A Comprehensive Review of Knowledge Graph–Based Explainable Recommender Systems

Sonu G K[1], Dr Jithendra P R Nayak[2]
[1]Reseacrh Scholar, Srinivas University, mukka Mangalore
[2]Associate Professor, Department of Computer Science and Engineering Srinivas Institute of Technology Valachil, Mangalore, Srinivas University, mukka Mangalore

Abstract - ***In the big data era recommender systems are necessary because giving users relevant and tailored information across a variety of domains. Conventional recommendation techniques including content-based filtering collaborative filtering and hybrid models face difficulties with data sparsity cold start and interpretability. It has been demonstrated that comprehension graph-based recommender systems which use graph structures to incorporate strong semantic links between users items and features are an effective form of rehabilitation. By depicting entities and relationships as triples knowledge graphs increase the precision variety and relevance of recommendations. In order to learn higher-order relationships and adaptively aggregate neighbourhood information recent research combines Neural networks in graphs and graph attention networks. Knowledge graph-based recommendation models are categorized into embedded-based path-based and propagation-based models in this review study which also discusses their benefits and explain ability characteristics. Furthermore, in order to offer further thorough and reliable recommendations a unified framework incorporating item user and feature mediated explanation is emphasized. Scalable learning dynamic knowledge graphs standardized evaluation of explanations human-in-the-loop systems and fairness-aware recommender models are among the open research issues and feature research directions that are finally discussed.***

Keywords: Knowledge graphs, collaborative filtering, content-based filtering, hybrid recommendation and recommender systems.

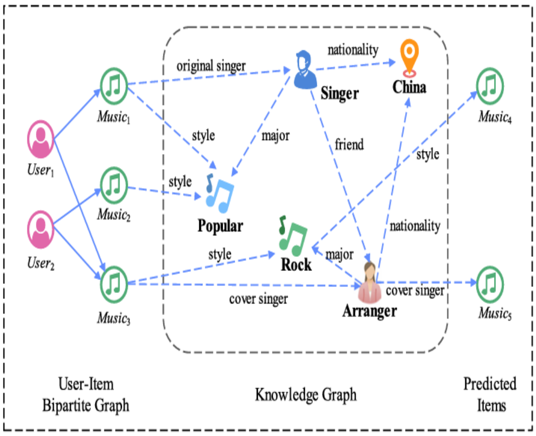
I. INTRODUCTION

In the big data era recommendation engines are crucial for supplying consumers with fast dependable customized information and item recommendations. While hybrid recommendations have issues like data sparsity cold start and poor interpretability, they can partially satisfy user needs. Conventional recommendation techniques include collaborative content-based filtering [5]. To ensure that build a massive knowledge network Google launched the graph of knowledge project in 2012 by combining a lot of online data [1]. This greatly improved the users personalized recommendation experience while also reducing data sparsity and issues with cold starts [3]. Researches have recently concentrated more on techniques that integrate knowledge graphs with graph attention networks. Knowledge graphs a structured information representation in the shape of triples can successfully handle problems like data sparsity by adding rich entity relationships and qualities through external knowledge increasing the accuracy and diversity of recommendation outcomes [6]. By adaptively assigning weights the graph attention method has enabled data aggregation resulting in more accurate recommendation results [4].

Knowledge graph-based recommendations incorporate information about user and item interactions in addition to item knowledge graph information. Knowledge graphs contain rich entity relationship data about items enabling the development of comprehensive item features and helping to uncover hidden connections between things which improves recommendation accuracy. To help with understanding we use collaborative filtering (CF) to show the interaction graph between users and items and knowledge graphs (KG) to show the connectivity graph between items [2]. Knowledge graphs have become a potent tool for improving explain ability in recommender systems in this context. Knowledge graphs use nodes to represent entities like user's objects and features and edges to capture semantic relationships. Recommender systems are able to reason over explicit connections such as shared attributes contextual similarities or past interactions thanks to this structured and relational representation. Complex relationships can be encoded into low-dimensional representation while maintaining semantic meaning by utilizing knowledge graph embeddings. The interpretability and contextual awareness of recommendations can therefore be enhanced by providing a meaningful explanation that traces paths through the knowledge graph.

The complexity of the item and user preferences relationships cannot be fully captured by current explainable recommender systems even though they frequently concentrate on a single explanation strategy such as item-based or user-based reasoning. User-mediated explanations rely on user behavior and interaction history item-mediated explanations highlight item attributes and content similarity and feature-mediate explanations concentrate on shared or influential features connecting users and items. Each strategy offers insightful information albeit incomplete. This constraint drives the incorporation of various forms of explanation into a cohesive framework. Recommender systems can produce richer more accurate and more reliable explanations by merging item user and feature-mediated explanations and bolstering them using a knowledge graph representation. The evolution of explainable and user-centric recommender systems is greatly

enhanced by such an integrated approach.



Figures 1: A schematic diagram of the music KGs recommendation system

An example of KGs is shown in Figure 1, which can represent node associations like <Singer, Friend, arranger>, additionally to node properties like <Music1, Style, Popular>. Many large-scale KGs, like Satori, Freebase, and Google's Knowledge Graph, have been proposed recently. By adding relatedness between entities, these KGs can help recommender systems by facilitating the creation of KGs for recommendation, enhancing entity information, and generating explainability.

II. RECOMMENDER SYSTEMS OVERVIEW

Many industries have used recommender systems, such as film [24], [17], music [15], [10], POIs [9], [12], news [16], [18], education [13], [8], etc. The recommendation task, which can be broken down into the following processes, is to suggest one or more unobserved things to a specific user. The system first learns a representation ui and vj for the provided item vj and user ui. It then picks up a scoring function f: ui × vj → $\widehat{y_{i,j}}$, This simulates ui preference for vj. Finally, the item preference scores can be ranked to generate the recommendation. There are three primary methods to learning the scoring function and the user/item representation as will be discussed below.

A. Collaborative Filtering

Customers may be drawn to products selected by others with similar interaction histories according to CF. The interaction could be explicit [30] [27] like ratings or implicit [34] [7] like click and watch. The interaction data from multiple users and items required to implement CF-based recommendation further forms the user-item interaction matrix. The two main techniques employed in the CF-based approach are memory-based CF and model-based CF [29]. In particular memory-based CF first calculates the user-user similarity using the user-item interaction data. Unobserved objects are then recommended to the specific user according to the interaction histories of others who resemble them. Alternatively, some models look at the user's previous purchases to see if they are similar. The model-based CF technique attempts to address the sparsity issue by building an inference model. One popular method is the latent factor model [35] [20] which extracts the latent representation of the item and the user from the high dimensional user-item interaction matrix and then uses the inner product or other methods to determine the degree of resemblance between the user and item.

B. Content-based Filtering

Content-based approaches use the items content to display the user and item as opposed to the CF-based model which uses global user-item interaction data to learn the representation of the user and item. The premise behind content-based filtering is that consumers might be attracted to goods that are similar to those they have already used. While the user representation is according to the traits of personally engaged items, the item representation is obtained by extracting attributes from the item's auxiliary information, such as words, photos, etc. In essence, the process of comparing candidate items with the user profile involves matching them with the user's prior data.

Consequently, this approach frequently suggests products that are comparable to those that a customer has previously loved [26].

C. Hybrid Method

The hybrid approach uses a variety of recommendation strategies to get over the drawbacks of utilizing just one kind of approach. One significant The sparsity of user-item interaction data, which makes it challenging to identify related things or users from the standpoint of interaction, is a problem with CF-based recommendation. The cold-start problem is a specific instance of this problem, which makes it challenging to promote new users or items because it is impossible to assess user-user and item-item similarity without interaction histories. Better recommendation performance can be attained by integrating user and item content information, also referred to as user side information and item side information, into the CF-based system [14]. Item attributes [31], [33], [37], such as brands and categories; item multimedia information, such as item reviews [22], [19], textual descriptions [11], image features [23], audio signals [25], and video features [21]. Common options for user side information involve

user's demographic information [28], including occupation, gender, and hobbies; and user network [36], [32]. In this study, recommender system based on KG combine the CF-based method using the KG as supplementary data to make recommendations that are more accurate.

III. KNOWLEDGE GRAPH–BASED METHODS

Knowledge Graph–Based methods are those that make use of structured external information in the format of triples (head entity, relation, tail entity) to enhance recommendation systems. They rely on the knowledge graph's semantic relationships and features to understand the complex interrelations of users, items, and other entities, thus, contributing to the solution of issues like data sparsity and cold start. These techniques are generally divided into three categories:

1 Embedding-Based Methods: It embeds both the entities and connections into a continuous vector space, where spatial proximity denotes similarity.

2 Path-Based Methods: They trace the semantic paths that connect different entities in the knowledge graph, thereby unveiling explicit multi-hop relationships.

3 Propagation-Based Methods: They are the methods that learn the higher-order relationships and feature representations after multiple iterations of information propagation across the graph.

1. Embedding-Based Methods

Embedding-based approaches represent entities and connections within a knowledge graph in the form of low-dimensional vectors. The primary aim is retaining semantic structure between users, items, and features in a vector space. Such embeddings facilitate efficient computation of similarities and recommendations.

Embedding models such as TransE, DistMult, ComplEx, and RotatE are usually applied in recommender systems. The models optimize a score function based on the possibility of relationships among entities to learn the entity representation. The learned representation for users and items can then be combined with a traditional model for the prediction of user preference.

The embedding-based approaches have efficiency and scalability. The approaches can also handle the issue of sparse data by learning the hidden associations between entities. The disadvantage is that certain techniques might not be explainable because the learned entity embeddings represent abstract entities that convey no reasoning chain for recommendations.

A knowledge graph is represented as a set of triples:

$$(h, r, t)$$

where

- h= head entity
- r= relation
- t= tail entity

Each entity and relation is mapped to a vector:

$$h, r, t \in \mathbb{R}^d$$

In TransE, the relation is modeled as a translation:

$$h + r \approx t$$

The scoring function is:

$$f(h, r, t) = \left||h + r - t|\right| \qquad eq(1)$$

Lower scores indicate stronger relationships.

Recommendation Score

For a user $u$and item i:

$$score(u, i) = u^T \text{I} \qquad \text{eq(2)}$$

2.Path-Based Methods

Path-based methods concentrate more on identifying important paths in the knowledge graph that reveal a linkage between the users and the items. The paths can be considered as the semantic relations in terms of common attributes or interactions where the relations can be directly utilized to ascertain the preference of the users. In these techniques, the recommendation ratings are determined using the number, type, and/or importance of paths connecting the user and the items. For instance, the recommendation might be justified based on a path demonstrating that the user likes a movie that has the same genre and director as the recommended movie. Path-based techniques are extremely interpretable since the reasoning process can easily be tracked using the graph. Even if these techniques are extremely explainable, High computational complexity may have several drawbacks, particularly when handling, large or denser graphs. Also, finding meaningful paths is a difficult task.
A path $p$between user $u$and item $i$is defined as:

$$p = \left(u \xrightarrow{r_1} e_1 \xrightarrow{r_2} \ldots\ldots \xrightarrow{r_k} i\right) \qquad \text{eq(3)}$$

The path score is computed as:

$$score_p(u,i) = \prod_{j=1}^{k} w(r_j) \qquad \text{eq(4)}$$

where $w(r_j)$is the weight of relation r_j.

Overall Recommendation Score

$$\mathrm{score}(\mathrm{u},\mathrm{i}) = \sum_{\mathrm{p}\in\mathrm{P}} \mathrm{score}(\mathrm{u},\mathrm{i}) \qquad \text{eq(5)}$$

where $P(u,i)$ is the set of valid paths?

3.Propagation-Based Methods

With the aid of nearby nodes in the graph knowledge propagation mechanisms are used by propagation-based learning techniques to collect knowledge. These methods typically use GNNs such as Graph Convolutional Networks (GCNs) to update node representations in a methodical manner with the assistance of nearby nodes. The node can learn graph structure both locally and globally by aggregating features from the entities it is connected to at each propagation step. This technique enables the system to learn characteristics that represent the graph structure of several hops in the knowledge graph. The propagation techniques can combine explanations and have a high recommendation accuracy. However, these techniques are computationally demanding may not be scalable to large graphs and may over smooth. The development of propagation strategies for these algorithms is an ongoing field of study.

Let $h_v^{(l)}$be the representation of node $v$at layer l.

Node update rule:

$$h_v^{(l+1)} = \sigma\left(\sum_{u\epsilon N(v)} \frac{1}{|N(v)|} W^{(l)} h_u^{(l)}\right) \qquad \text{eq(6)}$$

Where,

- $N(v)$=neighbors of node v
- $W^{(l)}$= learnable weight matrix
- $\sigma(\cdot)$= activation function

Final Recommendation Score

$$score(u,i) = \left(h_u^{(L)}\right)^T h_u^{(L)} \qquad \text{eq(7)}$$

where L is the final propagation layer.

Simple Comparison

Method Type	Mathematical Core	Explainability	Complexity
Embedding-Based	Vector translation (TransE, RotatE)	Low	Low
Path-Based	Path probability/ranking	High	Medium
Propagation-Based	Neighbor aggregation (GCN)	Medium–High	High

IV. FUTURE DIRECTIONS

The explainability and accuracy of Knowledge Graph-based Recommenders have significantly improved but there are still a lot of unresolved research issues. To develop reliable and user-focused recommenders these issues must be resolved.

1. Unified Explainable Recommendation Frameworks: The goal of future studies should be to create frameworks that integrate the ideas of propagation-based embedding and path-based models. The interpretability of paths the efficacy of embeddings and the representational capabilities of graph neural networks can all help these models generate more precise and superior explanations.

2. Standardized Explanation Evaluation Measures: There are a few standard metrics that can be used to assess explainable recommender's explanations. The feature study can create both quantitative and qualitative metrics to evaluate the degree of explanation satisfaction as well as the precision and openness of explanations. Comparing different explanation models for recommenders will be made simpler as a result.

3. Efficiency and Scalability Gains: Scalability becomes a major issue as knowledge graphs become more complex. In feature works scalable methods can be investigated for efficient graph sampling and learning at lower computational costs without affecting the recommendation performance metric.

4. Dynamic and Changing Knowledge Graphs: Despite the fact that the data is actually dynamic all current models have taken into account a static knowledge graph. It is necessary to look into learning strategies that modify the knowledge graph representation in real time based on user preferences and item information.

5. User-centered and human-in-the-loop explanations: Future recommender systems should consider feedback to enhance explanations and recommendations. Customized explanations can be provided by human-in-the-loop systems to improve system clarity and trust satisfaction.

6. Fairness Bias and Ethical Consideration: The fairness and equity of knowledge graph-based recommenders should be covered in the feature research. For specialized fields like healthcare and education the question of objectivity ethics and explainability is especially crucial.

V. CONCLUSION

With a focus on explainability knowledge graph-based recommender systems have been thoroughly examined in this review. Despite their high efficiency traditional recommendation algorithms have a number of shortcomings including sparsity the cold start problem and interpretability problems. By adding semantic relationships between the user's items and features knowledge graphs can get around these restrictions and enhance the interpretability and accuracy of recommendations. The study has methodically divided previous research into three groups: embedding-based propagation-based and path-based. To develop dependable and user-centric recommender systems it is crucial to combine item user and feature-level explanations under a single knowledge graph framework even though embedding-based methods are scalable path-based methods have high explainability and propagation-based methods perform better using graph neural networks. This review has finally brought to light the most significant unresolved issues including standardized explanation evaluation ethics scalability and dynamic knowledge modelling.

REFERENCES

[1] Wang, H.; Li, Q.; Luo, H.; Tang, Y. The Graph Attention Recommendation Method for Enhancing User Features Based on Knowledge Graphs. Mathematics 2025, 13, 390. https://doi.org/10.3390/math13030390.

[2] Huang, J.; Xie, Z.; Zhang, H.; Yang, B.; Di, C.; Huang, R. Enhancing Knowledge-Aware Recommendation with Dual-Graph Contrastive Learning. Information 2024, 15, 534. https://doi.org/10.3390/info15090534.

[3] Yang, S.; Du, Q.; Zhu, G.; Cao, J.; Chen, L. Balanced influence maximization in social networks based on deep reinforcement learning. Neural Netw. 2024, 169, 334–351.

[4] Yang, S.; Du, Q.; Zhu, G.; Cao, J.; Qin, W. Neural attentive influence maximization model in social networks via reverse influence sampling on historical behavior sequences. Expert Syst. Appl. 2024, 249, 123491.

[5] Nesmaoui, R.; Louhichi, M.; Lazaae, M. A Collaborative Filtering Movies Recommendation System based on Graph Neural Network. Procedia Comput. Sci. 2023, 220, 456–461.

[6] Dai, G.; Wang, X.; Zou, X.; Liu, C.; Cen, S. MRGAT: Multi-Relational Graph Attention Network for knowledge graph completion. Neural Netw. 2022, 154, 234–245.

[7] C. Wang, H. Zhu, C. Zhu, C. Qin, and H. Xiong, "Setrank: A setwise bayesian approach for collaborative ranking from im plicit feedback," in Proceedings of the AAAI Conference on Artificial Intelligence, 2020.

[8] C. Qin, H. Zhu, C. Zhu, T. Xu, F. Zhuang, C. Ma, J. Zhang, and H.Xiong, "Duerquiz: A personalized question recommender system for intelligent job interview," in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2019, pp. 2165–2173.

[9] D. Xi, F. Zhuang, Y. Liu, J. Gu, H. Xiong, and Q. He, "Modelling of bi-directional spatio-temporal dependence and users dynamic preferences for missing poi check-in identification," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, 2019, pp. 5458–5465.

[10] H. Wang, F. Zhang, M. Zhao, W. Li, X. Xie, and M. Guo, "Multi task feature learning for

knowledge graph enhanced recommendation," in The World Wide Web Conference, ser. WWW '19. New York, NY, USA: ACM, 2019, pp. 2000–2010.

[11] J. Han, L. Zheng, Y. Xu, B. Zhang, F. Zhuang, S. Y. Philip, and W. Zuo, "Adaptive deep modeling of users and items using side information for recommendation," IEEE transactions on neural networks and learning systems, 2019.

[12] P. Zhao, H. Zhu, Y. Liu, J. Xu, Z. Li, F. Zhuang, V. S. Sheng, and X. Zhou, "Where to go next: A spatiotemporal gated network for next poi recommendation," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, 2019, pp. 5877–5884.

[13] Z. Huang, Q. Liu, C. Zhai, Y. Yin, E. Chen, W. Gao, and G. Hu, "Exploring multi-objective exercise recommendations in online education systems," in Proceedings of the 28th ACM International Conference on Information and Knowledge Management, 2019, pp. 1261–1270.

[14] Z. Sun, Q. Guo, J. Yang, H. Fang, G. Guo, J. Zhang, and R. Burke, "Research commentary on recommendations with side information: A survey and research directions," Electronic Commerce Research and Applications, vol. 37, p. 100879, 2019.

[15] B. Hu, C. Shi, W. X. Zhao, and P. S. Yu, "Leveraging meta path based context for top-n recommendation with a neural co-attention model," in Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2018, pp. 1531–1540.

[16] H. Wang, F. Zhang, J. Wang, M. Zhao, W. Li, X. Xie, and M. Guo, "Ripplenet: Propagating user preferences on the knowledge graph for recommender systems," in Proceedings of the 27th ACM International Conference on Information and Knowledge Management. ACM, 2018, pp. 417–426.

[17] J. Huang, W. X. Zhao, H. Dou, J.-R. Wen, and E. Y. Chang, "Im proving sequential recommendation with knowledge-enhanced memory networks," in The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. ACM, 2018, pp. 505–514.

[18] H. Wang, F. Zhang, X. Xie, and M. Guo, "Dkn: Deep knowledge aware network for news recommendation," in Proceedings of the 2018 World Wide Web Conference, ser. WWW '18. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2018, pp. 1835–1844.

[19] Y. Xu, Y. Yang, J. Han, E. Wang, F. Zhuang, and H. Xiong, "Exploiting the sentimental bias between ratings and reviews for enhancing recommendation," in 2018 IEEE International Conference on Data Mining (ICDM). IEEE, 2018, pp. 1356–1361.

[20] F. Zhuang, Z. Zhang, M. Qian, C. Shi, X. Xie, and Q. He, "Rep resentation learning via dual-autoencoder for recommendation," Neural Networks, vol. 90, pp. 83–89, 2017.

[21] J. Chen, H. Zhang, X. He, L. Nie, W. Liu, and T.-S. Chua, "Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention," in Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval. ACM, 2017, pp. 335–344.

[22] L. Zheng, V. Noroozi, and P. S. Yu, "Joint deep modeling of users and items using reviews for recommendation," in Proceedings of the Tenth ACM International Conference on Web Search and Data Mining. ACM, 2017, pp. 425–434.

[23] W.-T. ChuandY.-L. Tsai, "A hybrid recommendation system considering visual information for predicting favorite restaurants," World Wide Web, vol. 20, no. 6, pp. 1313–1331, 2017.

[24] F. Zhang, N. J. Yuan, D. Lian, X. Xie, and W.-Y. Ma, "Collaborative knowledge base embedding for recommender systems," in Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ser. KDD '16. New York, NY, USA: ACM, 2016, pp. 353–362.

[25] D. Liang, M. Zhan, and D. P. Ellis, "Content-aware collaborative music recommendation using pre-trained neural networks." in ISMIR, 2015, pp. 295–301.

[26] P. Lops, M. De Gemmis, and G. Semeraro, "Content-based recommender systems: State of the art and trends," in Recommender systems handbook. Springer, 2011, pp. 73–105.

[27] G. Jawaheer, M. Szomszor, and P. Kostkova, "Comparison of implicit and explicit feedback from an online music recommendation service," in proceedings of the 1st international workshop

on 16 information heterogeneity and fusion in recommender systems, 2010, pp. 47–51.

[28] Z. Gantner, L. Drumond, C. Freudenthaler, S. Rendle, and L. Schmidt-Thieme, "Learning attribute-to-feature mappings for cold-start recommendations." in ICDM, vol. 10. Citeseer, 2010, pp. 176–185.

[29] X. Su and T. M.Khoshgoftaar, "A survey of collaborative filtering techniques," Advances in artificial intelligence, vol. 2009, 2009.

[30] X. Amatriain, J. M. Pujol, and N. Oliver, "I like it... i like it not: Evaluating user ratings noise in recommender systems," in International Conference on User Modeling, Adaptation, and Personalization. Springer, 2009, pp. 247–258.

[31] S. Sen, J. Vig, and J. Riedl, "Tagommenders: connecting users to items through tags," in Proceedings of the 18th international conference on World wide web. ACM, 2009, pp. 671–680.

[32] M. Jamali and M. Ester, "Trustwalker: a random walk model for combining trust-based and item-based recommendation," in Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2009, pp. 397–406.

[33] Y. Zhen, W.-J. Li, and D.-Y. Yeung, "Tagicofi: tag informed col laborative filtering," in Proceedings of the third ACM conference on Recommender systems. ACM, 2009, pp. 69–76.

[34] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in 2008 Eighth IEEE International Conference on Data Mining. Ieee, 2008, pp. 263–272.

[35] R. Salakhutdinov and A. Mnih, "Bayesian probabilistic matrix factorization using markov chain monte carlo," in Proceedings of the 25th international conference on Machine learning, 2008, pp. 880 887.

[36] P. Massa and P. Avesani, "Trust-aware recommender systems," in Proceedings of the 2007 ACM conference on Recommender systems. ACM, 2007, pp. 17–24.

[37] C.-N. Ziegler, G. Lausen, and L. Schmidt-Thieme, "Taxonomy driven computation of product recommendations," in Proceedings of the thirteenth ACM international conference on Information and knowledge management. ACM, 2004, pp. 406–415.