Sparse Embeddings for Recommender Systems with Knowledge Graphs

Discussion Paper

Vito Walter Anelli[1], Tommaso Di Noia[1], Eugenio Di Sciascio[1], Antonio Ferrara[1] and Alberto Mancino[1]

[1]Politecnico di Bari, Bari, Italy

Abstract

Collaborative filtering models have undoubtedly dominated the scene of recommender systems. However, these methods do not take into account valuable item characteristics. On the other side, content-based algorithms only use this kind of information and may fail to generalize. Some collaborative filtering techniques have recently used side information about items, but they end up being huge models using thousands of features for modeling a single user-item interaction. In this paper, we present KGFlex, a sparse and expressive model based on feature embeddings. KGFlex studies which features are considered by each user when consuming an item. Then, it models each user-item interaction as a factorized entropy-driven combination of the only item features relevant to the user. An extensive experimental evaluation shows the approach's effectiveness, considering the recommendation results' accuracy, diversity, and induced bias.

Keywords

recommender systems, knowledge graphs, feature embeddings, feature factorization

1. Introduction

The outstanding accuracy of collaborative filtering techniques has undoubtedly helped recommender systems getting famous. However, these methods are based on the simple idea to recommend certain items since "similar users have experienced those items", or "other users, who have experienced the same items, have also experienced those items". On the contrary, content-based recommendation algorithms aim to recommend new items that share the same patterns of features of items liked in the past. The use of content features can make the model interpretable [1] but these techniques may fail to recommend items that have different characteristics with respect to the items enjoyed in the past. To get the benefits of the two approaches and mitigate their drawbacks, scientists worked to integrate into collaborative filtering the side information used in content-based approaches such as tags [2], demographic data [3], structured knowledge [4]. However, this may lead to very large models that need to take into account hundreds or thousands of features for predicting user-item interactions.

In this work, we introduce KGFlex, a knowledge-aware recommendation system, that tackles this issue with a sparse and expressive model based on feature embeddings. KGFlex describes the catalog using features extracted from publicly available knowledge graphs, one of the most impactful and relevant sources for knowledge-aware recommender systems. Then, low-dimensionality

11th Italian Information Retrieval Workshop (IIR 2021), September 13–15, 2021, Bari, Italy
antonio.ferrara@poliba.it (A. Ferrara)

embeddings are adopted to represent the semantic item features. Using an entropy-based strategy, KGFlex analyzes the users' history to study the user-specific decision-making process of consuming or not consuming an item. Thus, the subsets of item features relevant to the user in her decision-making process are adopted to model the user-item interaction.

To evaluate the performance of KGFlex, we conduct extensive experiments on two different publicly available datasets. We evaluate the accuracy and diversity of recommendation results and analyze whether the algorithm produces biased recommendations. The results show that KGFlex has competitive accuracy performance and, at the same time, generates highly diversified recommendations with a low induced bias.

2. Basics of KGFlex

KGFlex exploits the knowledge encoded in a knowledge graph as side information to characterize both items and users. One of the main assumptions is that users decide to enjoy an item based on a subset of its characteristics, implying that not all the item features are equally important. In the following, we show how KGFlex describes each user and item with a set of features. Taking a cue from information theory KGFlex exploits the notion of information gain to measure the relevance of a feature for a user in deciding to consume or not an item.

From $\mathcal{KG}s$ to Decision-Making. A knowledge graph $\mathcal{KG}$ can be represented as a set of triples where entities are linked to each other by binary relations. Each connection in $\mathcal{KG}$ is then a triple $\sigma \xrightarrow{\rho} \omega$, where σ is a subject entity, ρ is a relation (predicate), and ω is an object entity. If we consider chains of predicates that connect two entities at a higher depth, a n-hop predicate can be defined as $\rho = \langle \rho_1, ..., \rho_n \rangle$ if $\sigma \xrightarrow{\rho_1} \omega_1 \xrightarrow{\rho_2} ... \xrightarrow{\rho_n} \omega_n \in \mathcal{KG}$. For convenience, $h(\rho) = n$ for $\rho : \sigma \xrightarrow{\rho} \omega_n \in \mathcal{KG}$ denotes the depth of the predicate chain. When no confusion arises, from now on we will use $\sigma \xrightarrow{\rho} \omega$ to denote a generic chain with $h(\rho) \in \{1, ..., n\}$.

Given a collection of items $\mathcal{I}$ and a knowledge graph $\mathcal{KG}$, we assume each element in $i \in \mathcal{I}$ has a mapping to a corresponding entity in $\mathcal{KG}$. Under this assumption, an item i can be explored, at depth n, to identify the set $\mathcal{F}_i^{(n)}$ of the semantic features describing it:

$$\mathcal{F}_i^{(n)} = \{ \langle \rho, \omega \rangle \mid i \xrightarrow{\rho} \omega \in \mathcal{KG}, h(\rho) \in \{1, ..., n\} \}. \tag{1}$$

We describe each user $u \in \mathcal{U}$ with the set $\mathcal{F}_u = \bigcup_{i \in \mathcal{I}_u} \mathcal{F}_i^{(n)}$, i.e. all the features representing the items $\mathcal{I}_u \subseteq \mathcal{I}$ enjoyed by u. Finally, we define the overall set of features in the system as $\mathcal{F}^{(n)} = \bigcup_{i \in \mathcal{I}} \mathcal{F}_i^{(n)}$ of features in the system: In the following, the (n) superscript is omitted whenever it is not relevant in the context.

Once items and users have been associated with their set of features, we use the notion of information gain to measure the importance of each feature for a user in deciding whether to consume or not consume an item, i.e. in distinguishing positive from negative items in the dataset. Indeed, given a dataset $\mathcal{D}$ with a certain extent of entropy (uncertainty) on the target attribute, the information gain $IG(\mathcal{D}, x_d)$ measures the expected reduction in information entropy obtained from the observation of the value of the d-th attribute of a sample $\mathbf{x}$. To this aim, we build, for each user u, a balanced dataset $\mathcal{D}_u$ with all the consumed items from $\mathcal{I}_u$ and the same amount of negative items randomly picked up from $\bigcup_{v \in \mathcal{U}, v \neq u} \mathcal{I}_v \setminus \mathcal{I}_u$. For each of these positive and negative items, $\mathcal{D}_u$ is provided with a sample whose attributes correspond to the features in $\mathcal{F}_u$ and indicate the presence ($f = 1$) or the absence ($f = 0$) of the corresponding feature f in $\mathcal{F}_i$. Therefore, the attribute f provides an information gain in distinguishing positive from negative samples equal to $IG(\mathcal{D}_u, f) = 1 - H(\mathcal{D}_u | f = 1) - H(\mathcal{D}_u | f = 0)$.

We finally associate a weight $k_{uf} = IG(\mathcal{D}_u, f)$ to each pair of user u and feature f to represent the influence of a feature —in the view of the user— in the prediction of user-item interactions.

Sparse Embeddings. KGFlex models the features in $\mathcal{F}$ as collaboratively learned embeddings in a latent space. Since KGFlex promotes the idea of having user fine-tuned versions of the same model, we have both a global representation of the features in $\mathcal{F}$ and a personal view, for each user u, of the features in $\mathcal{F}_u \subseteq \mathcal{F}$. Notably, the model is structured into two distinct parts. On the one hand, KGFlex keeps a set $\mathcal{G}$ of global trainable embeddings and biases shared among all the users, with $\mathcal{G} = \{(\mathbf{g}_f \in \mathbb{R}^E, b_f \in \mathbb{R}), \forall f \in \mathcal{F}\}$. On the other hand, each user in KGFlex also has his/her personal representation of the features he/she interacted with, i.e., the features in $\mathcal{F}_u$. These embeddings are collected within the set $\mathcal{P}^u$, defined as $\mathcal{P}^u = \{\mathbf{p}_f^u \in \mathbb{R}^E, \forall f \in \mathcal{F}_u\}$. Then, the inner product between the personal representation $\mathbf{p}_u^f$ and the global representation $\mathbf{g}_f$, plus a bias value b_f, estimates the affinity of user u to feature f. The sum of such affinities for all the features in $\mathcal{F}_{ui} = \mathcal{F}_u \cap \mathcal{F}_i$, weighted according to the pre-computed entropy-based coefficients, estimates the interaction $\hat{x}_{ui}$ between user u and item i:

$$\hat{x}_{ui} = \sum_{f \in \mathcal{F}_{ui}} k_{uf} (\mathbf{p}_f^u \mathbf{g}_f + b_f). \tag{2}$$

Eq. (2) encodes the strategy KGFlex exploits to handle the features: it takes advantage of user profile to involve only a small subset of them in the estimate of the user-item affinity.

To learn the model parameters, KGFlex adopts Bayesian Personalized Ranking (BPR), the most common pair-wise Learning to Rank strategy, that, given a training set $\mathcal{T} = \{(u, i^+, i^-) \mid i^+ \in \mathcal{I}_u \wedge i^- \in \mathcal{I} \setminus \mathcal{I}_u, \forall u \in \mathcal{U}\}$, optimizes the loss $L = \sum_{(u,i^+,i^-) \in \mathcal{T}} \ln \sigma(\hat{x}_{ui^+} - \hat{x}_{ui^-})$, with the assumption that a user u prefers a consumed item i^+ over a non-consumed item i^-.

3. Exploratory Evaluation

Experimental Setting. The evaluation of the performance of KGFlex is conducted on two well-known datasets: *Yahoo! Movies* and *Facebook Books.* The datasets have been binarized, retaining ratings of 3 or higher, and have been preprocessed with iterative 10-core and 5-core, respectively. The semantic features have been retrieved through a 2-depth exploration of the DBpedia $\mathcal{KG}$, removing some useless features [5]. Finally, we removed the features associated with less than ten items, and we kept the user's 100 most informative features from the 1- and 2- hop exploration.

We compare KGFlex with BPR-MF [6], a latent factor model based on the same pair-wise optimization criterion used in KGFlex, a batch version of Rendle et al. [7] MF, NeuMF [8], and kaHFM [4], a factorization-based model making use of knowledge graphs. For the sake of reproducibility, we provide our code and all the details about the experiments[1].

[1]https://split.to/kgflex

Table 1
Comparison of KGFlex with baselines. The best result is in boldface, the second-best result is underlined. For all the metrics, the cutoff is 10.

	Yahoo! Movies						Facebook Books					
	nDCG	**IC**	**Gini**	**ACLT**	**PopREO**	**PopRSP**	**nDCG**	**IC**	**Gini**	**ACLT**	**PopREO**	**PopRSP**
BPR-MF	0.1857	151	0.0219	0.0006	0.9954	0.9999	0.0947	17	0.0132	0.0000	1.0000	1.0000
MF	<u>0.2897</u>	455	0.0902	0.0823	0.8735	0.9865	<u>0.0956</u>	87	0.0238	0.0000	1.0000	1.0000
NeuMF	0.0918	50	0.0113	0.0006	1.0000	0.9999	0.0714	17	0.0125	0.0000	1.0000	1.0000
kaHFM	**0.3006**	<u>757</u>	<u>0.1659</u>	<u>0.4624</u>	<u>0.7610</u>	<u>0.9234</u>	**0.1267**	<u>540</u>	<u>0.1387</u>	<u>0.3294</u>	<u>0.8766</u>	<u>0.9420</u>
KGFlex	0.2464	**851**	**0.2802**	**2.1447**	**0.4477**	**0.6336**	0.0853	**606**	**0.3070**	**3.0264**	**0.1521**	**0.4485**

We have measured the recommendation accuracy with nDCG [9]. We have also evaluated the diversity, adopting Item Coverage (IC) [10] and Gini Index (Gini) [11]. Finally, three bias metrics have been used to evaluate how the algorithms consider the items from the long-tail: ACLT [12], PopREO and PopRSP, specific applications of RSP, and REO [13]. PopREO estimates the equal opportunity of items, encouraging the True Positive Rate of popular and unpopular items to be the same. PopRSP measures statistical parity, assessing whether the ranking probability distributions for popular and unpopular items are the same in the recommendation.

Main Results. Table 1 depicts the evaluation outcome for the aforementioned metrics with a cut-off of 10. For Yahoo! Movies, KGFlex is outperformed exclusively by kaHFM and MF, but continues to show acceptable accuracy results. It is noteworthy that KGFlex significantly outperforms BPR-MF, albeit both are learned with a pair-wise BPR optimization, hence underlining the beneficial role of the extracted knowledge. Moreover, examining the item coverage and Gini values, we note the high degree of personalization provided by KGFlex. We link this result to the personalized view of the knowledge granted by the framework. Moreover, in KGFlex the collaborative signal on explicit user interests ensures to recommend diverse items among the ones sharing characteristics of interest for the user. The aforementioned behavior is not confirmed in Facebook Books. Indeed, the accuracy results seem to remain below the performance of other approaches. However, the diversity results show how BPR-MF, MF, and NeuMF may have been flooded by popularity signal, which led them to perform poorly regarding the item coverage and Gini metrics. Instead, KGFlex does not suffer from this problem and approaches the superior performance of Item-kNN in terms of diversity.

Oftentimes, recommender systems fail to recommend unpopular items, which tend to remain underrepresented [14], thus causing a fairness issue for items and an inappropriate recommendation for users who do not prefer very popular items. From Table 1, it is noteworthy that KGFlex always outperforms all the other factorization-based approaches and generally outperforms the other approaches. The value of ACLT (the higher the better) is comparable with the value obtained by VSM. This result is further supported by the values of PopREO and PopRSP (the smaller the better). Concerning those metrics, KGFlex and VSM continue to grant the less biased recommendations. Interestingly, while both exploit the same optimization criterion, we notice how KGFlex consistently improves BPR-MF, which is known to be vulnerable to imbalanced data and to produce biased recommendations [13].

References

[1] Y. Zhang, X. Chen, Explainable recommendation: A survey and new perspectives, CoRR abs/1804.11192 (2018). URL: http://arxiv.org/abs/1804.11192. arXiv:1804.11192.

[2] Y. Zhu, Z. Guan, S. Tan, H. Liu, D. Cai, X. He, Heterogeneous hypergraph embedding for document recommendation, Neurocomputing 216 (2016) 150–162. URL: https://doi.org/10.1016/j.neucom.2016.07.030. doi:10.1016/j.neucom.2016.07.030.

[3] W. X. Zhao, S. Li, Y. He, L. Wang, J. Wen, X. Li, Exploring demographic information in social media for product recommendation, Knowl. Inf. Syst. 49 (2016) 61–89. URL: https://doi.org/10.1007/s10115-015-0897-5. doi:10.1007/s10115-015-0897-5.

[4] V. W. Anelli, T. D. Noia, E. D. Sciascio, A. Ragone, J. Trotta, How to make latent factors interpretable by feeding factorization machines with knowledge graphs, in: C. Ghidini, O. Hartig, M. Maleshkova, V. Svátek, I. F. Cruz, A. Hogan, J. Song, M. Lefrançois, F. Gandon (Eds.), The Semantic Web - ISWC 2019 - 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part I, volume 11778 of *Lecture Notes in Computer Science*, Springer, 2019, pp. 38–56. URL: https://doi.org/10.1007/978-3-030-30793-6_3. doi:10.1007/978-3-030-30793-6_3.

[5] T. Di Noia, C. Magarelli, A. Maurino, M. Palmonari, A. Rula, Using ontology-based data summarization to develop semantics-aware recommender systems, in: A. Gangemi, R. Navigli, M. Vidal, P. Hitzler, R. Troncy, L. Hollink, A. Tordai, M. Alam (Eds.), The Semantic Web - 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3-7, 2018, Proceedings, volume 10843 of *Lecture Notes in Computer Science*, Springer, 2018, pp. 128–144. URL: https://doi.org/10.1007/978-3-319-93417-4_9. doi:10.1007/978-3-319-93417-4_9.

[6] S. Rendle, C. Freudenthaler, Z. Gantner, L. Schmidt-Thieme, BPR: bayesian personalized ranking from implicit feedback, in: J. A. Bilmes, A. Y. Ng (Eds.), UAI 2009, Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, June 18-21, 2009, AUAI Press, 2009, pp. 452–461. URL: https://dslpitt.org/uai/displayArticleDetails.jsp?mmnu=1&smnu=2&article_id=1630&proceeding_id=25.

[7] S. Rendle, W. Krichene, L. Zhang, J. R. Anderson, Neural collaborative filtering vs. matrix factorization revisited, in: R. L. T. Santos, L. B. Marinho, E. M. Daly, L. Chen, K. Falk, N. Koenigstein, E. S. de Moura (Eds.), RecSys 2020: Fourteenth ACM Conference on Recommender Systems, Virtual Event, Brazil, September 22-26, 2020, ACM, 2020, pp. 240–248. URL: https://doi.org/10.1145/3383313.3412488. doi:10.1145/3383313.3412488.

[8] X. He, T. Chua, Neural factorization machines for sparse predictive analytics, in: N. Kando, T. Sakai, H. Joho, H. Li, A. P. de Vries, R. W. White (Eds.), Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017, ACM, 2017, pp. 355–364. URL: https://doi.org/10.1145/3077136.3080777. doi:10.1145/3077136.3080777.

[9] W. Krichene, S. Rendle, On sampled metrics for item recommendation, in: R. Gupta, Y. Liu, J. Tang, B. A. Prakash (Eds.), KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020, ACM, 2020, pp. 1748–1757. URL: https://doi.org/10.1145/3394486.3403226. doi:10.1145/3394486.3403226.

[10] G. Adomavicius, Y. Kwon, Improving aggregate recommendation diversity using ranking-based techniques, IEEE Trans. Knowl. Data Eng. 24 (2012) 896–911. URL:

https://doi.org/10.1109/TKDE.2011.15. doi:10.1109/TKDE.2011.15.
[11] P. Castells, N. J. Hurley, S. Vargas, Novelty and diversity in recommender systems, in: F. Ricci, L. Rokach, B. Shapira (Eds.), Recommender Systems Handbook, Springer, 2015, pp. 881–918. URL: https://doi.org/10.1007/978-1-4899-7637-6_26. doi:10.1007/978-1-4899-7637-6_26.
[12] H. Abdollahpouri, R. Burke, B. Mobasher, Managing popularity bias in recommender systems with personalized re-ranking, in: R. Barták, K. W. Brawner (Eds.), Proceedings of the Thirty-Second International Florida Artificial Intelligence Research Society Conference, Sarasota, Florida, USA, May 19-22 2019, AAAI Press, 2019, pp. 413–418. URL: https://aaai.org/ocs/index.php/FLAIRS/FLAIRS19/paper/view/18199.
[13] Z. Zhu, J. Wang, J. Caverlee, Measuring and mitigating item under-recommendation bias in personalized ranking systems, in: J. Huang, Y. Chang, X. Cheng, J. Kamps, V. Murdock, J. Wen, Y. Liu (Eds.), Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020, ACM, 2020, pp. 449–458. URL: https://doi.org/10.1145/3397271.3401177. doi:10.1145/3397271.3401177.
[14] H. Abdollahpouri, M. Mansoury, R. Burke, B. Mobasher, The unfairness of popularity bias in recommendation, in: R. Burke, H. Abdollahpouri, E. C. Malthouse, K. P. Thai, Y. Zhang (Eds.), Proceedings of the Workshop on Recommendation in Multi-stakeholder Environments co-located with the 13th ACM Conference on Recommender Systems (RecSys 2019), Copenhagen, Denmark, September 20, 2019, volume 2440 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2019. URL: http://ceur-ws.org/Vol-2440/paper4.pdf.