Large Language Models-Enhanced Semantic Diffusion for User-Centric Recommendation

Xian Mo
mxian168@nxu.edu.cn
School of Information Engineering
Ningxia University
Yinchuan, Ningxia, China

Yijun Hu*
12024130831@stu.nxu.edu.cn
School of Information Engineering
Ningxia University
Yinchuan, Ningxia, China

Jun Pang
jun.pang@uni.lu
Faculty of Science, Technology and Medicine
University of Luxembourg
Esch-sur-Alzette, Luxembourg

Abstract

Recently, knowledge graphs have been utilised in recommendation systems to improve accuracy by integrating item-side auxiliary information. However, structural user-side knowledge is difficult to construct and integrate due to inherent scarcity and improper granularity. This paper introduces a graph contrastive learning with Semantic transitions-Enhanced DIffusion architecture based on Large Language Models (LLMs) for user-side knowledge-aware Recommendation (SEDIRec). Specifically, our SEDIRec first leverages LLMs to infer user interests from historical behaviors, integrating this user-side information with item-side and collaborative data to construct main views. Then, two contrastive views are generated using diffusion models with semantic transitions: one at the user-side level and the other at the item-side level. For both contrastive views, we integrate user-side or item-side information with collaborative data to generate a user-item graph. Subsequently, each user-item graph is transformed into collaborative data spaces via diffusion models for generating contrastive views. This procedure not only enhances the alignment between user/item-side information and the semantic spaces of collaborative data but also effectively eliminates noise. Extensive experiments on three datasets reveal the superiority of SEDIRec, especially for users with sparse interactions. Our code is available[1].

CCS Concepts

• **Information systems** → **Recommender systems**.

Keywords

User-centric recommendation, Large language model, Diffusion model, Contrastive learning

ACM Reference Format:
Xian Mo, Yijun Hu, and Jun Pang. 2026. Large Language Models-Enhanced Semantic Diffusion for User-Centric Recommendation. In *Proceedings of the ACM Web Conference 2026 (WWW '26), April 13–17, 2026, Dubai, United Arab Emirates.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3774904.3792148

*Corresponding authors.
[1]https://github.com/HCoder-PY/SEDIRec_WWW2026

ACM ISBN 979-8-4007-2307-0/2026/04
https://doi.org/10.1145/3774904.3792148

1 Introduction

In recent years, the rapid development of web-based applications has led to the generation of a vast and growing volume of information [9, 21]. The overwhelming online information can easily lead to information overload, thereby complicating users' efforts to locate and retrieve the target information. Recommender systems function as vital tools within web applications, enabling users to effectively traverse the extensive and intricate domain of online information. It has been effectively integrated into diverse web applications, such as e-commerce platforms [9, 30], content-sharing systems [28, 31], and applications designed for recipe recommendations [16].

Over the past few years, knowledge graphs (KG) [15] have been integrated as supplementary components within recommendation systems on the item side, leading to notable improvements in the effectiveness of recommendation models. For example, DiffKG [9] aligns knowledge-aware item semantics with collaborative relations through a diffusion model to enhance recommendation performance, while HKGR [11] proposes a hypercomplex KG-aware model that captures interactions within both the user-item graph and item-side KG in hypercomplex space to improve recommendation accuracy. However, most existing approaches merely leverage KG to provide auxiliary information on the item side, while limited efforts have been devoted to exploring the construction and integration of auxiliary information on the user side into recommendation models. The underlying reasons can be summarised as follows: (1). **Inherent scarcity**: Unlike item-side features, which vary significantly across different items—for example, "news A" may have attributes such as "has category" and "content", while "movie A" includes attributes like "genre" and "comedy"—user-side features tend to be more uniform. User meta-features typically include basic demographic information such as "gender", "age", and "nationality". This inherent scarcity presents substantial challenges in effectively structuring and integrating user-side features within graph-based recommendation models. (2). **Improper granularity**: User meta-features, including attributes like "gender", "age", and "nationality", are often too generic to capture fine-grained user characteristics, potentially causing over-smoothing effects in graph-based recommendation systems. Hence, although user preferences are crucial for recommendations, they often remain abstract and are seldom accompanied by explicit feedback, which complicates their effective extraction and application.

Fortunately, the rapid progress of large language models (LLMs) has exhibited exceptional capabilities in understanding and simulating human behavior [3]. Leveraging LLMs to interpret users' historical behavior enables the extraction of comprehensive and meaningful user interest knowledge, thereby contributing to the advancement of knowledge-enhanced recommendation systems. For example, KERL [16] integrates food KG with LLMs to deliver personalised food recommendation services, while PicCDR [24] employs LLMs to strengthen both domain-invariant and domain-specific features, facilitating effective cross-domain recommendation tasks. However, most existing research leveraging LLMs primarily focuses on either incorporating recommendation tasks with dialogue systems or utilising natural language features generated by LLMs as enriched semantic information for collaborative data or item-side representations, while neglecting the construction of user-side knowledge. From the preceding discussion, it can be concluded that *effectively prompting LLMs to generate and construct structural knowledge on the user side, and integrating this with knowledge-based recommendation models, remains an open challenge.*

Furthermore, knowledge-based recommendation models mainly adopt Graph Neural Networks (GNNs) [22] to capture high-order relationships on the user-item graph and have exhibited excellent performance. However, most existing knowledge-based recommendation models rely heavily on obtaining sufficient high-quality labeled data for supervised training. In practical recommendation scenarios, network-labelled data typically faces the issue of sparsity, which hampers the effectiveness of generating precise embeddings capable of reflecting intricate user preferences. To mitigate the problem of data sparsity, some studies have applied contrastive learning (CL) [35] by graph augmentation techniques to generate self-supervised training signals, which can generate self-supervised signals from unlabelled data and demonstrate outstanding effectiveness in recommendation tasks. For example, CoGCL[36] improves graph-based contrastive learning by generating contrastive views that incorporate enhanced collaborative signals through discrete codes. Subsequently, contrastive learning with diffusion augmentation has achieved notable success in generating robust contrastive views. This approach is based on the assumption that the original user-item graph is governed by an unknown probability distribution, and it seeks to approximate this distribution using a neural network in order to reconstruct the graph and enable adaptive data augmentation. For example, InDiRec [17] introduces an intent-aware diffusion mechanism integrated with contrastive learning to enhance sequential recommendation performance.

Unfortunately, content (e.g., user-side knowledge) generated by LLMs may include unintended noise due to the hallucination issue [5], which is irrelevant to user interests. After graph convolution operations on user-item graphs, the representations of both users and items are implicitly influenced by this noisy information. As a result, augmented contrastive views by diffusion models may be contaminated with unintended noise, leading to inaccurate self-supervised signals and undermining recommendation performance. Moreover, user-side knowledge generated by LLMs offers rich and meaningful semantic information, thereby enhancing the understanding of user preferences. However, user-side knowledge predominantly exists in natural language form, and the effective transfer of information from the user-side knowledge domain to the recommendation domain remains underexplored. More recently, CIKGRec [5] designs a cross-domain contrastive learning module devised to maximise the mutual information between the auxiliary information domain and the recommendation domain. However, most existing methods rely on manually designed modules for simple alignment and exhibit limited effectiveness in facilitating semantic transitions within contrastive views. From the above discussion, it is obvious that *constructing a diffusion model that can generate contrastive views with semantic transitions while effectively reducing noise generated by LLMs remains another significant challenge.*

In light of the two challenges identified above, our work proposes a graph contrastive learning with Semantic transitions-Enhanced DIffusion architecture based on Large Language Models (LLMs) for user-side knowledge-aware Recommendation (SEDIRec), as displayed in Figure 1. Specifically, SEDIRec first utilises LLMs to infer user interests from historical behaviors and formalises user interests into a structured knowledge format, integrating this user-side information with item-side and collaborative data to construct main views. Thus, both user-side and item-side knowledge are incorporated as supplementary components into collaborative relations, thus enriching collaborative data and improving knowledge-aware recommendation performance.

To address the noise issue, we then design a knowledge-aware graph diffusion model with semantic transitions to construct two contrastive views: one at the user-side level and the other at the item-side level. In particular, we combine user-side or item-side features with collaborative information to create a user-item interaction graph for both contrastive views. Subsequently, each user-item graph is transformed into the corresponding collaborative data space through diffusion models to generate contrastive views, thereby enhancing the alignment between user/item-side information and collaborative data spaces. More specifically, our approach coordinates the forward and backward diffusion processes, wherein each user-item graph functions independently as the diffusion source, while collaborative data supplies denoising training signals, thereby facilitating the construction of contrastive views. In this way, each user-item graph is mapped into the corresponding collaborative data space through diffusion models. This facilitates the precise modeling of semantic transitions from collaborative data to each user-item graph (i.e., from recommendation domains to the user/item-side knowledge domains), thereby enabling the accurate extraction of task-relevant information for the generation of contrastive views. Furthermore, during the transformation of the user-item graph into collaborative data spaces, the proposed method effectively mitigates noise in user/item-side features. Finally, we contrast the augmented view representations with the main-view representations to update model parameters and improve recommendation performance.

In summary, this paper makes the following contributions:

(1). We develop an LLMs-based user-side knowledge construction method that improves knowledge-based recommendation models by integrating user-side interest knowledge. (2). We design a knowledge-aware graph diffusion model with semantic transitions to construct two contrastive views, effectively addressing the challenges of noise and semantic transitions that arise when integrating LLMs-generated user-side information into the recommendation domain. (3). Comprehensive experiments conducted on three datasets reveal the effectiveness of our SEDIRec compared to various state-of-the-art approaches. Notably, on the Book-Crossing dataset, SEDIRec showed a significant improvement, with 9.78% higher Recall@50 and 15.64% higher NDCG@50.

2 Related Work

KG-based recommendations. In recent years, KGs have been increasingly incorporated into recommendation systems as auxiliary elements, enhancing the performance and accuracy of recommendation models. KGAT [25] proposes a knowledge graph attention network that explicitly captures high-order connectivities within KG to enhance recommendation performance, while KGIN [26] proposes a KG-based intent network that models each intent as an attention-based combination of KG relations, promoting independence among intents to enhance model performance. More recently, HKGR [11] proposes a hypercomplex KG-aware model that captures interactions in both the user-item graph and item-side KG to enhance recommendation accuracy. Most existing KG-based recommendation models require abundant labelled data for supervised training. However, labelled data in real-world scenarios often suffer from sparsity. To mitigate this issue, some researchers apply contrastive learning to generate self-supervised signals for KG-based recommendation. KGCL [33] proposes a contrastive learning framework for KG-enhanced recommender systems to reduce information noise, while MCCLK [37] introduces a multi-level cross-view contrastive learning mechanism designed to enhance knowledge-aware recommendations. KGRec [32] later proposes an attention-based mechanism to score knowledge triplets via contrastive self-supervision to improve recommendation performance. Recently, Diffusion models have shown significant success in generating effective contrastive views for recommendation systems. DiffKG [9] leverages a diffusion model to align knowledge-aware item semantics with collaborative relations, thereby improving recommendation performance, while DiffMM [8] integrates cross-modal contrastive learning with a modality-aware graph diffusion model to improve knowledge-aware recommendation performance. However, most existing approaches only use KG to provide auxiliary item-side information, with little attention given to integrating auxiliary user-side features into recommendation models.

LLMs-based recommendations. In recent years, the rapid advancement of LLMs has drawn significant attention due to their promising applications in recommendation tasks. Specifically, LLMs serve two main functions in recommendation tasks [5]: acting as recommenders or enhancers. As recommenders, LLMs incorporate recommendation tasks into conversational settings by tokenizing item and user identifiers. For example, LLM4POI [12] utilises pretrained LLMs for next point-of-interest recommendations, while IFairLRS [7] investigates the item-side fairness properties of LLMs-based recommendation systems. More recently, KERL [16] combines a food KG with LLMs to provide customized food recommendation services, while RTA [3] aims to enhance the efficiency of conversational recommendation systems by compressing multi-token item titles into single-token representations within LLMs. However, these models struggle to capture the collaborative signals [10], which limits their advantages compared to traditional collaborative filtering methods. Some studies have explored improving conventional recommendation algorithms by integrating features derived from LLMs. For example, SeRALM [19] combines LLMs with traditional ID-based sequential recommender approaches to enhance sequential recommendation performance, while LLMRec [27] improves recommendation systems through the application of three straightforward but effective LLMs-based graph augmentation techniques. PicCDR [24] later utilises LLMs to strengthen both domain-invariant and domain-specific features, thereby supporting more effective cross-domain recommendation tasks. However, most existing approaches fail to incorporate auxiliary user-side features into the recommendation framework by LLMs, resulting in limited recommendation performance.

3 Problem Definition

A user-item graph is formally defined as $G = (U, V, Y)$, where $U = \{u_1, \ldots, u_i, \ldots, u_I\}$ with cardinality $|U| = I$ represents the set of users, and $V = \{v_1, \ldots, v_j, \ldots, v_J\}$ with cardinality $|V| = J$ denotes the set of items. $Y = [y_{i,j}]_{I\times J} \in \{0, 1\}$ represents a user u_i has interacted with an item v_j. If $y_{ij} = 1$, it indicates there exists an interaction between user u_i and item v_j, and $y_{ij} = 0$ otherwise. To enhance graph G with user/item-side information, we first leverage LLMs to infer user interests to build a user–entity graph $G_{uo} = \{(u, r_{uo}, o)\}$. $o \in O$ represents various types of entities within users, and O denotes the set of all entities. If $r_{uo} = 1$, it describes there exists an interaction between user u and entity o, and $r_{uo} = 0$ otherwise. Then, we adopt the existing item-side information to construct an item-entity graph $G_{vs} = \{(v, r_{vs}, s)\}$. $s \in S$ represents various types of entities within items, and S denotes the set of all entities. If $r_{vs} = 1$, it describes there exists an interaction between item v and entity s, and $r_{vs} = 0$ otherwise.

As discussed in Section 1, a key challenge lies in prompting LLMs to effectively generate and construct structured user-side knowledge, as well as in designing a diffusion model capable of generating contrastive views with semantic transitions while simultaneously mitigating the noise introduced by LLMs. To tackle this challenge, we formulate the recommendation task as follows:

Given a graph $G = (U, V, Y)$, we first employ LLMs to infer user interests from historical behaviors and formalise user interests into a structured knowledge format G_{uo}, integrating this user-side information with item-side information G_{vs} and collaborative data G to construct the main views G_t. To address the noise issue, we then design a knowledge-aware graph diffusion model with semantic transitions to construct contrastive views at both the user-side level G_{c1} and the item-side level G_{c2}. The recommendation task aims to predict unobserved interactions between users and items, denoted as y_{ij}, using the corresponding encoded representations, which are obtained by the input of $\{G_t, G_{c1}, G_{c2}\}$ based on a knowledge-aware contrastive learning paradigm.

4 Methodology

In this section, we present our proposed SEDIRec model, which is depicted in Figure 1. Overall, the SEDIRec model comprises two components: an LLMs-based user-side KG construction and a knowledge-aware graph diffusion model with semantic transitions for recommendations.

4.1 LLMs-based User-side KG construction

In this section, we provide a detailed description of LLM-based user-side knowledge generation and how to transform such knowledge from natural language into structured user-side information.

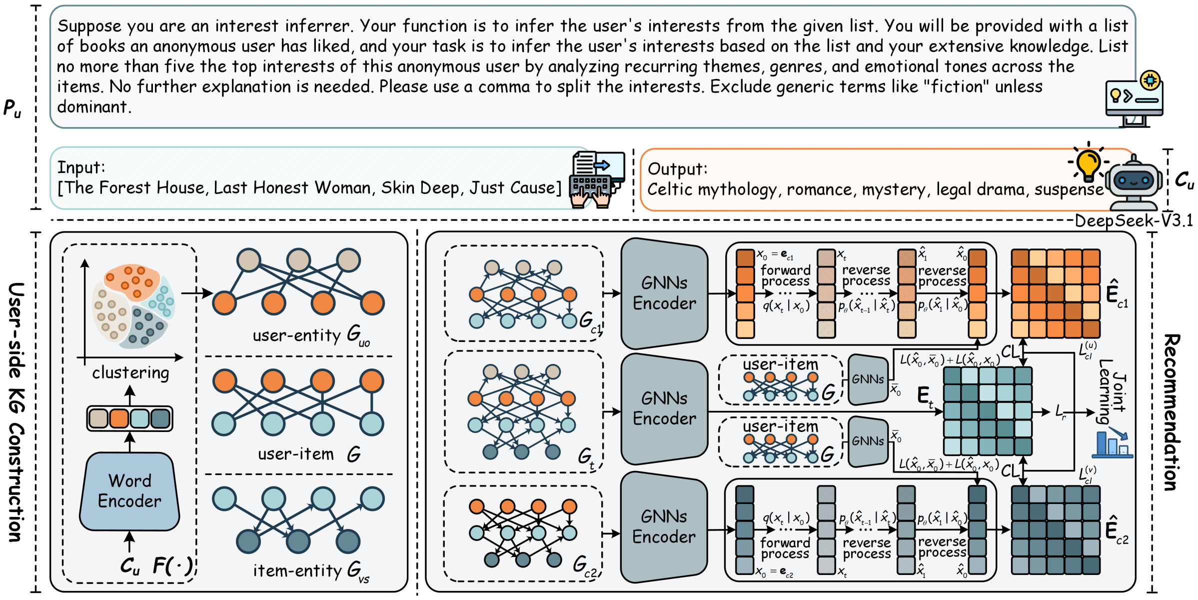

Figure 1: Overall architecture of SEDIRec. The upper half displays LLMs-based user interest C_u inference, the lower left half displays LLM-based user-side KG construction, and the lower right half introduces a knowledge-aware graph diffusion model with semantic transitions for recommendations.

User Interest Inference: As discussed in Section 1, accurately capturing and representing user preferences plays a critical role in enhancing the performance of recommendation systems. However, the acquisition and representation of user-side knowledge presents significant challenges due to inherent scarcity and improper granularity. Motivated by the strong capability of LLMs in comprehending user behavior [27], it becomes feasible to explicitly model user interests. Therefore, we utilise LLMs to infer and generate user interests by carefully devising prompts based on user behaviors extracted from the dataset, enabling the discovery of interests not initially contained within the dataset. Using the extensive knowledge and reasoning capabilities of LLMs, the model is capable of extracting abstract, high-level user interests through the analysis of item relationships derived from user interactions, instead of depending on conventional user-side metadata features. A concrete example is illustrated in Figure 1, and the LLMs-based user interest inference is formally defined in Formula 1:

$$C = \{C_1, C_2, \ldots, C_{|I|}\}, \quad C_u = LLMs(P_u, G) \tag{1}$$

where P_u denotes the textual prompt associated with the user u, and C denotes the textual information generated to represent user interests. Thus, all textual information C related to user interests is derived by employing LLMs.

User Interest Knowledge Structuration: Structuring user interest knowledge aims to capture user-side multi-hop information and to ensure vector space consistency. Specifically, representing user interests in a structured format enables the model to effectively not only capture complex, higher-order relationships among users with aligned preferences, but also achieve a more coherent alignment between semantic signals and collaborative signals. Technically, we first employ classic sentence embedding techniques, specifically fastText [1], to encode semantic information into dense vector representations. Then, the text clustering method HDBSCAN [14] is utilized to merge similar interests. In particular, we initially define a minimum sample threshold a for clustering and a threshold b for the number of adjacent points around a point. Only when this value is exceeded is the point considered qualified to be a core point for clustering. Subsequently, we use the Euclidean distance to calculate the pairwise distances between the samples for clustering. This approach ensures a balance between cluster density and size, thereby preventing erroneous clustering and mitigating semantic inconsistencies that may arise when the same interests are expressed differently due to the stochastic nature of LLM outputs. Finally, we establish connections between the merged interests and users to construct a user interest knowledge graph G_{uo}, which is formally defined in Formula 2:

$$G_{uo} = \{(u, r_{uo}, o)\}, \quad O = \{o_1, o_2, \ldots, o_k\} = F(C, a, b) \tag{2}$$

where $F(\cdot)$ represents the word encoding and clustering method. If $r_{uo} = 1$, it indicates that there is an interaction between user u and entity o, and $r_{uo} = 0$ otherwise.

To enhance users' semantic information, we integrate KG G_{uo} into users U within the user-item graph G. Specifically, we use a relational knowledge embedding layer [4] to aggregate various types of entity o within the KG into user representations U, thus enriching the semantic information associated with users. We also use the relation-sensitive knowledge embedding layer to incorporate KG G_{vs} into the item representations V within the user-item graph G, so that we can improve the semantic information associated with the items. Both user-side and item-side knowledge are integrated as supplementary elements into collaborative relations, thereby enriching collaborative data and serving as the main view

G_t. Subsequently, we utilise a GNNs-based encoder [22] to process the main view G_t and obtain the main view embeddings $e \in \mathbf{E}_t$.

4.2 Knowledge-aware Graph Diffusion Model

As discussed in Section 1, achieving a robust contrastive view with semantic transitions while effectively reducing noise generated by LLMs remains a significant challenge. Motivated by the effectiveness of diffusion models [8] in preserving essential data patterns within their generated outputs, our SEDIRec model designs a knowledge-aware graph diffusion model with semantic transitions to construct two contrastive views: one at the user-side level and the other at the item-side level. Particularly, the user-side KG G_{uo} and the item-side KG G_{vs} are separately integrated with the collaborative graph G to construct a user-side level user-item graph G_{c1} and an item-side level user-item graph G_{c2}. Our knowledge-aware semantic transition strategy aims to maximise the mutual information between the user/item-side level user-item graph G_{c1}/G_{c2} and the user-item graph G through diffusion models, this facilitates the transfer of information from the user/item-side knowledge domains to the recommendation domains while effectively reducing noise.

Diffusion Process: We execute the reverse process to reconstruct the user-item graph G_{c1}/G_{c2}, thus identifying task-relevant information for contrastive view generation. We corrupt the representations, denoted as $x_0 \in \mathbf{X}_0 = \mathbf{E}_{c1}/\mathbf{E}_{c2}$, within the original graph G_{c1}/G_{c2} by introducing noise during the diffusion phase, as described in Formula 3. Here, $x_0 = e_{c1}/e_{c2} \in \mathbf{E}_{c1}/\mathbf{E}_{c2}$ is acquired via the GNNs-based encoding process on user-item graph G_{c1}/G_{c2}.

$$q(x_t|x_0) = N(x_t; \sqrt{\bar{a}_t}x_0, (1-\bar{a}_t)\mathbf{I}), \quad \bar{a}_t = \prod_{t'=1}^{t}(1-\beta_{t'}) \tag{3}$$

where $t \in \{1, \ldots, T\}$ indicates the diffusion step, N denotes the Gaussian distribution, $\mathbf{I}$ is the identity matrix, and $\beta_t \in (0, 1)$ determines the scale of Gaussian noise introduced at each step t. Eventually, the state x_T converges to a standard Gaussian distribution as $T \to \infty$.

Reverse Process: In the reverse process, our aim is to iteratively recover the representations x_0 from the pure Gaussian noise x_T. The diffusion model is designed to learn the denoising process, allowing it to recover x_{t-1} from x_t through neural networks, as illustrated in Formula 4:

$$p_\theta(x_{t-1}|x_t) = N(x_{t-1}; u_\theta(x_t, t), \Sigma_\theta(x_t, t)), \tag{4}$$

where $u_\theta(x_t, t)$ and $\Sigma_\theta(x_t, t)$ represent the mean and covariance of the Gaussian distribution, which can be derived using neural networks with parameters θ. By reparameterising the mean u_θ [9], as shown in Formula 5, the model is able to learn the noise introduced in the time step t:

$$u_\theta(x_t, t) = \frac{1}{\sqrt{\bar{a}_t}}(x_t - \frac{\beta_t}{\sqrt{1-\bar{a}_t}}\epsilon_\theta(x_t, t)) \tag{5}$$

where $\epsilon_\theta(x_t, t)$ is implemented using a Multi-Layer Perceptron (MLP) [20], which computes the predicted value $\hat{x_t}$ from the input x_t and the corresponding time step t. To recover embeddings $\hat{x}_0$, we use the embedding $\bar{x}_0 \in \mathbf{E}$ of the collaborative graph G and the embedding $x_0 \in \mathbf{E}_{c1}/\mathbf{E}_{c2}$ as the evidence lower bound (ELBO) [9] with sample time step t from $\{1, 2, \ldots, T\}$ to update the model parameters, as presented in Formula 6. In this case, $\mathbf{E}$ is derived similarly using the GNNs-based encoding procedure on the collaborative graph G.

$$\begin{aligned} L_{elbo} = E_{t\sim U(1,T)}(E_{q(x_0)}[||\epsilon_\theta(x_t, t) - x_0||_2^2] + \\ E_{q(\bar{x}_0)}[||\epsilon_\theta(x_t, t) - \bar{x}_0||_2^2]) \end{aligned} \tag{6}$$

Equation 6 includes two components: a reconstruction loss term defined as $||\epsilon_\theta(x_t, t) - x_0||_2^2$ and a semantic transition loss term defined as $||\epsilon_\theta(x_t, t) - \bar{x}_0||_2^2$. Thus, our model enables $\hat{x}_0$ to approximate both x_0 and $\bar{x}_0$, which helps to update the model parameters. In this manner, user/item-side knowledge embeddings x_0 are transformed into collaborative data spaces $\bar{x}_0$ via diffusion models. This transformation allows for accurate modeling of semantic transitions between recommendation domains and user/item-side knowledge domains, which in turn supports the effective extraction of task-relevant information for generating contrastive views. Additionally, when transforming user/item-side knowledge embeddings x_0 into collaborative data representations $\bar{x}_0$, the noise present in the user / item knowledge is efficiently removed. After acquiring the reconstructed $\hat{x}_0$, we use it to generate the reconstructed contrastive view $\hat{G}_{c1}/\hat{G}_{c2}$ along with the associated embeddings $\hat{\mathbf{e}}_{c1} \in \hat{\mathbf{E}}_{c1}$ and $\hat{\mathbf{e}}_{c2} \in \hat{\mathbf{E}}_{c2}$.

4.3 Model Optimisation

We employ the InfoNCE loss [23] to contrast augmented-view embeddings with main-view embeddings for training the model parameters. The contrastive loss for users U is formulated as Formula 7:

$$L_{cl}^{(u)} = \sum_{m\in\{1,2\}}\sum_{u\in U} -\log\frac{\exp((s(\mathbf{e}^{(\mathbf{u})}, \hat{\mathbf{e}}_{\mathbf{cm}}^{(\mathbf{u})})/\tau)}{\sum_{v\in U}\exp((s(\mathbf{e}^{(\mathbf{u})}, \hat{\mathbf{e}}_{\mathbf{cm}}^{(\mathbf{v})})/\tau)}, \tag{7}$$

where τ and $s(\cdot)$ represent the temperature parameter and the cosine similarity, respectively. The pairs $(\mathbf{e}^{(\mathbf{u})}, \hat{\mathbf{e}}_{\mathbf{cm}}^{(\mathbf{u})})$ represent the same nodes in the main view and two contrastive views $\hat{G}_{c1}$ and $\hat{G}_{c2}$ as positive pairs, while $(\mathbf{e}^{(\mathbf{u})}, \hat{\mathbf{e}}_{\mathbf{cm}}^{(\mathbf{v})})$ $(v \in U)$ denote any two distinct nodes in the main view and two contrastive views as negative pairs. This approach enables the model parameters to be optimised by enhancing the similarity between positive pairs while declining the similarity between negative pairs, thus effectively addressing the challenge of data sparsity. The contrastive loss $L_{cl}^{(v)}$ for item V is defined in an analogous way. The main objective function is optimised together with the contrastive loss to train the model parameters, as described in Formula 8.

$$L = L_r + \theta_1(L_{cl}^{(u)} + L_{cl}^{(v)}) + \theta_2 \cdot ||\Theta||_2^2, \tag{8}$$

The contribution of the model parameters Θ is controlled by θ_2, while θ_1 is used to adjust the influence of contrast loss. We denote the main objective function as L_r, which is formulated in Formula 9. The value $y_{u,i}$ represents the predicted score for the positive item v corresponding to user u, while $y_{u,j}$ signifies the predicted score for the negative item v associated with user u.

$$L_r = \sum_{(u,i,j)\in O} -log(y_{u,i} - y_{u,j}) \tag{9}$$

Complexity Analysis: Let d represent the feature dimension, L the number of GNN layers, $|Y|$ the number of edges in user-item graphs, I and J the number of nodes in sets U and V, B the batch size, T the number of diffusion steps, k the number of interests, and $|o|$ the average number of words per interest. The computational complexity of the GNN-based encoding process with L layers is $O(L \times |Y| \times d)$. The diffusion model used to generate the contrastive views $\hat{\mathbf{G}}_{c1/c2}$ has a complexity of approximately $O(|Y| \times d^2 \times T)$. The contrastive learning framework requires $O(B \times L \times (I + J) \times d)$, while the user KG construction requires $O(k \times |o| + |o|^2)$.

5 Evaluation

5.1 Experimental Settings

Datasets: We conduct our experiments on three widely used real-world recommendation datasets from online applications—namely, DBbook2014, Book-Crossing, and MovieLens-1M—all of which include item-side auxiliary knowledge. In particular, **DBbook2014** and **Book-Crossing** dataset are used for book recommendation and contain both explicit ratings and implicit feedback from users on books, while **MovieLens-1M** dataset is used for movie recommendations and includes information such as movie titles, release years, genres, and user attributes. Following previous studies [5, 6], we filter out users with fewer than 10 interactions in MovieLens-1M and fewer than 5 in DBbook2014 and Book-Crossing. Dataset statistics are presented in Table 1.

Table 1: Statistics of Three Datasets

Dataset	MovieLens-1M	DBbook2014	Book-Crossing
User	6,040	5,576	6,616
Item	3,260	2,680	8,853
Interactions	998,539	65,961	110,662
Relations	20	13	4
Entities	14,377	8,762	1,404
Triplets	415,104	134,223	1,137

Baselines: To evaluate the effectiveness of our SEDIRec model, we select various baseline methods, including GNNs-based recommendation methods and GNNs-based knowledge-aware recommendation models. **GNNs-based recommendation methods: LightGCN** [2] designs a simplified GNNs to improve conciseness and suitability for recommendations. **SGL** [29] explores self-supervised learning to enhance the robustness of GNNs for recommendations. **SimGCL** [34] presents a simple CL-based GNNs model for recommendations. **GNNs-based knowledge-aware recommendation models: KGAT** [25] presents a knowledge graph attention network within item-side KG for recommendations. **KGIN** [26] introduces a KG-based intent network to enhance model performance. **KGCL** [33] presents a CL framework for KG-enhanced recommendation. **MCCLK** [37] proposes a multi-level cross-view CL approach to improve knowledge-aware recommendations. **KGRec** [32] proposes a CL-based method to score knowledge triples for recommendations. **DiffKG** [9] employs a diffusion model to align knowledge-aware item semantics with collaborative relations for recommendations. **DiffMM** [8] utilise a CL-based graph diffusion model for knowledge-aware recommendations. **CIKGRec** [5] designs an LLM-based user-side knowledge recommendation.

Parameter Settings: We set the learning rate searched from 0.001 to 0.005, the number of layers L in GNNs is selected from the set $\{2, 4, 6, 8, 10\}$, the minimum sample threshold to $a = 5$, and the number of adjacent points to $b = 2$. The L_2 regularisation decay term θ_2 is selected from the set $\{1e^{-1}, 1e^{-2}, 1e^{-3}, 1e^{-4}, 1e^{-5}\}$; the parameter θ_1 is searched from $\{0.01, 0.02, 0.03, 0.04\}$; the number of steps is fixed at $t = 200$; and the temperature τ is searched across $\{0.1, 0.2, 0.3\}$. The baseline parameters are carefully adjusted to ensure fair comparative analysis and "DeepSeek-V 3.1" is adopt as LLMs. The NDCG@N and Recall@N metrics [5], with default values of $N = 50/100$, are used to evaluate our proposed model (abbreviated as ND@N and R@N). We report average performance based on 10 experimental runs. The experiments are conducted on an Ubuntu 20.04 system equipped with an NVIDIA Corporation Device A100 GPU, and Python 3.8.

5.2 Recommendation Performance

Table 2 shows that our model SEDIRec outperforms other baselines on three real-world recommendation datasets. LightGCN utilises a GNNs-based framework to generate node embeddings, whereas CL-based methods, such as SGL and SimGCL, effectively improve the recommendation performance of GNNs. However, these approaches ignored the integration of user- and item-side information in enhancing knowledge-aware recommendations, leading to suboptimal performance. Some models, including MCCLK and KGRec, incorporate item-side information into recommendation systems, thereby enhancing recommendation performance. We find that these baseline methods are highly dependent on the presence of item-side knowledge. Specifically, these approaches achieve good performance on datasets that contain rich item-side knowledge, such as DBbook2014 and MovieLens-1M, but show degraded performance on datasets with limited item-side information, like Book-Crossing. Notably, SEDIRec showed significant improvement on the Book-Crossing dataset, with 9.78% higher Recall@50 and 15.64% higher NDCG@50. DiffKG constructs an extra knowledge graph to enhance items' semantic information using diffusion models, demonstrating satisfactory performance in sparse item-side information on Book-Crossing. However, this model focuses exclusively on item-side information and overlooks user-side features. CIKGRec designs an LLM-based user-side knowledge recommendation framework with simplified alignment modules, which exhibits limited effectiveness in addressing semantic transitions and mitigating LLM-generated noise. Our SEDIRec model outperforms all baseline models and offers the following key advantages, i.e., SEDIRec is able to tackle noise and semantic transition issues in integrating LLMs-generated user data into recommendations, thanks to the proposed LLM-based method to build user-side knowledge and design a knowledge-aware graph diffusion model with semantic transitions to generate contrastive views.

5.3 Ablation Study

This section conducts an ablation study assessing each module's contribution, along with ten experiments reporting the average Recall@50 and NDCG@50 scores on three datasets. Table 3 clearly shows a significant performance drop across all three datasets when

Table 2: Recommendation Performance on three datasets

Datasets	DBbook2014				Book-Crossing				MovieLens-1M			
	R@50	R@100	ND@50	ND@100	R@50	R@100	ND@50	ND@100	R@50	R@50	ND@50	ND@100
SGL	0.4258	0.5249	0.2350	0.2567	0.1613	0.2154	0.0860	0.0982	0.4094	0.5505	0.4004	0.4433
LightGCN	0.4214	0.5358	0.2230	0.2481	0.1607	0.2246	0.0765	0.0911	0.4086	0.5574	0.3977	0.4432
SimGCL	0.4280	0.5369	0.2376	0.2610	0.1653	0.2211	0.0860	0.0988	0.4131	0.5560	0.4048	0.4482
KGIN	0.4147	0.5337	0.2200	0.2462	0.1468	0.2075	0.0681	0.0819	0.4095	0.5581	0.4004	0.4453
KGAT	0.4174	0.5309	0.2110	0.2360	0.1569	0.2133	0.0865	0.1004	0.4059	0.5527	0.3928	0.4378
MCCLK	0.4376	0.5491	0.2054	0.2273	0.1476	0.2039	0.0656	0.0774	0.4176	0.5655	0.3876	0.4339
KGCL	0.4308	0.5271	0.2453	0.2666	0.1562	0.2049	0.0859	0.0971	0.4083	0.5496	0.3996	0.4430
KGRec	0.4415	0.5497	0.2373	0.2611	0.1473	0.2075	0.0699	0.0836	0.4136	0.5633	0.4072	0.4524
DiffKG	0.3554	0.4384	0.1987	0.2168	0.1104	0.1475	0.0624	0.0717	0.3442	0.4756	0.3386	0.3789
DiffMM	0.4194	0.5396	0.2130	0.2384	0.1493	0.2187	0.0670	0.0827	0.2985	0.4336	0.3002	0.3408
CIKGRec	0.4642	0.5756	0.2494	0.2739	0.1791	0.2455	0.0889	0.1041	0.4250	0.5718	0.4162	0.4609
SEDIRec	**0.4723**	**0.5837**	**0.2569**	**0.2813**	**0.1966**	**0.2601**	**0.1028**	**0.1176**	**0.4384**	**0.5821**	**0.4329**	**0.4756**
%Improve	1.74%	1.41%	3.01%	2.70%	9.78%	5.75%	15.64%	12.97%	3.15%	1.80%	4.01%	3.19%

Table 3: Ablation Study on Key Components

Dataset	DBbook2014		Book-Crossing		MovieLens-1M	
Variants	R@50	ND@50	R@50	ND@50	R@50	ND@50
w/o UIK	0.4196	0.2149	0.1742	0.0963	0.3929	0.3587
w/o CL	0.4705	0.2536	0.1920	0.0994	0.4239	0.4141
w/o DM	0.4678	0.2562	0.1952	0.1025	0.1503	0.1416
w/o ST	0.4646	0.2518	0.1965	**0.1028**	0.4221	0.4182
Ours	**0.4723**	**0.2569**	**0.1966**	**0.1028**	**0.4384**	**0.4329**

user interest knowledge (UIK) is excluded. This highlights the critical role of LLMs-generated structured knowledge on the user side. Similarly, removing contrastive learning (CL) and substituting the diffusion model (DM) with a denoising autoencoder [13] results in reduced performance. Experimental results show that contrastive learning effectively addresses data sparsity, while diffusion models enhance model performance by enabling precise extraction of task-relevant information to generate contrastive views. Furthermore, eliminating the semantic transition loss term (ST) from L_{elbo} also leads to a decrease in most metric performance. Experimental results demonstrate that this module enables accurate modeling of semantic transitions between recommendation domains and user/item-side knowledge domains, while significantly reducing noise from integrating LLM-generated user information into recommendations.

5.4 Sparsity Analysis

We evaluate the effectiveness of SEDIRec in addressing user-side data sparsity, a challenge stemming from the difficulty of learning optimal representations for inactive users with limited interactions. Following prior research [25], we classify users into four groups based on their historical interactions in the training set and report the average NDCG@50 results for each group using our model. Figure 2 shows the results, with the horizontal axis indicating user interaction ranging from sparse to dense. Our model outperforms the baseline methods for user groups with sparse interactions across all three datasets, demonstrating its effectiveness in addressing interaction sparsity. This is mainly due to our LLM-based approach to construct user-side knowledge by incorporating user information and facilitating collaborative information mining, which effectively improves performance for users with sparse interactions.

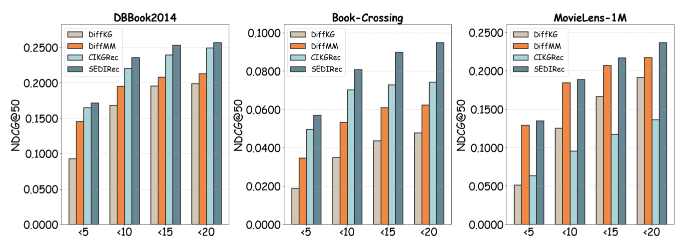

Figure 2: User-side Data Sparsity Analysis

Table 4: Robustness Analysis on DBbook2014 dataset

Noise Ratios	10%		30%		50%	
	R@50	ND@50	R@50	ND@50	R@50	ND@50
DiffKG	0.3040	0.1676	0.2063	0.1089	0.1030	0.0533
DiffMM	0.3873	0.1957	0.3066	0.1523	0.2126	0.0937
CIKGRec	0.4159	0.2165	0.3429	0.1736	0.2715	0.1294
Ours	**0.4364**	**0.2353**	**0.3673**	**0.1905**	**0.2900**	**0.1417**

5.5 Robustness Analysis

We design a knowledge-aware graph diffusion model with semantic transitions to reduce noise. To validate its effectiveness, we estimate the percentage of performance degradation under varying noise ratios. Specifically, 10%, 30%, and 50% of the edges in the user-item graph G are randomly replaced by noise signals. We evaluate SEDIRec, DiffKG, DiffMM, and CIKGRec on the DBbook2014 dataset using Recall@50 and NDCG@50. The results in Table 4 show that our model SEDIRec has a significantly smaller performance drop than the baseline models. These results demonstrate the strong denoising effectiveness of SEDIRec in overcoming noise challenges during the integration of LLMs-generated user-side information into recommendation systems.

5.6 Hyperparameter Analysis

We analyse key hyperparameters, with results shown in Figure 3. We evaluate how changes in the number of diffusion steps t, temperature τ, parameter θ_1, and GNNs layers L affect the recommendation performance. We conducted 10 experiments to report average Recall@50 and NDCG@50 across three datasets and summarize the following observations: (1) As shown in the analysis of t, we achieve satisfactory results when $t = 200$. Beyond this point, performance increases or remains stable. Considering the trade-off between computational cost and performance, we set $t = 200$. (2) For τ, the model achieves peak performance when it is within the range $\{0.1, 0.2, 0.3\}$. As τ increases beyond this range, performance decreases. (3) Satisfactory performance is achieved at different L in different datasets, while a large L may cause overfitting and degrade performance. (4) For θ_1, which control the loss weights, experimental results show that satisfactory performance is achieved when $\theta_1 = 0.01/0.35$ across datasets. As θ_1 increases, performance decreases. This may occur because larger values cause excessive focus on contrastive learning, distracting from the main task and degrading performance.

5.7 Scalability Analysis

LLMs-based and diffusion-based models often encounter undesirable time consumption. Furthermore, graph contrastive learning models typically involve high computational costs because they require constructing additional views. The Book-Crossing dataset includes more users than other datasets, making it suitable for scalability analysis. We evaluate the scalability of our model on the Book-Crossing dataset by setting the number of users to 500, 1000, 2000, 4000, and all nodes. Experiments show that SEDIRec converges in an average of 19, 30, 69, 133, and 171 minutes under different user settings. Thus, we conclude that the SEDIRec model linearly increases computational cost as the number of users grows, making it suitable for large-scale networks.

5.8 Case Study

To intuitively evaluate user-side knowledge effectiveness, we select user #u1968 from the Book-Crossing dataset, who has the sparsest user-item interactions and item-side knowledge among the three datasets.As shown in Figure 4, the left panel presents the LLMs-based interest knowledge generation process for user #u1968 and the top-K recommendation list, where higher-ranked items are more likely to be interacted with by the user. The right panel shows how user similarity is captured based on the shared interest #c9831 between user #u1968 and user #u6140 using our recommendation model. Item #v2534, a positively reviewed Christian apocalyptic ficition, appears in the test set for user #u1968 and the training set for user #u6140. Thus, our LLM-based user-side knowledge method effectively captures and summarises user #u1968's interests, as shown by the alignment between item #v2534 and interest #c9831. Furthermore, our recommendation model accurately predicts user #u1968's potential interaction with item #v2534, as shown by its prominent placement in the user's top-K recommendation list. This indicates that the model effectively leverages user-side knowledge to capture higher-order user-item similarity relationships.

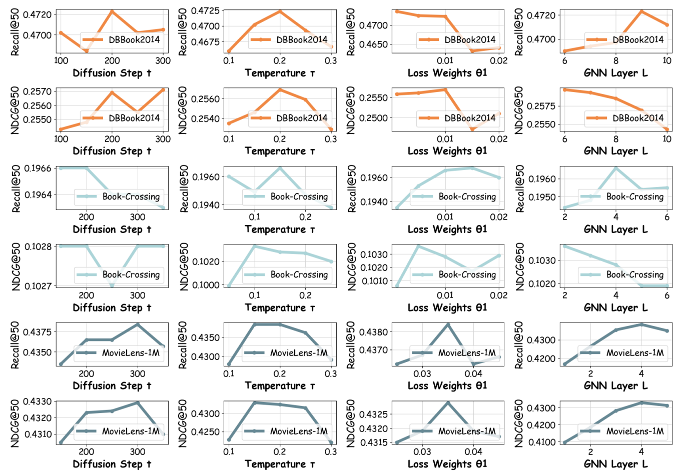

Figure 3: Hyperparameter analysis

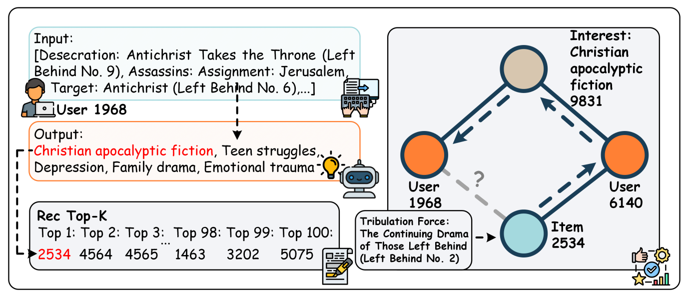

Figure 4: Case Study on Book-Crossing dataset

6 Conclusion

In this paper, we addressed the lack of structured user-side knowledge, a challenge often overlooked by previous knowledge-based recommender systems. We first proposed an LLM-based user-side knowledge construction method, comprising user interest inference and user interest knowledge structuration, to construct a user interest knowledge graph. We then designed a knowledge-aware graph diffusion model with semantic transitions to address the challenges of noise and semantic inconsistency when integrating LLMs-generated user-side information into recommendation systems. Comprehensive experiments on three datasets show SEDIRec outperforms baselines, especially in mitigating user data sparsity. Our future work will explore user-side interest correlations from LLMs and apply causal inference models [18] to remove irrelevant signals, aiming to further improve recommendation performance.

Acknowledgments

Our work was funded by the National Natural Science Foundation of China (62306157), the Natural Science Foundation of Ningxia (2024AAC05011), and the Key Research and Development Program of Ningxia (Special Program for Recruiting High-Caliber Talent) (2025BEH04048).

References

[1] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomás Mikolov. 2017. Enriching Word Vectors with Subword Information. *Transactions of The Association For Computational Linguistics* 5 (2017), 135–146.

[2] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yong-Dong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. In *Proceedings of the 43th International ACM SIGIR conference on Research and Development in Information Retrieval*. ACM, Virtual Event, China, 639–648. doi:10.1145/3397271.3401063

[3] Zhankui He, Zhouhang Xie, Harald Steck, Dawen Liang, Rahul Jha, Nathan Kallus, and Julian J. McAuley. 2025. Reindex-Then-Adapt: Improving Large Language Models for Conversational Recommendation. In *Proceedings of the 8th ACM International Conference on Web Search and Data Mining*. ACM, Hannover, 866–875.

[4] Fang Hu, Yin Zhang, Xingyu Yan, Mingfang Huang, and Xiong Zhang. 2023. An Improved Heterogeneous Graph Convolutional Network for Inter-Relational Medicine Representation Learning. *IEEE Transactions on Multimedia* 30, 1 (2023), 52–61.

[5] Zheng Hu, Zhe Li, Ziyun Jiao, Satoshi Nakagawa, Jiawen Deng, Shimin Cai, Tao Zhou, and Fuji Ren. 2025. Bridging the User-side Knowledge Gap in Knowledge-aware Recommendations with Large Language Models. In *Proceedings of the 39th nternational Conference on Association for the Advancement of Artificial Intelligence*. AAAI Press, Philadelphia, 11799–11807.

[6] Zheng Hu, Satoshi Nakagawa, Yan Zhuang, Jiawen Deng, Shimin Cai, Tao Zhou, and Fuji Ren. 2025. Hierarchical Denoising for Robust Social Recommendation. *IEEE Transactions on Knowledge and Data Engineering* 37, 2 (2025), 739–753.

[7] Meng Jiang, Keqin Bao, Jizhi Zhang, Wenjie Wang, Zhengyi Yang, Fuli Feng, and Xiangnan He. 2024. Item-side Fairness of Large Language Model-based Recommendation System. In *Proceedings of the ACM on Web Conference 2024*. ACM, Singapore, 4717–4726.

[8] Yangqin Jiang, Lianghao Xia, Wei Wei, Da Luo, Kangyi Lin, and Chao Huang. 2024. DiffMM: Multi-Modal Diffusion Model for Recommendation. In *Proceedings of the 32nd ACM International Conference on Multimedia*. ACM, Melbourne, 7591–7599.

[9] Yangqin Jiang, Yuhao Yang, Lianghao Xia, and Chao Huang. 2024. DiffKG: Knowledge Graph Diffusion Model for Recommendation. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*. ACM, Merida, 313–321.

[10] Wang-Cheng Kang, Jianmo Ni, Nikhil Mehta, Maheswaran Sathiamoorthy, Lichan Hong, Ed H. Chi, and Derek Zhiyuan Cheng. 2023. Do LLMs Understand User Preferences? Evaluating LLMs On User Rating Prediction. *CoRR* abs/2305.06474 (2023).

[11] Anchen Li, Bo Yang, Huan Huo, Farookh Hussain, and Guandong Xu. 2025. Hypercomplex Knowledge Graph-Aware Recommendation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, Padua, 2017–2026.

[12] Peibo Li, Maarten de Rijke, Hao Xue, Shuang Ao, Yang Song, and Flora D. Salim. 2024. Large Language Models for Next Point-of-Interest Recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, Washington, 1463–1472.

[13] Zongwei Li, Lianghao Xia, Hua Hua, Shijie Zhang, Shuangyang Wang, and Chao Huang. 2025. DiffGraph: Heterogeneous Graph Diffusion Model. In *Proceedings of the 8th ACM International Conference on Web Search and Data Mining*. ACM, Hannover, 40–49.

[14] Leland McInnes, John Healy, and Steve Astels. 2017. hdbscan: Hierarchical density based clustering. *Journal of Open Source Software* 2, 11 (2017), 205.

[15] Fanshen Meng, Zhenhua Meng, Ru Jin, Rongheng Lin, and Budan Wu. 2025. DOGE: LLMs-Enhanced Hyper-Knowledge Graph Recommender for Multimodal Recommendation. In *Proceedings of 39th International Conference on Association for the Advancement of Artificial Intelligence*. AAAI Press, Philadelphia, 12399–12407.

[16] Fnu Mohbat and Mohammed J. Zaki. 2025. KERL: Knowledge-Enhanced Personalized Recipe Recommendation using Large Language Models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Vienna, 19125–19141.

[17] Yuanpeng Qu and Hajime Nobuhara. 2025. Intent-aware Diffusion with Contrastive Learning for Sequential Recommendation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, Padua, 1552–1561.

[18] Shishir Rao, Mohammad Mamouei, Gholamreza Salimi-Khorshidi, Yikuan Li, Rema Ramakrishnan, Abdelaali Hassaïne, Dexter Canoy, and Kazem Rahimi. 2024. Targeted-BEHRT: Deep Learning for Observational Causal Inference on Longitudinal Electronic Health Records. *IEEE Transactions on Neural Networks and Learning Systems* 35, 4 (2024), 5027–5038.

[19] Yankun Ren, Zhongde Chen, Xinxing Yang, Longfei Li, Cong Jiang, Lei Cheng, Bo Zhang, Linjian Mo, and Jun Zhou. 2024. Enhancing Sequential Recommenders with Augmented Knowledge from Aligned Large Language Models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, Washington, 345–354.

[20] Matías Gabriel Rojas, Ana Carolina Olivera, and Pablo Javier Vidal. 2024. A genetic operators-based Ant Lion Optimiser for training a medical multi-layer perceptron. *Applied Soft Computing* 151 (2024), 111192.

[21] Jiwon Son, Jaeyoon Kim, Taekin Kim, Yeon-Chang Lee, and Sang-Wook Kim. 2025. CATER: A Cluster-Based Alternative-Term Recommendation Framework for Large-Scale Web Search at NAVER. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM, Toronto, 2548–2559.

[22] Fernando Terroso-Sáenz, Francisco Arcas-Túnez, and Andrés Muñoz. 2023. Nation-wide touristic flow prediction with Graph Neural Networks and heterogeneous open data. *Information Fusion* 91 (2023), 582–597. doi:10.1016/j.inffus.2022.11.005

[23] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation Learning with Contrastive Predictive Coding. *CoRR* abs/1807.03748 (2018), 1–13. http://arxiv.org/abs/1807.03748

[24] Ke Wang, Ji Zhang, and Kuan Liu. 2025. Enhancing Cross-Domain Recommendation with Plug-In Contrastive Representations from Large Language Models. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, Padua, 1799–1809.

[25] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. KGAT: Knowledge Graph Attention Network for Recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, Anchorage, 950–958.

[26] Xiang Wang, Tinglin Huang, Dingxian Wang, Yancheng Yuan, Zhenguang Liu, Xiangnan He, and Tat-Seng Chua. 2021. Learning Intents behind Interactions with Knowledge Graph for Recommendation. In *Proceedings of the Web Conference*. ACM, Ljubljana, 878–887.

[27] Wei Wei, Xubin Ren, Jiabin Tang, Qinyong Wang, Lixin Su, Suqi Cheng, Junfeng Wang, Dawei Yin, and Chao Huang. 2024. LLMRec: Large Language Models with Graph Augmentation for Recommendation. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*. ACM, Merida, 806–815.

[28] Yinwei Wei, Xiang Wang, Xiangnan He, Liqiang Nie, Yong Rui, and Tat-Seng Chua. 2022. Hierarchical User Intent Graph Network for Multimedia Recommendation. *IEEE Transactions on Multimedia* 24 (2022), 2701–2712.

[29] Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. 2021. Self-supervised Graph Learning for Recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, Canada, 726–735.

[30] Cong Xu, Yunhang He, Jun Wang, and Wei Zhang. 2025. STAIR: Manipulating Collaborative and Multimodal Information for E-Commerce Recommendation. In *Proceedings of 39th International Conference on Association for the Advancement of Artificial Intelligence*. AAAI Press, Philadelphia, 12899–12907.

[31] Toshihiko Yamasaki, Jiani Hu, Shumpei Sano, and Kiyoharu Aizawa. 2017. FolkPopularityRank: Tag Recommendation for Enhancing Social Popularity using Text Tags in Content Sharing Services. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. ijcai.org, Melbourne, 3231–3237.

[32] Yuhao Yang, Chao Huang, Lianghao Xia, and Chunzhen Huang. 2023. Knowledge Graph Self-Supervised Rationalization for Recommendation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM, Long Beach, 3046–3056.

[33] Yuhao Yang, Chao Huang, Lianghao Xia, and Chenliang Li. 2022. Knowledge Graph Contrastive Learning for Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, Madrid, 1434–1443.

[34] Junliang Yu, Hongzhi Yin, Xin Xia, Tong Chen, Lizhen Cui, and Quoc Viet Hung Nguyen. 2022. Are Graph Augmentations Necessary?: Simple Graph Contrastive Learning for Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, Madrid, 1294–1303.

[35] Yihao Zhang, Junlin Zhu, Ruizhen Chen, Weiwen Liao, Yulin Wang, and Wei Zhou. 2024. Mixed-curvature knowledge-enhanced graph contrastive learning for recommendation. *Expert Systems With Applications* 237, Part B (2024), 121569.

[36] Bowen Zheng, Junjie Zhang, Hongyu Lu, Yu Chen, Ming Chen, Wayne Xin Zhao, and Ji-Rong Wen. 2025. Enhancing Graph Contrastive Learning with Reliable and Informative Augmentation for Recommendation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM, Toronto, 2101–2112.

[37] Ding Zou, Wei Wei, Xian-Ling Mao, Ziyang Wang, Minghui Qiu, Feida Zhu, and Xin Cao. 2022. Multi-level Cross-view Contrastive Learning for Knowledge-aware Recommender System. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, Madrid, 1358–1368.