont issues des historiques de sessions des utilisateurs. Elles réunissent les actions faites par les utilisateurs dans l'ordre chronologique, la dimension temporelle y est donc primordiale.
2. **Données expertes sur les produits :** Ce sont des données métiers récoltées auprès des manufacturiers et qui caractérisent chaque produit de Rezulteo, elles sont statiques et associent une représentation unique et abstraite à chaque produit.
3. **Vecteurs latents des utilisateurs :** Les vecteurs latents des utilisateurs sont générés par une factorisation de matrices sur les interactions des utilisateurs via le modèle eALS : Element-wise Alternating Least Square (He et al. (2016)). Le but est de réutiliser l'une des deux composantes (matrice utilisateur) nécessaire à la reconstruction de

la matrice d’interaction implicite pour générer une représentation unique et abstraite pour chaque utilisateur (Kenaan et al. (2020)).

4. **Vue de comparabilité** : Cette vue statique est obtenue à partir de coefficients de similarité qui sont calculés entre chaque paire de produits. Ces derniers sont déterminés par un système expert qui se base sur des indicateurs métiers (âge du produit, volume de vente et sa présence sur les différentes parts de marché, etc.).
5. **Vue de compatibilité** : Cette vue statique liste tous les produits qui sont compatibles avec les sessions des utilisateurs, chaque session étant associée à une requête qui détermine cette compatibilité. Chaque véhicule n’étant compatible qu’avec un ensemble limité de pneumatiques, cette vue représente simplement la liste des produits compatibles avec la session de l’utilisateur. Elle se présente sous la forme d’un simple vecteur de valeurs booléennes de dimension n , avec n le nombre total de produits.

Les métriques utilisées pour évaluer les modèles sont les mêmes que celles utilisées au sein des travaux précédents (Bourhis et al. (2019); Kenaan et al. (2020)). Il s’agit du **HR (Hit Rank)**, qui mesure la capacité du système à donner toutes les solutions pertinentes et du **NDCG (Normalized Discounted Cumulative Gain)**, qui mesure la qualité du classement de la recommandation. Toutes deux varient de 0 à 100%.

La table 1 présente la qualité de chacune des vues disponibles pour ce projet en comparaison avec une vue composée de données aléatoires. Chaque valeur est obtenue par la moyenne de 10 expériences de 20 epochs chacune. Les vues statiques sont évaluées par un modèle de neurones à propagation avant. La vue séquentielle des données de sessions est évaluée avec un 3D-CNN (3D Convolutional Neural Network), un modèle à convolutions capable de gérer des données temporelles grâce à des convolutions tridimensionnelles (Tuan et Phuong (2017)). L’intervalle de confiance utilisé est l’écart-type.

Vue évaluée	HR@100	NDCG@100
Aléatoire	66.33 ± 0.08	19.83 ± 0.04
Sessions	82.01 ± 0.30	32.42 ± 0.36
Données expertes	83.58 ± 0.10	34.16 ± 0.13
Vue de comparabilité	83.83 ± 0.13	31.82 ± 0.13
Vue de compatibilité	85.37 ± 0.17	33.89 ± 0.18
Vecteurs des utilisateurs	89.56 ± 0.10	37.21 ± 0.18

TAB. 1 – Évaluation des vues disponibles en comparaison avec une vue aléatoire.

L’évaluation se faisant sur un top 100 la vue aléatoire obtient un HR et un NDCG pouvant paraître élevé, nous pouvons tout de même constater des résultats nettement supérieurs pour les autres vues. L’évaluation de la qualité des vues montre donc que toutes les vues disponibles offrent des informations utiles pour la recommandation.

2.2 Modèles évalués

Notre objectif est d’étudier les performances de l’apprentissage multi-vues appliqué au domaine de la recommandation sur des données industrielles. Notre choix s’est porté sur trois

modèles que nous comparons à notre Baseline issue des travaux précédemment réalisés sur le projet (Bourhis et al. (2019); Kenaan et al. (2020)).

Baseline (Kenaan et al. (2020)). Ce modèle est un 3D-CNN, directement issu des travaux précédents. Les données de sessions, données expertes et vecteurs latents des utilisateurs sont concaténées de sorte à conserver la dimension temporelle et servent d'entrée à ce modèle.

MV-DNN : Multi-View Deep Neural Network (Elkahky et al. (2015)). Ce modèle est composé d'autant de branches en parallèle que de vues différentes utilisées, chaque branche construit une représentation abstraite d'une vue. Le MV-DNN possède une vue pivot et vise à maximiser la somme des similarités entre le pivot et les autres vues en utilisant une fonction de similarité. Dans un souci de généralisation et d'adaptation à notre cas de figure, la version de ce modèle implémentée n'utilise ni vue pivot, ni fonction de similarité. Dans notre cas les résultats de toutes les branches sont concaténés pour servir d'entrée à un réseau de neurones final. Celui-ci permet d'obtenir un vecteur associant un poids de recommandation à chaque produit de Rezulteo. Toutes les branches du modèle sont constituées de réseaux de neurones à propagation avant ayant la même architecture, chaque branche est donc en mesure de gérer une vue statique. La vue séquentielle des sessions ne peut pas être exploitée par ce modèle.

TDSSM : Temporal Deep Semantic Similarity Model (Song et al. (2016)). Le TDSSM étend l'architecture d'un MV-DNN en intégrant un ou plusieurs modèles temporels (récurrent, 3D-CNN, etc.) au sein de ses branches. L'architecture du modèle développé est générique et peut gérer autant de vues statiques et séquentielles que nécessaire.

MV-AFM : Multi-View Attentional Factorization Machines (Liang et al. (2020)). Le modèle MV-AFM introduit deux nouveaux concepts aux architectures multi-vues, le concept d'attention hiérarchique et celui d'interactions par paires entre les vues. L'utilisation de mécanismes d'attention permet de pondérer chaque vue selon son importance et ainsi de mieux gérer l'aspect multi-vues. La combinaison des vues par paires permet de corrélérer des informations entre les vues et ainsi permettre l'apparition de connaissances utiles. Au sein du papier initial (Liang et al. (2020)), le modèle est mis en application sur des jeux de données issus de Google Play et de l'Apple App Store. L'architecture du modèle a été légèrement modifiée pour être adaptée à nos données, elle est présentée sur la figure 1. Le modèle possède une branche par vue, chaque branche commence par un modèle capable de gérer la nature de la vue attribuée (statique ou séquentielle), permettant d'obtenir un vecteur de taille fixe. Ce vecteur passe par une couche "d'embedding" générant x vecteurs de taille y grâce à x couches linéaires de y neurones chacune. Ces x vecteurs passent par un mécanisme d'attention (Feature-level attention), celui-ci génère un vecteur de contexte composé des informations les plus importantes pour chaque vue. La couche suivante (Pair-wise views interactions) opère un produit Hadamard entre chaque paire de vecteurs de contexte, permettant de modéliser les interactions entre toutes les paires de vues. Le mécanisme d'attention suivant (View-level attention) permet de générer un vecteur de contexte global pondérant l'importance des vues et de leurs interactions. Toutes les branches sont ainsi réunies en une seule et la sortie finale du modèle est obtenue à travers deux dernières couches linéaires.

Les différents mécanismes d'attention peuvent tous être modélisés par l'équation 1. La matrice F est composée des x vecteurs d'entrée à cette couche, tous de taille y , α est le vecteur d'alignement de taille x et c le vecteur de contexte généré de taille y . Les variables q , W et b sont des poids et des biais qui sont appris lors de l'entraînement global du modèle.

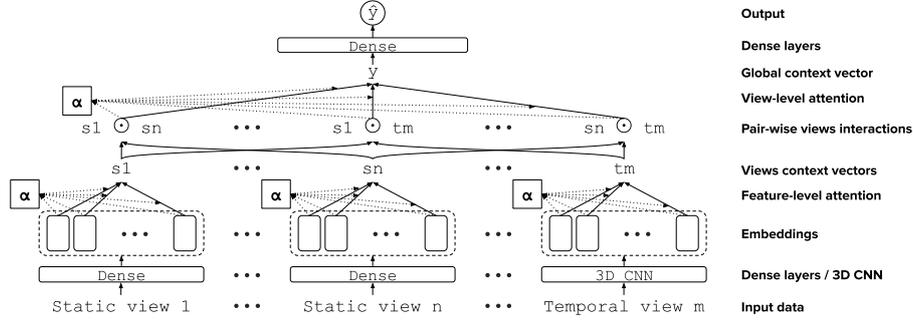


FIG. 1 – Architecture de la version du MV-AFM implémentée.

$$e_i = q \cdot \tanh(\mathbf{W} \cdot \mathbf{F}_i + \mathbf{b}) \quad \alpha = \frac{\exp(e_i)}{\sum_{k=1}^x \exp(e_k)} \quad \mathbf{c} = \sum_{i=1}^x \alpha_i \cdot \mathbf{F}_i \quad (1)$$

La couche modélisant les interactions par paires entre les vues (Pair-wise views interactions) prend en entrée les vecteurs de contexte obtenus avec les mécanismes d’attention précédents (Feature-level attention). Le nombre de vues résultantes est de $m + \frac{m \cdot (m-1)}{2}$, avec m le nombre de vues initiales. Chaque nouvelle vue est générée par la somme du produit Hadamard entre deux vecteurs de contexte. La sortie de cette couche est composée des m vecteurs d’entrée et des $\frac{m \cdot (m-1)}{2}$ vecteurs générés, sur la figure 1 ne figurent pas les m vecteurs d’entrée à ce niveau dans un souci de clarté.

3 Données industrielles et protocole expérimental

Nous avons mené nos expérimentations sur des données réelles provenant d’un comparateur en ligne de pneumatique. Nous avons deux jeux de données à notre disposition. Le premier, noté D1, comporte quatre vues (sessions, vecteurs utilisateurs, données expertes et vue de comparabilité), seule la vue de comparabilité y est manquante. Le second, noté D2, est un sous-ensemble du précédent de plus petite taille qui comporte les cinq vues disponibles. Nous avons construit la vue de comparabilité de D2 en ré-exécutant les requêtes des utilisateurs pour définir la liste des produits compatibles avec la requête.

Le jeu D1 est constitué de 114 359 sessions pour 102 613 utilisateurs et 7 726 produit, le jeu D2 contient 50 241 sessions pour 46 148 utilisateurs et 3 268 produits. La volumétrie des données montre environ une session par utilisateur, ce qui signifie que les utilisateurs n’utilisent généralement le système de recommandation qu’une fois, il est donc impossible de créer un profil d’utilisateur. Sans profils d’utilisateurs, notre système de recommandation doit faire face au problème du “cold-start”, c’est-à-dire que le système doit recommander des produits à un utilisateur pour lequel il n’a aucune information antérieure. De plus, les données récoltées durant l’utilisation du système sont des interactions implicites entre l’utilisateur et le site, il n’est donc pas possible de se baser sur des notes ou des retours explicites d’utilisateurs comme

c'est usuellement le cas pour les systèmes de recommandation. On note aussi que le système ne récolte pas d'informations validant l'achat d'un produit par l'utilisateur, puisque l'utilisateur est redirigé vers le site marchand s'il est intéressé par un produit, ce qui complexifie encore la recommandation.

Notre objectif est de démontrer la pertinence de l'approche multi-vues par rapport à une simple concaténation des vues pour notre projet. Notre Baseline utilisant l'approche précédente et est comparée à trois nouveaux modèles utilisant l'approche multi-vues. Les métriques utilisées pour évaluer les modèles sont les mêmes que pour les évaluations menées précédemment sur les vues : le HR et le NDCG. Chaque interaction entre l'utilisateur et le site concerne un produit, la variable cible que les modèles apprennent à recommander est le produit concerné par l'interaction suivante de l'utilisateur. La taille du classement à recommander pour les deux jeux de données est équilibrée en fonction du nombre de produits de chaque jeu de données.

4 Étude comparative

Le tableau 2 présentent les performances des modèles évalués sur les jeux de données D1 et D2. Les valeurs en gras correspondent aux valeurs les plus élevées pour chaque métrique. Chaque valeur est obtenue par la moyenne de 20 expériences pour lesquelles nous avons fait varier la graine de la descente de gradient stochastique. Chaque expérience est effectuée sur 50 epochs pour la Baseline et sur 10 epochs pour tous les autres modèles. En effet, les modèles multi-vues convergent plus rapidement que la Baseline, le temps d'entraînement de ces modèles est donc plus court.

Jeu	Métrique	Baseline	MV-DNN	TDSSM	MV-AFM
D1	HR@236	92.12 ± 0.13	93.45 ± 0.14	93.39 ± 0.16	93.47 ± 0.21
	HR@12	49.04 ± 0.40	51.36 ± 1.50	52.02 ± 1.02	51.11 ± 0.65
	NDCG@236	34.64 ± 0.21	35.12 ± 0.97	35.06 ± 0.76	34.55 ± 0.36
	NDCG@12	26.36 ± 0.26	26.97 ± 1.23	26.98 ± 0.96	26.29 ± 0.43
D2	HR@100	88.51 ± 0.15	91.79 ± 0.17	91.82 ± 0.12	91.91 ± 0.22
	HR@5	33.18 ± 0.28	40.13 ± 0.26	40.22 ± 0.38	40.33 ± 0.54
	NDCG@100	35.30 ± 0.17	39.84 ± 0.16	39.87 ± 0.17	39.57 ± 0.38
	NDCG@5	21.93 ± 0.22	27.09 ± 0.22	27.13 ± 0.31	26.78 ± 0.49

TAB. 2 – Évaluation des modèles avec les deux jeux de données.

De ces résultats nous pouvons observer que les trois modèles exploitant l'approche multi-vues obtiennent des résultats supérieurs à ceux de la Baseline. Cela montre que l'approche multi-vues pour ce projet est pertinente et efficace. Nous remarquons que le MV-AFM obtient des résultats légèrement supérieurs pour la métrique du Hit Rate et que le TDSSM obtient de meilleurs résultats pour le NDCG. Il pourra donc éventuellement être pertinent d'utiliser le MV-AFM si l'on souhaite maximiser les chances de recommander le produit le plus cohérent et d'utiliser le TDSSM si l'on vise à obtenir un ordre de recommandation plus pertinent. Toutefois l'écart entre les performances des trois modèles multi-vues étant très serré, il n'est pas possible d'affirmer que l'un de ces modèles soit supérieur aux autres en tout point.

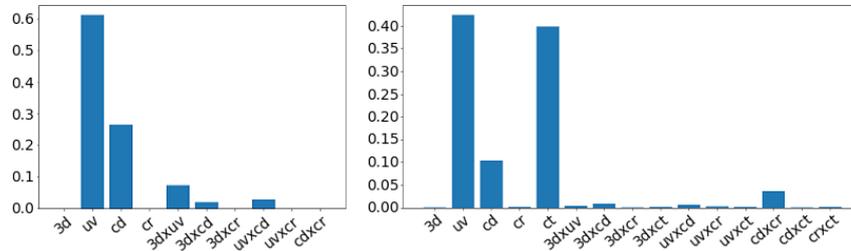


FIG. 2 – Niveaux d’attention moyens sur les jeux de données D1 (gauche) et D2 (droite).

La figure 2 présente les niveaux d’attention sur chaque vue et interaction de vues au sein du modèle MV-AFM en moyenne sur 1000 prédictions. Les ordonnées représentent le pourcentage d’attention portée et en abscisse figurent les vues et leurs interactions. On peut noter que, dans les deux cas, l’attention est majoritairement portée aux vecteurs latents des utilisateurs (uv), ce qui concorde avec l’évaluation des vues précédemment menée qui avait atteint les meilleurs résultats avec cette vue. Dans le cas de D1 les données expertes (cd) sont également largement exploitées et la vue temporelle des sessions des utilisateurs (3d) semble présenter un intérêt lorsqu’elle est corrélée avec les vecteurs des utilisateurs (3dxuv) et les données expertes (3dxcu). Dans le cas de D2 on retrouve un schéma similaire avec l’ajout de la vue de compatibilité (ct) qui a logiquement une importance capitale pour déterminer des recommandations cohérentes avec la requête de l’utilisateur.

5 Conclusion

Dans ce papier, nous présentons trois modèles exploitant l’approche d’apprentissage multi-vues appliquée à la recommandation dans le domaine du pneumatique. Nous présentons les résultats d’une étude comparative de ces trois modèles et de notre Baseline en application sur des données réelles. Nous observons que les modèles exploitant l’approche multi-vues obtiennent des résultats supérieurs à la Baseline, démontrant ainsi la pertinence de l’approche multi-vues dans le cadre de la recommandation. Le TDSSM montre les meilleurs résultats pour la métrique du NDCG, ce qui montre sa capacité à générer un classement de recommandation pertinent, tandis que le MV-AFM, utilisant des mécanismes d’attention, obtient les meilleurs résultats pour la métrique du HR montrant une plus haute précision pour recommander le produit cible. Les modèles comparés parviennent à obtenir une qualité de recommandation remarquable au vu des contraintes auxquelles le système doit faire face (problème du “cold-start”, interactions implicites et aucune confirmation d’achat).

Références

Adomavicius, G. et A. Tuzhilin (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.

- Betru, B. T., C. A. Onana, et B. Batchakui (2017). Deep Learning Methods on Recommender System: A Survey of State-of-the-art. *International Journal of Computer Applications* 162(10).
- Bourhis, K., K. Benabdeslem, et B. Canitia (2019). Entre factorisation de matrices et apprentissage profond pour la recommandation dans le domaine du pneumatique. In M.-C. Rousset et L. Boudjeloud-Assala (Eds.), *Extraction et Gestion des connaissances, EGC 2019, Metz, France, January 21-25, 2019*, Volume E-35 of *RNTI*, pp. 291–296. Éditions RNTI.
- Elkahky, A. M., Y. Song, et X. He (2015). A Multi-View Deep Learning Approach for Cross Domain User Modeling in Recommendation Systems. In *Proceedings of the 24th International Conference on World Wide Web - WWW '15*, Italy, pp. 278–288. ACM Press.
- He, X., H. Zhang, M.-Y. Kan, et T.-S. Chua (2016). Fast Matrix Factorization for Online Recommendation with Implicit Feedback. pp. 549–558. ACM Press.
- Kenaan, A., K. Benabdeslem, K. Bourhis, et B. Canitia (2020). Approches hybrides pour la recommandation dans le domaine du pneumatique. In A. Cornuéjols et E. Cuvelier (Eds.), *Extraction et Gestion des Connaissances, EGC 2020, Brussels, Belgium, January 27-31, 2020*, Volume E-36 of *RNTI*, pp. 133–144. Éditions RNTI.
- Liang, T., L. Zheng, L. Chen, Y. Wan, P. S. Yu, et J. Wu (2020). Multi-view factorization machines for mobile app recommendation based on hierarchical attention. *Knowledge-Based Systems* 187, 104821.
- Schein, A. I., A. Popescul, L. H. Ungar, et D. M. Pennock (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 - SIGIR '02*, Tampere, Finland, pp. 253. ACM Press.
- Song, Y., A. M. Elkahky, et X. He (2016). Multi-Rate Deep Learning for Temporal Recommendation. pp. 909–912. *SIGIR '16: Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*.
- Tuan, T. X. et T. M. Phuong (2017). 3D Convolutional Networks for Session-based Recommendation with Content Features. pp. 138–146. ACM Press.
- Xu, C., D. Tao, et C. Xu (2013). A Survey on Multi-view Learning. *arXiv:1304.5634*.
- Zhang, S., L. Yao, A. Sun, et Y. Tay (2019). Deep Learning based Recommender System: A Survey and New Perspectives. *ACM Computing Surveys* 52(1), 1–38. arXiv: 1707.07435.

Summary

We are constantly using recommender systems, often without even noticing. They build a profile of our person in order to recommend the content we will most likely be interested in. The data representing the users, their interactions with the system or the products may come from different sources and be of a various natures. Our goal is to use multi-view learning approaches to improve our recommender system and improve its capacity to manage multi-view data. We propose a comparative study between several state of the art multi-view models applied to our industrial data. Our study demonstrates the relevance of using multi-view learning within recommender systems.