



Original software publication

XRecSys: A framework for path reasoning quality in explainable recommendation

Giacomo Balloccu^{*}, Ludovico Boratto, Gianni Fenu, Mirko Marras

University of Cagliari, Italy

ARTICLE INFO

Keywords:

Recommender systems
 Explainability
 Knowledge graphs
 Evaluation
 Reasoning

ABSTRACT

There is increasing evidence that recommendations accompanied by explanations positively impact on businesses in terms of trust, guidance, and persuasion. This advance has been made possible by traditional models representing user–product interactions augmented with external knowledge modeled as knowledge graphs. However, these models produce textual explanations on top of reasoning paths extracted from the knowledge graph without considering relevant properties of the path entities. In this paper, we present XRecSys, a Python framework for the optimization of the reasoning path selection process according to properties deemed relevant by users (e.g., time relevance of the linking interaction or popularity of the entity linked to the explanation). Our framework leads to a higher reasoning path quality in terms of the considered properties and, consequently, textual explanations more relevant for the users.

Code metadata

Current code version
 Permanent link to repository used for this code version
 Permanent link to reproducible capsule
 Legal code license
 Code versioning system used
 Software code languages, tools, and services used
 Compilation requirements, operating environments, dependencies
 If available link to developer documentation/manual
 Support email for questions

0.1
<https://github.com/SoftwareImpacts/SIMPAC-2022-156>
<https://codeocean.com/capsule/2212769/tree/v1>
 MIT License
 Git, Github
 Python
 requirements.txt
<https://github.com/giacoballoccu/explanation-quality-recsys>
giacomo.balloccu@acm.org

1. Introduction

Explaining to users *why* certain results have been provided to them has become an essential feature of modern information retrieval systems. In the deep learning era, the underlying models tend to be complex and opaque and explanations can improve their understanding and transparency. Under certain conditions, increasing users' awareness on the system's outputs (e.g., via explanations) is also mandatory by law (e.g., European General Data Protection Regulation and California on Automated Decision Systems) [1]. Recommender systems (RSs)

represent a notable class of information retrieval systems whose outputs require accompanying explanations.

Existing explainable recommender systems augment traditional models representing user–product interactions with external knowledge modeled as knowledge graphs (KGs). On one hand, *post-hoc explainable models* [2] consider recommendation and explanation generation as two distinct steps. Methods pertaining to this family either pre-compute paths in the knowledge graph and attach them to the recommended products generated by traditional models [3,4] or generate counterfactual explanations [5–8]. In both cases, the ranked products are

The code (and data) in this article has been certified as Reproducible by Code Ocean: (<https://codeocean.com/>). More information on the Reproducibility Badge Initiative is available at <https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals>.

^{*} Corresponding author.

E-mail addresses: giacomo.balloccu@acm.org (G. Balloccu), ludovico.boratto@acm.org (L. Boratto), fenu@unica.it (G. Fenu), mirko.marras@acm.org (M. Marras).

<https://doi.org/10.1016/j.simpa.2022.100404>

Received 26 July 2022; Accepted 1 August 2022

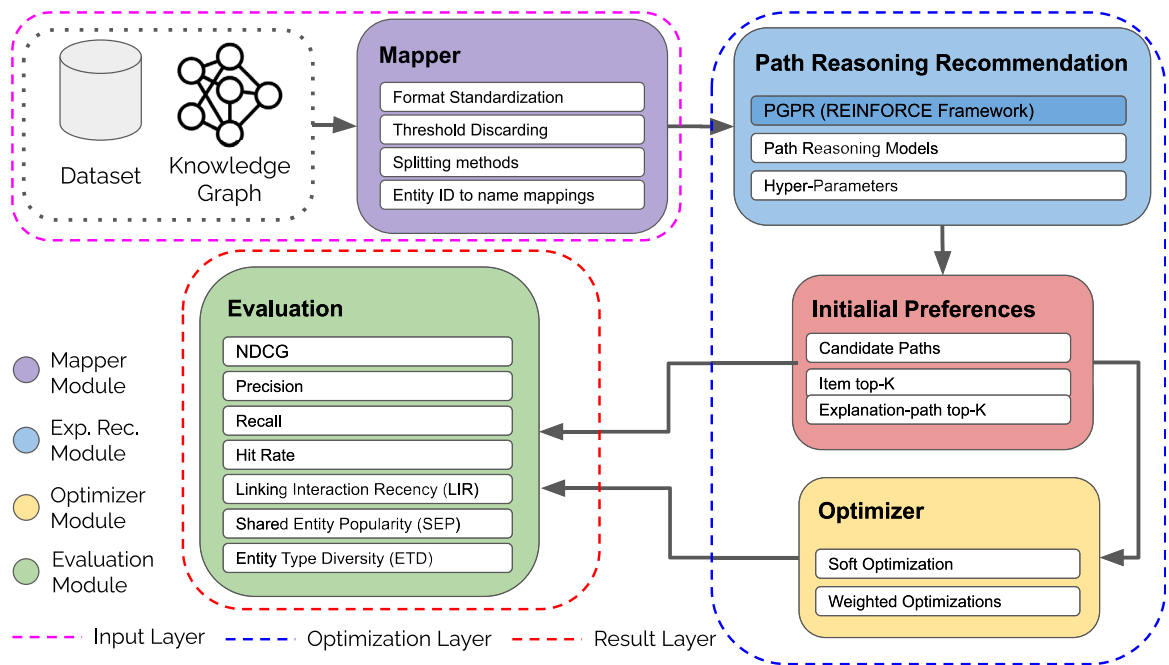


Fig. 1. The Mapper Module pre-processes and standardizes a raw data set and the associated knowledge graph completion. The Exp. Rec. module is responsible of training a recommendation model and generating recommended lists and candidate reasoning paths. The Optimizer allows to perform the re-ranking of the already generated recommendations and candidate reasoning paths according to a certain property of the latter.

optimized for utility and the training process of the recommendation model does not embed any constraint for the selection of accompanying reasoning paths. On the other hand, *self-explainable* [9] models perform recommendation and explanation generation jointly. Representative methods of this family include those performing reasoning either on the paths within the KG [10–13] or via neural symbolic techniques [14,15]. Though the resulting explanation is aligned to the associated recommendation, the utility of the recommendations could decrease and the explanations could not be aligned to the expectation of the users.

Usually, a reasoning path between a product the user already interacted with ($product_1$) and a product recommended to that user ($product_2$) is shaped in the form of *user interacted $product_1$ share $entity_1$ share $product_2$* . This path can be used to provide the textual explanation “ $product_2$ is recommended to you because it shares $entity_1$ with another product you interacted with”. These paths basically include a past linking interaction (*user interacted $product_1$*), a shared entity ($entity_1$), a sharing relationship (*shared*), and a recommend item ($product_2$). We believe that the conceptual parts of a path have key characteristics that can significantly influence the perceived explanation quality for the users. For instance, the recency of the linking interaction presented can help the user better remember the interaction that brings to the recommendation. Similarly, a popular entity has a higher likelihood of being already known by the user and, possibly, lead to a better understanding of the provided explanation. Furthermore, the diversity in terms of explanation path types accompanying a recommended list can increase users’ satisfaction and help users learn more connections.

In this paper, we present *XRecSys*, a Python framework for the optimization of the reasoning path selection process based on properties deemed relevant by users. Researchers and practitioners can optimize the suggested lists of products and the candidate reasoning paths generated for each product by a given recommendation model, according to properties of the reasoning paths. *XRecSys* is built on top of our paper recently published in the proceedings of the ACM SIGIR ’22 conference [16]. There, we proposed three novel user-level properties of the reasoning paths emerged from a mixed approach involving a literature review study and an online users’ survey, and

a suite of re-ranking approaches for their optimization. Case studies for our framework include, among others, the re-ranking of recommendations with explanations provided by the state-of-the-art model PGPR [11]. Being flexible, we expect that *XRecSys* can be generally used for optimizing and testing any reasoning path property in different domains and with other models. In addition, our framework can be used to study how explanations whose underlying reasoning paths are optimized for certain properties affect users.

2. Functionality and key features

The proposed framework, illustrated in Fig. 1, produces textual explanations optimized for the proposed properties (e.g., the time relevance of the linking interaction in a reasoning path). In what follows, we describe the three main layers of the framework, namely Input Layer, Optimization Layer, and Result Layer.

2.1. Input layer

The Input layer includes two modules: KG-Dataset and Mapper. The KG-Dataset is composed by two ready to use pairs data sets and associated Knowledge Graphs. The dataset can be selected and pre-processed using the Mapper module, where is possible to perform threshold discarding, selecting different splitting methods and map the data in a standardize format that can be leveraged by the Exp. Rec. Module.

2.2. Optimization layer

The Optimization Layer is composed by the Explainable Recommender and the Optimizer Module. The Explainable Recommender is represented by a representative state-of-the-art baseline, in our case PGPR [11], that produces as output the whole set of candidate paths and the original preferences represented by pairs of recommended items and associated explanations. This output can be assessed directly thorough the Result Layer or can be redirected as input for the Optimizer module that can achieve two type of optimizations based on

which output from the previous module is leveraged. The optimizations are distinguished in i) soft optimization, which performs a re-ranking of only the explanation paths, so the top- k recommended products are the same as the original model, while the correspondent explanation paths are selected by picking the path that maximizes the target metric(s), and (ii) weighted optimization, which takes as input the whole set of predicted paths and perform a re-ranking of both products and explanations.

2.3. Result layer

The Results layer can take as input either the original or the optimized top- k and corresponding explanation paths and allows for the assessment of average path explanation quality metrics and recommendation utility metrics, such as NDCG, Precision, Recall, and Hit Rate.

3. Software impact

Recommender systems based on path reasoning or neural symbolic reasoning are receiving increasing attention thanks to their capability of providing high fidelity explanations to users. Recent work has investigated the extent to which the characteristics of the reasoning paths selected for generating the textual explanations affect the quality of the recommendations perceived by the users. In this paper, we proposed a novel Python framework, *XRecSys*, with the purpose of promoting further studies touching on this line of research. For instance, we expect that our framework can ease the operationalization of new properties of reasoning paths as well as the assessment of the existing ones. Our codebase, is accompanied by a range of properties measurable on reasoning paths and optimizable via a suite of re-ranking approaches, according to a recently published paper. As a future direction, we plan to extend the range of available properties, experimental data sets, and recommendation models. Also we plan to incorporate the fairness dimension [17–19] to these methods to assess on what extend the recommendation produced are fair. The long-term goal of this framework is to support the community in improving the quality of textual explanations accompanying recommendations and foster the use of such strategies for academic and industrial purposes.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Giacomo Balloccu gratefully acknowledges the Department of Mathematics and Computer Science of the University of Cagliari for the financial support of his PhD scholarship under the “Analysis and development of methodologies in recommender systems” project (FONDI RICDIP RECSYS 2019 FENU).

References

[1] B. Goodman, S.R. Flaxman, European union regulations on algorithmic decision-making and a “Right to explanation”, *AI Mag.* 38 (3) (2017) 50–57, <http://dx.doi.org/10.1609/aimag.v38i3.2741>.
 [2] S. Xu, Y. Li, S. Liu, Z. Fu, X. Chen, Y. Zhang, Learning post-hoc causal explanations for recommendation, 2020, arXiv preprint [arXiv:2006.16977](https://arxiv.org/abs/2006.16977).

[3] Y. Cao, X. Wang, X. He, Z. Hu, T.-S. Chua, Unifying knowledge graph learning and recommendation: Towards a better understanding of user preferences, in: *The World Wide Web Conference, WWW '19*, Association for Computing Machinery, New York, NY, USA, 2019, pp. 151–161, <http://dx.doi.org/10.1145/3308558.3313705>.
 [4] Q. Ai, V. Azizi, X. Chen, Y. Zhang, Learning heterogeneous knowledge base embeddings for explainable recommendation, 2018, CoRR abs/1805.03352 [arXiv:1805.03352](https://arxiv.org/abs/1805.03352).
 [5] S. Verma, J. Dickerson, K. Hines, Counterfactual explanations for machine learning: A review, 2020, arXiv preprint [arXiv:2010.10596](https://arxiv.org/abs/2010.10596).
 [6] K.H. Tran, A. Ghazimatin, R. Saha Roy, Counterfactual explanations for neural recommenders, in: *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 1627–1631.
 [7] V. Kaffes, D. Sacharidis, G. Giannopoulos, Model-agnostic counterfactual explanations of recommendations, in: *Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization*, 2021, pp. 280–285.
 [8] Y. Ge, J. Tan, Y. Zhu, Y. Xia, J. Luo, S. Liu, Z. Fu, S. Geng, Z. Li, Y. Zhang, Explainable fairness in recommendation, 2022, arXiv preprint [arXiv:2204.11159](https://arxiv.org/abs/2204.11159).
 [9] A. Rana, D. Bridge, Explanations that are intrinsic to recommendations, in: *Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization, UMAP '18*, Association for Computing Machinery, New York, NY, USA, 2018, pp. 187–195, <http://dx.doi.org/10.1145/3209219.3209230>.
 [10] K. Zhao, X. Wang, Y. Zhang, L. Zhao, Z. Liu, C. Xing, X. Xie, Leveraging demonstrations for reinforcement recommendation reasoning over knowledge graphs, in: *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Association for Computing Machinery, New York, NY, USA, 2020*, pp. 239–248.
 [11] Y. Xian, Z. Fu, S. Muthukrishnan, G. de Melo, Y. Zhang, Reinforcement knowledge graph reasoning for explainable recommendation, in: *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'19*, Association for Computing Machinery, New York, NY, USA, 2019, pp. 285–294, <http://dx.doi.org/10.1145/3331184.3331203>.
 [12] X. Wang, K. Liu, D. Wang, L. Wu, Y. Fu, X. Xie, Multi-level recommendation reasoning over knowledge graphs with reinforcement learning, in: *Proceedings of the ACM Web Conference 2022, WWW '22*, Association for Computing Machinery, New York, NY, USA, 2022, pp. 2098–2108, <http://dx.doi.org/10.1145/3485447.3512083>.
 [13] D. Liu, J. Lian, Z. Liu, X. Wang, G. Sun, X. Xie, Reinforced anchor knowledge graph generation for news recommendation reasoning, in: *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, KDD '21*, Association for Computing Machinery, New York, NY, USA, 2021, pp. 1055–1065, <http://dx.doi.org/10.1145/3447548.3467315>.
 [14] N. Diaz-Rodriguez, A. Lamas, J. Sanchez, G. Franchi, I. Donadello, S. Tabik, D. Filliat, P. Cruz, R. Montes, F. Herrera, Explainable neural-symbolic learning (X-NeSyL) methodology to fuse deep learning representations with expert knowledge graphs: The MonuMAI cultural heritage use case, *Inf. Fusion* 79 (2022) 58–83.
 [15] Y. Xian, Z. Fu, H. Zhao, Y. Ge, X. Chen, Q. Huang, S. Geng, Z. Qin, G. de Melo, S. Muthukrishnan, Y. Zhang, CAFE: Coarse-to-fine neural symbolic reasoning for explainable recommendation, in: *Proceedings of the 29th ACM International Conference on Information & Knowledge Management, CIKM '20*, Association for Computing Machinery, New York, NY, USA, 2020, pp. 1645–1654, <http://dx.doi.org/10.1145/3340531.3412038>.
 [16] G. Balloccu, L. Boratto, G. Fenu, M. Marras, Post processing recommender systems with knowledge graphs for recency, popularity, and diversity of explanations, in: *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '22*, Association for Computing Machinery, New York, NY, USA, 2022, pp. 646–656, <http://dx.doi.org/10.1145/3477495.3532041>.
 [17] M. Kaminskas, D. Bridge, Diversity, serendipity, novelty, and coverage: A survey and empirical analysis of beyond-accuracy objectives in recommender systems, *ACM Trans. Interact. Intell. Syst.* 7 (1) (2017) 2:1–2:42, <http://dx.doi.org/10.1145/2926720>.
 [18] L. Boratto, G. Fenu, M. Marras, Interplay between upsampling and regularization for provider fairness in recommender systems, *User Model. User Adapt. Interact.* 31 (3) (2021) 421–455, <http://dx.doi.org/10.1007/s11257-021-09294-8>.
 [19] L. Boratto, G. Fenu, M. Marras, G. Medda, Consumer fairness in recommender systems: Contextualizing definitions and mitigations, in: M. Hagen, S. Verberne, C. Macdonald, C. Seifert, K. Balog, K. Norvåg, V. Setty (Eds.), *Advances in Information Retrieval - 44th European Conference on IR Research, ECIR 2022*, Stavanger, Norway, April 10–14, 2022, Proceedings, Part I, in: *Lecture Notes in Computer Science*, vol. 13185, Springer, 2022, pp. 552–566, http://dx.doi.org/10.1007/978-3-030-99736-6_37.