

KGUF: Simple Knowledge-aware Graph-based Recommender with User-based Semantic Features Filtering

Salvatore Bufi¹, Alberto Carlo Maria Mancino¹, Antonio Ferrara¹,
Daniele Malitesta^{2*}, Tommaso Di Noia¹, and Eugenio Di Sciascio¹

¹ Politecnico di Bari, `name.surname@poliba.it`

² CentraleSupélec, Inria, Université Paris-Saclay, `name.surname@centralesupelec.fr`

Abstract. The recent integration of Graph Neural Networks (GNNs) into recommendation has led to a novel family of Collaborative Filtering (CF) approaches, namely Graph Collaborative Filtering (GCF). Following the same GNNs wave, recommender systems exploiting Knowledge Graphs (KGs) have also been successfully empowered by the GCF rationale to combine the representational power of GNNs with the semantics conveyed by KGs, giving rise to Knowledge-aware Graph Collaborative Filtering (KGCF), which use KGs to mine hidden user intents. Nevertheless, empirical evidence suggests that computing and combining user-level intent might not always be necessary, as simpler approaches can yield comparable or superior results while keeping explicit semantic features. Under this perspective, user historical preferences become essential to refine the KG and retain the most discriminating features, thus leading to concise item representation. Driven by the assumptions above, we propose KGUF, a KGCF model that learns latent representations of semantic features in the KG to better define the item profile. By leveraging user profiles through decision trees, KGUF effectively retains only those features relevant to users. Results on three datasets justify KGUF’s rationale, as our approach is able to reach performance comparable or superior to SOTA methods while maintaining a simpler formalization.

Keywords: recommendation · knowledge graphs · graph neural networks.

1 Introduction

Recommender Systems find widespread application across diverse domains, owing to their proficiency in furnishing users with personalized items [9, 45, 47]. Typically, these systems rely on past user-item interactions to extrapolate new preferences and generate personalized suggestions [11, 19]. Collaborative Filtering [23] (CF) methods adeptly extract similarity patterns from user-item interactions, constituting (to date) the most diffused and winning recommendation paradigm [19, 26, 28]. Nevertheless, by design, the learned user and item profiles

* Work done while at Politecnico di Bari.

in traditional CF recommender systems cannot embed the information conveyed by the underlying multi-hop user-item relations, thus disregarding a rich source of potentially-useful information [27, 39]. Following the popular graph representation learning [17] wave, Graph Neural Networks (GNNs) have recently been applied to CF approaches, giving rise to a fresh family of recommender systems [18, 27, 43, 46], better known as Graph Collaborative Filtering (GCF). Despite the success of traditional CF and GCF recommendation systems, such approaches often overlook the nuances in user preferences related to specific item characteristics; indeed, this represents a missed opportunity for enhancing user personalization. As opposed to CF systems, content-based recommendation [14] leverages the attributes of historically-favored products, proposing unseen items that align with user distinct tastes [19]. In these systems, Knowledge Graphs (KGs) have been shown to provide a meaningful source of structured information [3, 10, 48, 49] used to craft features that enrich the item description, thanks to the mapping between items and semantic entities, and helps to tackle open challenges in recommendation, such as the graph sparsity.

As a recent trend, an increasing number of recommender system algorithms currently propose to combine the representational capabilities of GNNs with the rich information conveyed by KGs, resulting in a novel hybrid family of recommendation techniques which might be suitably named as Knowledge-aware Graph Collaborative Filtering [20, 40, 44, 48] (KGCF). Most of KGCF methods model the user intent to enhance recommendation quality and interpretability, thus outperforming similar baselines techniques [25, 38, 40]. However, such architectures often require convoluted aggregations of several ad-hoc methodologies and strategies to learn meaningful user and item profiles, making it hard to apply KGCF approaches in real-world scenarios. In this regard, we believe that these architectures can be dramatically simplified while maintaining comparable or even superior performance. In particular, we consider to utilize the content information from KGs solely for describing item representations, refraining from any attempt to directly incorporate it into building user profiles, which can be instead indirectly learned by means of collaborative filtering techniques.

Driven by these assumptions, this paper presents a novel KGCF recommendation algorithm named **KGUF** (short for **K**nowledge **G**raph **U**ser-based **F**iltering) designed to efficiently harness side information from KGs. From a technical perspective, KGUF learns user and item embeddings through linear propagation on the collaborative graph. In contrast to other approaches, our solution, sketched in Figure 2, is able to integrate latent representations for semantic features right into the item representation. Moreover, sharing the same principle but replacing it with a simple mechanism, we propose substituting the intent modeling with a decision tree mechanism that selects the most meaningful semantic features in the KGs and filters out noisy features that poorly contribute to the item description but heavily impact the computational costs. All in all, the key contributions of this paper are summarized as follows:

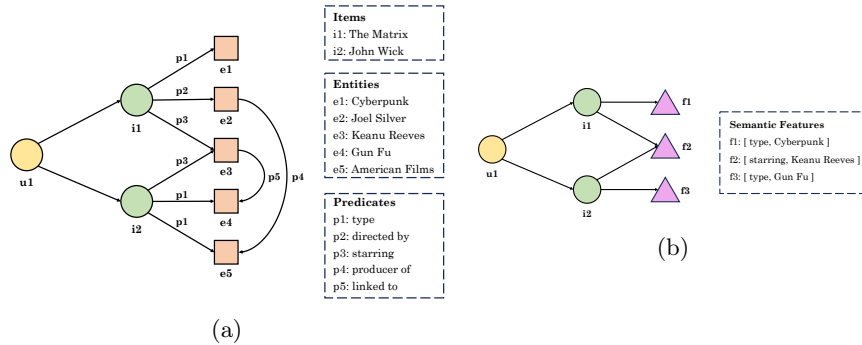


Fig. 1: An example of how KGUF models the item representation. Figure 1a shows the entities in the knowledge graph linked to the items. In figure 1b, the most relevant semantic features selected by KGUF are reported.

1. We propose a novel recommendation model, namely KGUF, which effectively selects and integrates user-relevant semantic features during the graph learning phase to improve item representation.
2. We conduct extensive experiments over three well-known recommendation datasets to assess the efficacy of our framework and evaluate the contribution of collaborative and content signals on KGUF.
3. We delve into using decision trees for selecting semantic features relevant to users, investigating how the effects of negative sampling and the presence of constraints regarding the tree depth affect the recommendation performance.

To foster the replicability of the experiments and results, we release the code for KGUF and the baselines, along with the datasets and configurations ³.

2 Related work

This section aims to survey and compare other approaches in recommendation literature leveraging KGs and graph representation learning strategies.

KGs information in recommendation. As a general trend, the incorporation of information coming from KGs into recommendation systems has consistently demonstrated improvements in model performance and interpretability across diverse domains [3, 10, 48, 49]. Indeed, the utilization of KGs has been effective in boosting different recommendation tasks, including hybrid collaborative/content-based recommendation [3, 24], where the KG is of support to the lack of collaborative information, eventually enhancing performance; explainable recommendation [5, 34], where the KG serves as a robust information source for delivering human-like explanations; and user modeling [4, 37], with the semantic description of items contributing to defining user profiles. KGs also offer a solution to

³ <https://github.com/sisinflab/KGUF>

the cold-start problem by augmenting observed interactions with KG-reachable unobserved items [35]. However, it is crucial to note that in some cases, improper extraction of information may potentially weaken the model capabilities [8].

Graph representation learning for recommendation. Recommender systems based on graph representation learning strategies have demonstrated their superior performance by mining high-order user-item relationships embedded within the graph structure [29, 30, 33, 43]. Inspired by the graph convolutional network layer (GCN) [21], early approaches like PinSage [45] and NGCF [39] have adapted GCN to the user-item bipartite graph. Later research has explored simplifying the GCN layer, as seen in methods like LightGCN [18] and LR-GCCF [7], which streamline model complexity by eliminating non-linearities and feature transformations in node embeddings. Additionally, UltraGCN [27] proposes a mathematical proxy to the infinite message-passing layer, thus limiting the negative effects of over-smoothing. Another avenue of research focuses on pruning noisy user-item interactions to discern meaningful preference patterns. Inspired by the graph attention network [36] (GAT), methodologies like DGCF [41] aim to identify the underlying intents of individual user-item interactions on a fine-grained level. Furthermore, works such as [2] prove how user-item edges can incorporate multimodal content, such as user reviews, to enhance recommendation and address the challenge of over-smoothing.

KGs information and graph representation learning for recommendation. Recent literature [20, 44, 48] has ventured beyond traditional collaborative filtering graph-based methods by incorporating information from KGs to enhance item descriptions. One example is KGAT [38], which explicitly models the KG in an end-to-end manner. It refines node embeddings using an attention mechanism to discern the importance of neighbors. In a different approach, the authors of KGIN [40] utilize the KG to define users’ intents as weights, combining these weights with their relations. CKAN [42] proposes a method that integrates collaborative signals with knowledge data. More recently, KGs have been employed to formulate first-order logical rules, combined with graph embeddings, enabling recommenders based on neuro-symbolic reasoning [31].

3 Background

In this section, we introduce the notion of KGs and the basics of decision trees used for content modelling, along with the problem we aim to address.

3.1 Knowledge graphs for recommendation

A knowledge graph stores structured information of real-world facts in the form of a heterogeneous graph which can be used to retrieve item attributes [6, 13]. Formally, it is presented as a set of triples in the form $\sigma \xrightarrow{P} \omega$. Each triple indicates that the head entity σ is connected to the tail entity ω through the

relation ρ . Given a set of entities \mathcal{V} and a set of relations \mathcal{R} , a knowledge graph \mathcal{KG} can be represented as follows:

$$\mathcal{KG} = \{\sigma \xrightarrow{\rho} \omega \mid \sigma, \omega \in \mathcal{V}, \rho \in \mathcal{R}\}. \quad (1)$$

Mapping items in the catalog using knowledge graphs, i.e., $\mathcal{I} \subseteq \mathcal{V}$, enables the descriptions of items based on semantic features.

3.2 Decision trees

A decision tree is a hierarchical tree-like structure used in machine learning for classification tasks. It iteratively partitions the dataset into subsets depending on the values of input features, resulting in a tree structure in which each internal node represents a feature-based decision. The decision-making process is guided by criteria such as Gini impurity or information gain.

In practical terms, decision trees aim to identify variables whose values contribute the most to reducing another random variable's entropy, or uncertainty. Mathematically, the entropy $H(V)$ of a random variable V with k possible values in v_1, \dots, v_k is formally defined as:

$$H(V) = - \sum_{i=1}^k \Pr(V = v_i) \log_2 \Pr(V = v_i). \quad (2)$$

If observing the value of another variable X with n possible values in x_1, \dots, x_n results in a reduction of the entropy of V , then X is said to exhibit information gain over V :

$$\begin{aligned} IG(V, X) &= H(V) - H(V|X), \\ H(V|X) &= \sum_{i=1}^n P(X = x_i) H(V|X = x_i). \end{aligned} \quad (3)$$

Decision trees are one possible lightweight solution to filter out noisy semantic features in the KG and select the most relevant ones from the user perspective.

4 Methodology

In this section, we present KGUF and its components. In particular, we delve into the semantic item modeling and its refinements based on user decisions. Then, we introduce our simple but effective knowledge-aware aggregation schema.

4.1 Modeling items through knowledge graph

Assuming that each item in the item set \mathcal{I} of our recommendation problem is associated with an entity in \mathcal{V} , essentially being one of the entities within the knowledge graph (\mathcal{KG}), we can model the items in the catalog by leveraging the knowledge embedded in the KG. The rationale behind this modeling lies in the

notion that each item can be comprehensively characterized by a distinct set of features, some of them influencing user decisions in varying ways.

Given this premise, the KG can be explored to identify connections with other entities for each item, serving as its attributes in the form $\sigma \xrightarrow{\rho} \omega$, where ρ and ω represent the predicate and object of the KG triple, respectively. As an example, if properly encoded in the reference knowledge graph, the movie *Harry Potter* could be semantically described by the predicate-object pairs $\langle \text{director}, \text{Chris Columbus} \rangle$ and $\langle \text{genre}, \text{Fantasy} \rangle$. Formally, we utilize the set of semantic features to depict the item $i \in \mathcal{I}$ based on its first-order connectivity extracted from the \mathcal{KG} :

$$\mathcal{F}_i = \{ \langle \rho, \omega \rangle \mid i \xrightarrow{\rho} \omega \in \mathcal{KG} \}. \quad (4)$$

4.2 Refining item features with user decision

When confronted with a novel item, users focus on a subset of its numerous features. To capture this user behavior, KGUF uses decision trees to conduct a behavioral analysis at the user level, leveraging her past ratings. The objective is to discern the most pertinent features within the knowledge graph that effectively describe items in the catalog while filtering out noisy ones. This tailored approach ensures that the model accentuates features that hold significant influence across all users in the system.

Formally, let $\mathcal{I}_u^+ \subseteq \mathcal{I}$ denote the set of items positively rated by each user u in the set \mathcal{U} of our recommendation problem, and let $\mathcal{I}_u^- = \mathcal{I} \setminus \mathcal{I}_u^+$ be the set of items not enjoyed by user u . Let $\tilde{\mathcal{I}}_u^- \subseteq \mathcal{I}_u^-$ be a selection of items randomly sampled from \mathcal{I}_u^- , such that $|\tilde{\mathcal{I}}_u^-| = |\mathcal{I}_u^+|$. Finally, $\mathcal{I}_u = \mathcal{I}_u^+ \cup \tilde{\mathcal{I}}_u^-$ is the set of items selected to depict the past behavior of the user.

For each user u , the set of semantic features $\mathcal{F}_u = \bigcup_{i \in \mathcal{I}_u} \mathcal{F}_i$ is used to compute the user decision tree T_u that, accordingly to the definition of Information Gain [3], is able to find the minimum subset of semantic features $\mathcal{F}_u^{T_u} \subseteq \mathcal{F}_u$ that can distinguish whether an item belongs to \mathcal{I}_u^+ or $\tilde{\mathcal{I}}_u^-$. Finally, each item i is formally described by the subset of its semantic features that appear in at least one user decision tree:

$$\mathcal{F}_i^* = \mathcal{F}_i \cap \bigcup_{u \in \mathcal{U}} \mathcal{F}_u^{T_u} \quad (5)$$

4.3 Knowledge-aware aggregation

Inspired by LightGCN [18], the embeddings of each user and item, denoted as $\mathbf{e}_u \in \mathbb{R}^d$ and $\mathbf{e}_i \in \mathbb{R}^d$ respectively, are acquired through the recursive aggregation of their multi-hop neighbors' representations via message propagation across L convolutional layers. The resulting representations at various levels, $(\mathbf{e}_u^{(0)}, \dots, \mathbf{e}_u^{(L)})$ and $(\mathbf{e}_i^{(0)}, \dots, \mathbf{e}_i^{(L)})$, emerge from the integration of content information with collaborative signals originating from first-hop neighbor nodes and propagated from the preceding layer.

By expressing knowledge graph features through latent vectors \mathbf{e}_f , KGUF adeptly represents items based on their explicit characteristics. This modeling

approach furnishes an explicit content-based representation for each item, situating it within the latent feature space as an amalgamation of vectors delineating its distinct characteristics. Formally:

$$\begin{aligned}\mathbf{e}_u^{(l)} &= \sum_{i \in \mathcal{N}(u)} \frac{1}{\sqrt{|\mathcal{N}(u)|} \sqrt{|\mathcal{N}(i)|}} \mathbf{e}_i^{(l-1)}, \\ \mathbf{e}_i^{(l)} &= \frac{\alpha}{|\mathcal{F}_i^*|} \sum_{f \in \mathcal{F}_i^*} \mathbf{e}_f + (1 - \alpha) \sum_{u \in \mathcal{N}(i)} \frac{1}{\sqrt{|\mathcal{N}(u)|} \sqrt{|\mathcal{N}(i)|}} \mathbf{e}_u^{(l-1)},\end{aligned}\tag{6}$$

where α serves as a weight factor for balancing content and collaborative contributions in item representations.

It is worth noting that, in contrast to LightGCN, KGUF introduces a knowledge-aware correction term in the item embeddings, which will affect also user representation through message passing schema. The collaborative and content signals collaboratively contribute to capturing long-range influences and leveraging the rich information derived from knowledge graphs. The use of decision trees to mediate content information guarantees that only relevant features engage in embedding modeling, ensuring a succinct and meaningful representation of users and items.

After completing L propagation hops, the user and item embeddings are derived as follows:

$$\mathbf{e}_u = \sum_{l=0}^L \frac{1}{1+l} \mathbf{e}_u^{(l)}, \quad \mathbf{e}_i = \sum_{l=0}^L \frac{1}{1+l} \mathbf{e}_i^{(l)}.\tag{7}$$

The introduction of the scaling factor $1/(1+l)$ serves to mitigate the oversmoothing problem that may arise with an increased number of explored hops [18].

4.4 Model learning

The training stage involves learning the values of user, item, and feature embeddings to align user-item estimated interactions with the actual preferences of users. Given the embeddings \mathbf{e}_u and \mathbf{e}_i , we employ the dot product of user and item representations as the prediction, expressed as:

$$\hat{r}_{ui} = \mathbf{e}_u^\top \mathbf{e}_i.\tag{8}$$

To optimize the model parameters, we utilize the pairwise optimization algorithm Bayesian Personalized Ranking (BPR). This algorithm operates on the assumption that a user u favors a consumed item i^+ over a non-consumed item i^- . It optimizes the model by maximizing, for each pair of i^+ and i^- , a function based on the difference $r_{ui^+} - r_{ui^-}$. Specifically, by constructing a training set $\mathcal{T} = \{(u, i^+, i^-) \mid (u, i^+) \in \mathcal{R}, (u, i^-) \notin \mathcal{R}, i^- \in \mathcal{I}\}$, BPR aims to optimize the following loss, where σ represents the sigmoid function:

$$\mathcal{L}_{\text{BPR}} = \sum_{(u, i^+, i^-) \in \mathcal{T}} -\ln \sigma(\hat{r}_{ui^+} - \hat{r}_{ui^-}),\tag{9}$$

Finally, we use a joint learning approach to include the L2-regularization loss as follows:

$$\mathcal{L} = \mathcal{L}_{\text{BPR}} + \lambda \|\Theta\|_2^2, \quad (10)$$

here Θ encompass all the learnable parameter, specifically $\Theta = \{\mathbf{e}_u, \mathbf{e}_i, \mathbf{e}_f \mid u \in \mathcal{U}, i \in \mathcal{I}, f \in \bigcup_{i \in \mathcal{I}} \mathcal{F}_i^*\}$, while λ controls the regularization term.

5 Experimental settings

This section delineates the experimental configuration employed to evaluate the recommendation performance of KGUF.

5.1 Datasets and Knowledge Graph

We compare the performance of KGUF with state-of-the-art solutions on three well-known recommendation datasets: *MovieLens 1M*⁴, *Yahoo! Movies*⁵, and *Facebook Books*⁶. MovieLens 1M and Yahoo! Movies are two datasets in the movie recommendation domain that collect explicit ratings (ranging from 1 to 5) from 6,040 and 4000 users, respectively, over catalogs of 3,706 and 2,626 items. Overall, they report 1,000,209 and 69,846 ratings, respectively. *Facebook Books* is a dataset that collects 18,978 implicit user feedback in the book recommendation domain. Since its ratings refer to 4,000 users and 2,626 items, it is characterized by greater sparsity. All the datasets are evaluated in an implicit feedback scenario: explicit ratings are binarized by applying a threshold value of 3. Only ratings greater or equal to the threshold value are recorded as implicit positive feedback. In order to reduce the noisy contribution of users and items with too few occurrences, we apply an iterative 5-core strategy, dropping users and items with less than five appearances. Through explicit linkings, the items in the dataset catalogs are directly linked to their semantic representation in *DBpedia*, a well-recognized public knowledge graph. For a fair comparison, we filter the noisy item-linked entities following, for each model, the filtering approach reported by the original authors.

5.2 Baselines

We empirically validate the effectiveness of KGUF by comparing its performance in terms of recommendation accuracy with different baselines from five related categories described in the following.

Unpersonalized We test Random and Most Popular recommendations as a basis for discerning significance and behaviors influenced by popularity.

⁴ MovieLens 1M Dataset: <https://grouplens.org/datasets/movielens/1m/>

⁵ Yahoo! Movies User Ratings and Descriptive Content Information, v.1.0: <https://webscope.sandbox.yahoo.com/catalog.php?datatype=r>

⁶ Facebook Books (Linked Open Data challenge co-located with ESWC 2015): <https://2015.eswc-conferences.org/important-dates/call-RecSys.html>

Collaborative Filtering BPR-MF [28] and Item-kNN [22] are two widely recognized collaborative approaches. The former follows a pair-wise learning approach as KGUF, and the latter exploits similarities between items by computing the nearest neighbors.

Knowledge-Aware Among these models, we selected KaHFM [4] and KGFlex [3, 12], since they share with KGUF the feature representation but differ in the way they exploit it. KaHFM uses knowledge for initializing latent factors in factorization machines. KGFlex, instead, models the user-item interaction by means of a sparse collection of the most relevant semantic features.

Graph-Based LightGCN [18] rethinks the classic GCN aggregation schema by discarding feature transformation and non-linear activation functions. It results in linear aggregations with improved performance. DGFC [41] introduces a graph disentangling strategy for intercepting hidden user intents to model intent-dependent user-item interactions.

Graph-based with Knowledge KGIN [40] exploits the KG for computing users’ intents as a weighted combination of semantic relations. User-item connections are interpreted by means of a limited set of relevant intents. KG-TORe [25] mines the user motivations by looking at the discriminant semantic features in her history. These explicit features are then used to enrich the user and the item node representation.

5.3 Evaluation protocol and reproducibility

For the sake of reproducibility, we provide our code, data, and comprehensive instructions to enable the replication of all stages in our experimental procedures⁷. This paper follows the same evaluation protocol and experimental setting of [25]. Moreover, in order to compare KGUF with the same results, we adopt the reproducibility framework Elliot [1]. Each dataset is split into train, test, and validation sets by retaining 72%, 20%, and 8% of the original user ratings, respectively [16], and adopting the *all unrated protocol* [32].

Coherently with [25], the embedding size is fixed to 64, and each vector is randomly initialized using a Xavier initializer [15]. Then, we train KGUF on 20 different combinations of hyperparameters explored with a Bayesian optimization algorithm and apply early-stopping with a 5-epoch patience. We validate the our model on nDCG@10, since this metric enhances the capability of predicting top-ranking items. Moreover, the batch size is adapted proportionally to the dataset size: 64, 256, and 2048, respectively, for *Facebook Books*, *Yahoo! Movies*, and *Movielens 1M*.

6 Results and discussion

This section examines the performance of KGUF when compared to state-of-the-art baselines. Additionally, we present hyperparameter and ablation studies

⁷ <https://github.com/sisinflab/KGUF>

Table 1: Accuracy performance comparison of KGUF with the selected baselines. All the models are grouped into their respective categories.

	Facebook Books			Yahoo! Movies			MovieLens 1M		
	nDCG	HR	Recall	nDCG	HR	Recall	nDCG	HR	Recall
Random	0.0074	0.0256	0.0110	0.0065	0.0342	0.0090	0.0096	0.0836	0.0036
MostPop	0.0787	0.2535	0.1117	0.1185	0.3238	0.1501	0.1730	0.6537	0.0773
BPR-MF	0.0945	0.2973	0.1358	0.2303	0.5228	0.2843	0.3100	0.8811	0.1627
Item-kNN	0.1039	0.3353	0.1556	0.2258	0.5033	0.2658	0.2932	0.8504	0.1490
KGFlex	0.0712	0.2235	0.1001	0.1758	0.4360	0.2022	0.0111	0.1058	0.0053
kaHFM	0.0843	0.2776	0.1221	0.2085	0.5064	0.2621	0.2992	0.8654	0.1526
LightGCN	0.0969	0.3170	0.1435	0.2230	0.5085	0.2712	0.3081	0.8813	0.1614
DGCF	0.1001	0.3192	0.1444	0.2356	0.5308	0.2848	0.3124	0.8819	0.1637
KGIN	0.1023	0.3112	0.1448	0.2368	0.5500	<u>0.2971</u>	0.3082	0.8854	0.1638
KGTORe	0.1188	0.3601	0.1698	<u>0.2471</u>	<u>0.5566</u>	0.2944	<u>0.3212</u>	<u>0.8957</u>	<u>0.1710</u>
KGUF	<u>0.1156</u>	<u>0.3594</u>	<u>0.1697</u>	0.2561	0.5685	0.3058	0.3277	0.8998	0.1755

conducted to evaluate the effectiveness of the proposed approach. Specifically, this section delves into experimental analyses aimed at answering the following research questions:

- **RQ1 Overall performance:** How does KGUF performance compare to that of SOTA recommenders? Additionally, what is the impact of the proposed knowledge-aware aggregation schema on the overall performance?
- **RQ2 Decision Tree Construction:** To what extent does the construction of trees impact the performance of the proposed framework?

6.1 RQ1: Overall performance

Firstly, we compare the recommendation accuracy of all the methods on three different datasets and report the results in Table 1. We denote the highest-performing result by boldfacing it and underscore the second-best result.

We observe that KGUF shows performance comparable or superior to all baselines over the three datasets. This improvement can be attributed to two key factors. Firstly, by employing user-based filtering, KGUF selectively learns the most pertinent characteristics for describing items in the catalog. In other words, it effectively filters knowledge information that proves genuinely valuable for the recommendation task.

Secondly, the linear aggregation schema adeptly combines collaborative and knowledge signals, thus improving both item and user representations. Another observation is that GNN-based techniques (LightGCN, DGCF, KGIN, KGTORe) consistently outperform non-graph-based ones, underscoring the efficacy of these architectures in leveraging long-term user-item relationships. The superior performance of knowledge-aware GNN-based methods is particularly

Table 2: Performance of KGUF regarding nDCG@10 when varying the knowledge contribution. Table 2a reports the impact of the parameter α , which weights the knowledge contribution. Table 2b shows the impact of considering the knowledge or the collaborative signals only.

(a)		(b)	
Configuration	nDCG@10	Setting	nDCG@10
$KGUF_{\alpha=.2}$	0.25255	$KGUF_{w/o.knowledge}$	0.23031
$KGUF_{\alpha=.4}$	0.25640	$KGUF_{w/o.collaborative}$	0.25074
$KGUF_{\alpha=.6}$	0.25233	$KGUF_{\alpha=.4}$	0.25640
$KGUF_{\alpha=.8}$	0.25322		

noteworthy, highlighting how graph models effectively incorporate and exploit knowledge graphs.

Finally, we observe that not all knowledge-aware models (KGFlex, kaHFM) outperform pure collaborative approaches, emphasizing how merely utilizing knowledge-aware information may not necessarily lead to improved performance.

To delve deeper into assessing the efficacy of the proposed aggregation schema, we analyze how varying the weight assigned to the knowledge-aware component within the KGUF impacts the outcome on the dataset *Yahoo! Movies*. The results in Table 2a involve varying the weight of α in Eq. 6 across a spectrum from 0.2 to 0.8, i.e., increasing the contribution of knowledge on the collaborative information. The results show that searching for the best balance between content and collaborative contributions is necessary, which in this case is 40% content and 60% collaborative. However, it is even more interesting to question the collaborative and the knowledge signals’ separate impact when switched on/off. Performance in Table 2b makes it clear that removing the knowledge information affects the model capabilities harder than removing the collaborative one. This shows how KGUF benefits from semantic information and how combining the two signals gives superior performance to considering only one.

6.2 RQ2: Decision Tree Construction

In this study, we explore how the tree construction strategies can impact the overall performance of KGUF, assessing how the negative sampling augmentation for tree building and the maximum depth of the trees contribute to the overall recommendation quality.

Effect of negative sampling. We investigate how KGUF behaves when we enhance the negative sampling employed in constructing decision trees by introducing a number of negative items equal to η times the number of positive items, such that $|\tilde{\mathcal{I}}_u^-| = \eta |\mathcal{I}_u^+|$. The results depicted in Figure 2a indicate an improvement in accuracy performance with respect to the target metric as η increases.

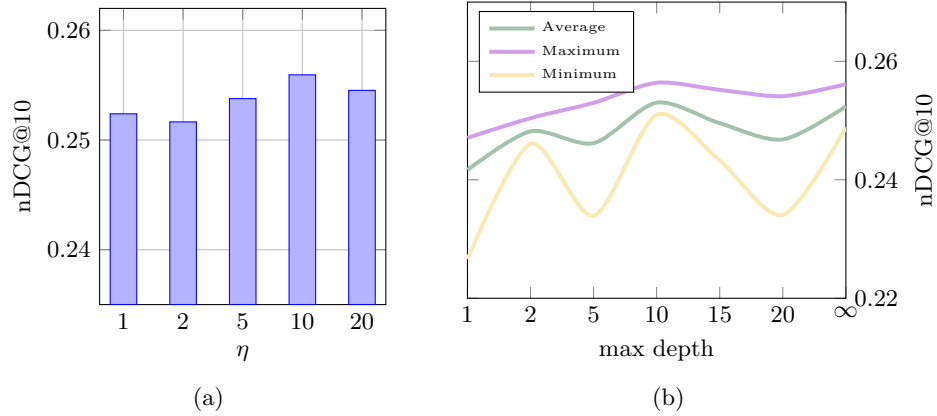


Fig. 2: An example of how KGUF models the item representation. Figure 1a shows the entities in the knowledge graph linked to the items. In figure 1b, the most relevant semantic features selected by KGUF are reported.

We attribute this trend to the higher values of η mitigating the influence of randomly selecting negative items, consequently reducing noise impact. However, it is important to note that increasing the value of η does not guarantee a constant improvement in performance. Therefore, there is no ideal value for η , and this parameter must be explored individually for each dataset.

Effect of tree depth. The maximum depth of a decision tree, representing the length of the longest path from the tree root to a leaf, directly influences the selection of features. We investigate how imposing constraints on the tree depth may impact the recommendation performance. To address this, we trained the model with diverse maximum tree depths—1, 2, 5, 10, 15, 20, and without a limit (∞). We evaluated accuracy performance repeating the experiment five times and presenting the average, minimum, and maximum values of the target metric in Figure 2b. While unlimited depth yields satisfactory results, it is noteworthy that exploring the entire tree incurs higher computational and memory costs. Excessive features can lead to an overly generalized model, and comparable or even superior performance can be achieved without traversing the entire tree, as it is evident at depth 10, where we witness the least variance around the average performance.

7 Conclusion and future work

This paper addresses the need for simplification in existing KGCF recommender systems, proposing a novel algorithm called **KGUF** (Knowledge Graph User-based Filtering). We advocate for a streamlined approach, asserting that simpler strategies have the potential to yield comparable or even superior accuracy.

In KGUF, we focus on efficiently utilizing KG side information by learning user and item embeddings through linear propagation on the user-item interaction graph. Unlike other approaches, our solution integrates latent representations for semantic features directly into item representations. Additionally, we exploit the simple but effective decision tree mechanism to select meaningful semantic features from KGs and filter out noise triples.

We conduct extensive experiments on well-known recommendation datasets, evaluating the effectiveness of our framework. Furthermore, we explore improving the decision tree capability in selecting relevant semantic features. We investigate the impact of negative sampling and tree depth constraints and evaluate their impact on the overall recommendation performance.

Bibliography

- [1] Anelli, V.W., Bellogín, A., Ferrara, A., Malitesta, D., Merra, F.A., Pomo, C., Donini, F.M., Noia, T.D.: Elliot: A comprehensive and rigorous framework for reproducible recommender systems evaluation. In: SIGIR. pp. 2405–2414. ACM (2021)
- [2] Anelli, V.W., Deldjoo, Y., Noia, T.D., Sciascio, E.D., Ferrara, A., Malitesta, D., Pomo, C.: Reshaping graph recommendation with edge graph collaborative filtering and customer reviews. In: DL4SR@CIKM. CEUR Workshop Proceedings, vol. 3317. CEUR-WS.org (2022)
- [3] Anelli, V.W., Noia, T.D., Sciascio, E.D., Ferrara, A., Mancino, A.C.M.: Sparse feature factorization for recommender systems with knowledge graphs. In: RecSys. pp. 154–165. ACM (2021)
- [4] Anelli, V.W., Noia, T.D., Sciascio, E.D., Ragone, A., Trotta, J.: How to make latent factors interpretable by feeding factorization machines with knowledge graphs. In: ISWC (1). Lecture Notes in Computer Science, vol. 11778, pp. 38–56. Springer (2019)
- [5] Anelli, V.W., Noia, T.D., Sciascio, E.D., Ragone, A., Trotta, J.: Semantic interpretation of top-n recommendations. IEEE Trans. Knowl. Data Eng. **34**(5), 2416–2428 (2022)
- [6] Cao, Y., Wang, X., He, X., Hu, Z., Chua, T.: Unifying knowledge graph learning and recommendation: Towards a better understanding of user preferences. In: Liu, L., White, R.W., Mantrach, A., Silvestri, F., McAuley, J.J., Baeza-Yates, R., Zia, L. (eds.) The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13–17, 2019. pp. 151–161. ACM (2019). <https://doi.org/10.1145/3308558.3313705>, <https://doi.org/10.1145/3308558.3313705>
- [7] Chen, L., Wu, L., Hong, R., Zhang, K., Wang, M.: Revisiting graph based collaborative filtering: A linear residual graph convolutional network approach. In: AAAI. pp. 27–34. AAAI Press (2020)
- [8] Chen, Y., Yang, Y., Wang, Y., Bai, J., Song, X., King, I.: Attentive knowledge-aware graph convolutional networks with collaborative guidance for personalized recommendation. In: ICDE. pp. 299–311. IEEE (2022)
- [9] Covington, P., Adams, J., Sargin, E.: Deep neural networks for youtube recommendations. In: Sen, S., Geyer, W., Freyne, J., Castells, P. (eds.) Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, September 15–19, 2016. pp. 191–198. ACM (2016). <https://doi.org/10.1145/2959100.2959190>, <https://doi.org/10.1145/2959100.2959190>
- [10] Du, Y., Zhu, X., Chen, L., Zheng, B., Gao, Y.: HAKG: hierarchy-aware knowledge gated network for recommendation. In: SIGIR. pp. 1390–1400. ACM (2022)
- [11] Ebesu, T., Shen, B., Fang, Y.: Collaborative memory network for recommendation systems. In: Collins-Thompson, K., Mei, Q., Davison, B.D.,

- Liu, Y., Yilmaz, E. (eds.) The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018. pp. 515–524. ACM (2018). <https://doi.org/10.1145/3209978.3209991>, <https://doi.org/10.1145/3209978.3209991>
- [12] Ferrara, A., Anelli, V.W., Mancino, A.C.M., Noia, T.D., Sciascio, E.D.: Kgfex: Efficient recommendation with sparse feature factorization and knowledge graphs. *ACM Trans. Recomm. Syst.* (apr 2023). <https://doi.org/10.1145/3588901>, <https://doi.org/10.1145/3588901>, just Accepted
 - [13] Gao, B., Liu, T., Wei, W., Wang, T., Li, H.: Semi-supervised ranking on very large graphs with rich metadata. In: Apté, C., Ghosh, J., Smyth, P. (eds.) *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Diego, CA, USA, August 21-24, 2011. pp. 96–104. ACM (2011). <https://doi.org/10.1145/2020408.2020430>, <https://doi.org/10.1145/2020408.2020430>
 - [14] de Gemmis, M., Lops, P., Musto, C., Narducci, F., Semeraro, G.: Semantics-aware content-based recommender systems. In: *Recommender Systems Handbook*, pp. 119–159. Springer (2015)
 - [15] Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feed-forward neural networks. In: *AISTATS. JMLR Proceedings*, vol. 9, pp. 249–256. JMLR.org (2010)
 - [16] Gunawardana, A., Shani, G.: Evaluating recommender systems. In: *Recommender Systems Handbook*, pp. 265–308. Springer (2015)
 - [17] Hamilton, W.L.: *Graph Representation Learning. Synthesis Lectures on Artificial Intelligence and Machine Learning*, Morgan & Claypool Publishers (2020)
 - [18] He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., Wang, M.: Lightgcn: Simplifying and powering graph convolution network for recommendation. In: Huang, J.X., Chang, Y., Cheng, X., Kamps, J., Murdock, V., Wen, J., Liu, Y. (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*. pp. 639–648. ACM (2020). <https://doi.org/10.1145/3397271.3401063>, <https://doi.org/10.1145/3397271.3401063>
 - [19] He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.: Neural collaborative filtering. In: Barrett, R., Cummings, R., Agichtein, E., Gabrilovich, E. (eds.) *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*. pp. 173–182. ACM (2017). <https://doi.org/10.1145/3038912.3052569>, <https://doi.org/10.1145/3038912.3052569>
 - [20] Huang, R., Han, C., Cui, L.: Entity-aware collaborative relation network with knowledge graph for recommendation. In: *CIKM*. pp. 3098–3102. ACM (2021)
 - [21] Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: *ICLR (Poster)*. OpenReview.net (2017)

- [22] Koren, Y.: Factor in the neighbors: Scalable and accurate collaborative filtering. *ACM Trans. Knowl. Discov. Data* **4**(1), 1:1–1:24 (2010)
- [23] Koren, Y., Bell, R.M., Volinsky, C.: Matrix factorization techniques for recommender systems. *Computer* **42**(8), 30–37 (2009)
- [24] Li, J., Xu, Z., Tang, Y., Zhao, B., Tian, H.: Deep hybrid knowledge graph embedding for top-n recommendation. In: *WISA. Lecture Notes in Computer Science*, vol. 12432, pp. 59–70. Springer (2020)
- [25] Mancino, A.C.M., Ferrara, A., Bufl, S., Malitesta, D., Noia, T.D., Sciascio, E.D.: Kgtore: Tailored recommendations through knowledge-aware GNN models. In: Zhang, J., Chen, L., Berkovsky, S., Zhang, M., Noia, T.D., Basilico, J., Pizzato, L., Song, Y. (eds.) *Proceedings of the 17th ACM Conference on Recommender Systems, RecSys 2023, Singapore, Singapore, September 18-22, 2023*. pp. 576–587. ACM (2023). <https://doi.org/10.1145/3604915.3608804>, <https://doi.org/10.1145/3604915.3608804>
- [26] Mao, K., Zhu, J., Wang, J., Dai, Q., Dong, Z., Xiao, X., He, X.: Simplex: A simple and strong baseline for collaborative filtering. In: *CIKM*. pp. 1243–1252. ACM (2021)
- [27] Mao, K., Zhu, J., Xiao, X., Lu, B., Wang, Z., He, X.: Ultragcn: Ultra simplification of graph convolutional networks for recommendation. In: *CIKM*. pp. 1253–1262. ACM (2021)
- [28] Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: BPR: bayesian personalized ranking from implicit feedback. In: Bilmes, J.A., Ng, A.Y. (eds.) *UAI 2009, Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, June 18-21, 2009*. pp. 452–461. AUAI Press (2009), https://www.auai.org/uai2009/papers/UAI2009_0139_48141db02b9f0b02bc7158819ebfa2c7.pdf
- [29] Shen, Y., Wu, Y., Zhang, Y., Shan, C., Zhang, J., Letaief, K.B., Li, D.: How powerful is graph convolution for recommendation? In: *CIKM*. pp. 1619–1629. ACM (2021)
- [30] Shuai, J., Zhang, K., Wu, L., Sun, P., Hong, R., Wang, M., Li, Y.: A review-aware graph contrastive learning framework for recommendation. In: *SIGIR*. pp. 1283–1293. ACM (2022)
- [31] Spillo, G., Musto, C., de Gemmis, M., Lops, P., Semeraro, G.: Exploiting neuro-symbolic graph embeddings based on first-order logical rules for knowledge-aware recommendations. In: *DP@AI*IA. CEUR Workshop Proceedings*, vol. 3419, pp. 1–11. CEUR-WS.org (2022)
- [32] Steck, H.: Evaluation of recommendations: rating-prediction and ranking. In: *RecSys*. pp. 213–220. ACM (2013)
- [33] Tian, C., Xie, Y., Li, Y., Yang, N., Zhao, W.X.: Learning to denoise unreliable interactions for graph collaborative filtering. In: *SIGIR*. pp. 122–132. ACM (2022)
- [34] Tiddi, I., Lécué, F., Hitzler, P. (eds.): *Knowledge Graphs for eXplainable Artificial Intelligence: Foundations, Applications and Challenges, Studies on the Semantic Web*, vol. 47. IOS Press (2020)

- [35] Togashi, R., Otani, M., Satoh, S.: Alleviating cold-start problems in recommendation through pseudo-labelling over knowledge graph. In: WSDM. pp. 931–939. ACM (2021)
- [36] Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y.: Graph attention networks. In: ICLR (Poster). OpenReview.net (2018)
- [37] Wang, H., Zhang, F., Wang, J., Zhao, M., Li, W., Xie, X., Guo, M.: Exploring high-order user preference on the knowledge graph for recommender systems. *ACM Trans. Inf. Syst.* **37**(3), 32:1–32:26 (2019)
- [38] Wang, X., He, X., Cao, Y., Liu, M., Chua, T.: KGAT: knowledge graph attention network for recommendation. In: Teredesai, A., Kumar, V., Li, Y., Rosales, R., Terzi, E., Karypis, G. (eds.) *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*. pp. 950–958. ACM (2019). <https://doi.org/10.1145/3292500.3330989>, <https://doi.org/10.1145/3292500.3330989>
- [39] Wang, X., He, X., Wang, M., Feng, F., Chua, T.: Neural graph collaborative filtering. In: SIGIR. pp. 165–174. ACM (2019)
- [40] Wang, X., Huang, T., Wang, D., Yuan, Y., Liu, Z., He, X., Chua, T.: Learning intents behind interactions with knowledge graph for recommendation. In: Leskovec, J., Grobelnik, M., Najork, M., Tang, J., Zia, L. (eds.) *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*. pp. 878–887. ACM / IW3C2 (2021). <https://doi.org/10.1145/3442381.3450133>, <https://doi.org/10.1145/3442381.3450133>
- [41] Wang, X., Jin, H., Zhang, A., He, X., Xu, T., Chua, T.: Disentangled graph collaborative filtering. In: Huang, J.X., Chang, Y., Cheng, X., Kamps, J., Murdock, V., Wen, J., Liu, Y. (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*. pp. 1001–1010. ACM (2020). <https://doi.org/10.1145/3397271.3401137>, <https://doi.org/10.1145/3397271.3401137>
- [42] Wang, Z., Lin, G., Tan, H., Chen, Q., Liu, X.: CKAN: collaborative knowledge-aware attentive network for recommender systems. In: SIGIR. pp. 219–228. ACM (2020)
- [43] Wu, J., Wang, X., Feng, F., He, X., Chen, L., Lian, J., Xie, X.: Self-supervised graph learning for recommendation. In: SIGIR. pp. 726–735. ACM (2021)
- [44] Yang, Y., Huang, C., Xia, L., Li, C.: Knowledge graph contrastive learning for recommendation. In: SIGIR. pp. 1434–1443. ACM (2022)
- [45] Ying, R., He, R., Chen, K., Eksombatchai, P., Hamilton, W.L., Leskovec, J.: Graph convolutional neural networks for web-scale recommender systems. In: Guo, Y., Farooq, F. (eds.) *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*. pp. 974–983. ACM (2018). <https://doi.org/10.1145/3219819.3219890>, <https://doi.org/10.1145/3219819.3219890>

- [46] Yu, J., Xia, X., Chen, T., Cui, L., Hung, N.Q.V., Yin, H.: Xsingcl: Towards extremely simple graph contrastive learning for recommendation. *IEEE Trans. Knowl. Data Eng.* **36**(2), 913–926 (2024)
- [47] Yuan, F., He, X., Karatzoglou, A., Zhang, L.: Parameter-efficient transfer from sequential behaviors for user modeling and recommendation. In: Huang, J.X., Chang, Y., Cheng, X., Kamps, J., Murdock, V., Wen, J., Liu, Y. (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*. pp. 1469–1478. ACM (2020). <https://doi.org/10.1145/3397271.3401156>, <https://doi.org/10.1145/3397271.3401156>
- [48] Zou, D., Wei, W., Mao, X., Wang, Z., Qiu, M., Zhu, F., Cao, X.: Multi-level cross-view contrastive learning for knowledge-aware recommender system. In: Amigó, E., Castells, P., Gonzalo, J., Carterette, B., Culpepper, J.S., Kazai, G. (eds.) *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*. pp. 1358–1368. ACM (2022). <https://doi.org/10.1145/3477495.3532025>, <https://doi.org/10.1145/3477495.3532025>
- [49] Zou, D., Wei, W., Wang, Z., Mao, X., Zhu, F., Fang, R., Chen, D.: Improving knowledge-aware recommendation with multi-level interactive contrastive learning. In: *CIKM*. pp. 2817–2826. ACM (2022)