

Adaptive Fusion Graph Contrastive Learning for Recommendation

Yu Zhang^a, Yiwen Zhang^{a*}, Lei Sang^a, Yi Zhang^a and Yun Yang^b

^aSchool of Computer Science and Technology, Anhui University, Hefei 230601, China

^bDepartment of Computer Science and Software Engineering, Swinburne University of Technology, Melbourne VIC 3122, Australia

ARTICLE INFO

Keywords:

Recommender Systems
Graph Neural Network
Contrastive Learning
Self-Supervised Learning

ABSTRACT

Self-supervised learning (SSL) has recently attracted significant attention in the field of recommender systems. Contrastive learning (CL) stands out as a major SSL paradigm due to its robust ability to generate self-supervised signals. Mainstream graph contrastive learning (GCL)-based methods typically implement CL by creating contrastive views through various data augmentations. Despite these methods are effective, we argue that there still exist several challenges. **i)** Data augmentation requires additional graph convolution (GCN) or modeling operations, significantly increasing time costs. Moreover, graph augmentation disrupts the intrinsic properties of the user-item graph by randomly removing nodes/edges, while feature augmentation applies noise to all nodes, neglecting their unique characteristics. **ii)** Existing GCL-based methods use traditional CL objectives to capture self-supervised signals. However, few studies have explored obtaining more beneficial CL objectives from more perspectives and have attempted to fuse the varying self-supervised signals from these CL objectives to enhance recommendation performance.

To overcome these challenges, we propose Adaptive Fusion Graph Contrastive Learning (AFGCL) for recommendation. Instead of relying on data augmentation, AFGCL exploits structural information naturally produced during graph propagation to construct contrastive representations. Specifically, we introduce an adaptive fusion strategy that estimates the contributions of different propagation depths to the primary recommendation task and adaptively combines their representations. Furthermore, we construct an explicit representation for each observed user-item interaction and propose a fused contrastive objective, which uses the interaction representation as a shared anchor to integrate user-side and item-side supervision within a joint candidate space. Experimental results on three public datasets demonstrate the superior recommendation performance and training efficiency of AFGCL compared with state-of-the-art baselines.

1. Introduction

Recommender systems are essential to a variety of web platforms [1, 2, 3], including e-commerce and streaming services, where they enhance user experiences by delivering personalized content. However, they often face the challenge of data sparsity. Self-supervised learning (SSL) [4] has gained increasing recognition for its effectiveness in addressing data sparsity [5]. The capability of SSL to extract self-supervised signals from large volumes of unlabeled data allows the approach to compensate for missing information, leading to widespread adoption in numerous studies [6]. Among various SSL paradigms, contrastive learning (CL) [7] stands out by acquiring self-supervised signals through maximizing mutual information between positive pairs. In recent years, the success of graph contrastive learning (GCL)-based methods [8] have gained significant attention in the field of recommendation [9, 10, 11].

In general, for CL to be effective, GCL-based methods require at least two distinct contrastive views. Inspired by various domains [12, 13], most existing methods employ data augmentation techniques to generate these views. As illustrated in Fig. 1, data augmentation techniques can be broadly classified into two categories: graph augmentation

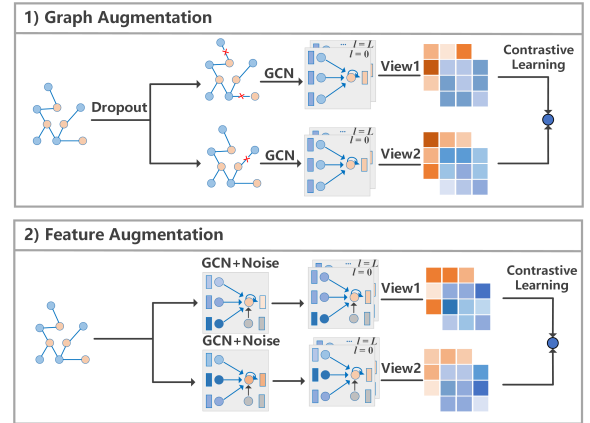


Figure 1: GCL-based methods utilizing data augmentation. Top: Graph augmentation; Bottom: Feature augmentation.

[8] and feature augmentation [14]. Graph augmentation involves creating interaction subgraphs by randomly discarding nodes or edges on the user-item interaction graph and then generating different contrastive views through graph convolution operations. Feature augmentation generates different contrastive views by adding noise to the embeddings during the graph convolution process. With the generated contrastive views, these methods effectively mine users' deep preferences, thereby providing more personalized recommendations.

*Corresponding author.

✉ e125111002@stu.ahu.edu.cn (Yu Zhang^a); zhangyiwen@ahu.edu.cn (Yiwen Zhang^a); sanglei@ahu.edu.cn (Lei Sang^a); zhangyi@stu.ahu.edu.cn (Yi Zhang^a); yyang@swin.edu.au (Yun Yang^b)

Despite the necessity of generating contrastive views, data augmentation techniques and multiple CL objectives pose several drawbacks. **On the one hand.** To create diverse contrastive views, in addition to the neighborhood-aggregated embeddings required for the main recommendation task, data augmentation generally requires additional graph convolution and modeling operations based on the same initial embeddings, which significantly increase training cost per epoch. Furthermore, graph augmentation in data augmentation [8, 15] often randomly drop nodes or edges, thereby disrupting the inherent properties of the input graph. In contrast, feature augmentation alleviate this issue by adding noise to each node, but they still overlook the unique characteristics of individual nodes. **On the other hand.** Existing GCL-based methods construct CL objectives [14, 16] from traditional contrastive views (e.g., user-based and item-based). While the generated self-supervised signals can improve performance to some extent, the fusion of these signals remains problematic, leading to suboptimal recommendation (as analyzed in Section 3.2).

Based on the above analysis, we identify two key challenges critical for advancing this field:

- How to efficiently obtain high-quality contrastive views without data augmentations?
- How to better fuse diverse self-supervised signals captured from different contrastive learning objectives?

To address these challenges, we propose a Adaptive Fusion Graph Contrastive Learning (AFGCL) framework for recommendation. As discussed in Section 4.2, representations obtained from different propagation depths encode collaborative information at different structural ranges and contribute unequally to the recommendation task. We therefore introduce an adaptive representation fusion strategy, which estimates the responses of different propagation layers to the primary recommendation objective and uses them to initialize the layer weights. This enables AFGCL to emphasize propagation signals that are more beneficial to recommendation while retaining the flexibility of end-to-end learning. Moreover, conventional contrastive objectives generally optimize user-side and item-side signals separately, without explicitly modeling the joint semantics of observed interactions. To address this limitation, we construct layer-specific user-item interaction representations and aggregate them using the learned propagation weights. Based on the resulting interaction representation, we further design a fusion contrastive objective, which treats the interaction representation as a shared anchor and incorporates its corresponding user and item as complementary positive signals within a joint candidate space. In this way, AFGCL integrates user-side, item-side, and interaction-level information through a unified contrastive objective without relying on additional data augmentation. Overall, AFGCL provides an effective and efficient framework for learning informative collaborative representations and improving recommendation performance. The main contributions of this paper are as follows:

- We reveal that most GCL-based methods are time-consuming to generate contrastive views and that existing CL-based methods struggle to fuse self-supervised signals.
- We introduce a novel graph contrastive learning paradigm from the perspective of more effective contrastive views construction, and propose **AFGCL** for recommendation.
- We leverage GCN layers to generate adaptive CL representation, and design a fusion CL objective to better integrate self-supervised signals.
- We conduct various experimental studies on three publicly datasets, and the results show that AFGCL has significant advantages in terms of recommendation performance and method training efficiency compared to existing state-of-the-art baselines.

2. Related Works

2.1. GNN-based Recommendation

Graph Neural Networks (GNN) [17, 18] have emerged as a pivotal research direction in recommender systems owing to their ability to capture high-order connectivity patterns through recursive neighborhood aggregation. A seminal contribution in this area is NGCF [19], which first integrated graph message passing into collaborative filtering, enabling the propagation of neighborhood signals across user-item graphs. Subsequently, SGCN [20] simplified this architecture by removing nonlinear activation functions and layer-specific weight matrices, thus enhancing training efficiency without sacrificing performance. Among these, LightGCN [21] has gained widespread adoption for its elegant and effective design, which retains only the essential neighborhood aggregation component and removes all other complexities, yielding a strong baseline for graph-based recommendation.

Beyond general recommendation, GNN have also been widely adopted in domain-specific recommendation scenarios [22]. In sequential recommendation, methods such as GCE-GNN [23] utilize item-item and item-user graphs constructed from interaction sequences to model complex contextual relevance. In social recommendation, models like GraphRec [24] and DVGRL [25] incorporate both interaction graphs and social networks, thereby capturing mutual influence among users alongside personal preferences. While these GNN-based models demonstrate strong representation capabilities, they often lack self-supervised signals, which limits their ability to learn discriminative embeddings under noisy data conditions.

2.2. CL-based Recommendation

Recent studies have increasingly adopted contrastive learning (CL) to derive self-supervised signals for recommendation under sparse supervision [26, 27, 28]. Early graph-based methods mainly focused on view construction. SGL [8] applies node and edge dropout to the user-item

graph, whereas SimGCL [14] generates contrastive views by injecting embedding perturbations during graph propagation. Building on this paradigm, BIGCF [29] performs bi-level contrastive learning over individual and collective intent representations. More recent studies have shifted attention from simply increasing view diversity to improving view reliability. SGCL [15] introduces a symmetric contrastive objective to reduce the influence of noisy augmented views. DimCL [30] analyzes dimension-level learning difficulty and removes redundant, false-positive, and inconsistent dimensional noise. CoGCL [31] instead quantizes collaborative representations into discrete codes and uses them to construct structurally and semantically informative contrastive views.

Another line of work reduces the dependence on explicit data augmentation or revisits the role of contrastive objectives. RecDCL [32] combines batch-wise and feature-wise CL to improve sample discrimination. SCCF [16] theoretically unifies graph convolution and CL, showing that high-order collaborative connectivity can be captured without explicit GNN propagation. CL has also been widely explored in sequential recommendation. ICSRec [33] constructs coarse- and fine-grained intent supervision from cross-user subsequences with shared target items. CaDiRec [34] employs context-aware diffusion to generate semantically consistent sequence augmentations, while TCLARec [35] learns to remove noisy items and insert transition-consistent items, and distinguishes learned and random augmentations through a triplet contrastive objective.

3. Exploring Graph Contrastive Learning for Recommendation

In this section, we explore the key aspects of GCL-based recommendation methods, with a particular focus on data augmentation techniques and contrastive views.

3.1. Effectiveness of Data Augmentation

Existing research on CL underscores the critical role of data augmentations [8, 14, 36, 29], which is attributed to CL that necessitates the creation of diverse contrastive views. In recommendation, there are two main types in Fig. 1: (1) graph augmentation [8] (2) feature augmentation [14]. Graph augmentation typically involves generating two distinct subgraphs by randomly discarding edges/nodes from original graph: $\mathcal{G}'/\mathcal{G}'' = \text{Dropout}(\mathcal{G}(\mathcal{N}, \mathcal{E}), p)$, where $\mathcal{G}$ denotes the original graph, $\mathcal{N}$ and $\mathcal{E}$ denote the nodes and edges, respectively, $\mathcal{G}'/\mathcal{G}''$ denotes the augmentation subgraphs, p denotes keeping rate. Then, we perform graph convolution (GCN) operations by $\mathcal{G}'$ and $\mathcal{G}''$ to obtain different views $\mathbf{e}'$ and $\mathbf{e}''$ for the same node $\mathbf{e}$.

Additionally, feature augmentation has garnered significant attention by introducing noises: $\mathbf{e}'/\mathbf{e}'' = \mathbf{e} + \Delta$, where Δ denotes the added noise (e.g., gaussian noise or uniform noise). After obtaining the contrastive views, contrastive

Table 1

Per-epoch runtime comparison of GCN- and GCL-based methods (with data augmentation) on three datasets (s: seconds; x: slowdown vs. LightGCN).

Method	Amazon-book	Yelp2018	Tmall
LightGCN	67.9s	34.3s	54.4s
SGL-ED	188.1s (2.8x)	92.5s (2.7x)	154.5s (2.8x)
SimGCL	194.5s (2.9x)	83.5s (2.4x)	167.9s (3.1x)
NCL	137.1s (2.0x)	60.5s (1.8x)	89.5s (1.7x)
CGCL	140.5s (2.1x)	55.6s (1.6x)	132.2s (2.4x)
VGCL	98.7s (1.5x)	51.4s (1.5x)	81.7s (1.5x)
SGCL	200.2s (2.9x)	98.6s (2.9x)	172.4s (3.2x)

learning is performed using the CL loss:

$$C(\mathbf{e}_a, \mathbf{e}_b) = - \sum_{\langle a, b \rangle \in \mathcal{B}} \log \frac{\exp(\text{sim}(\mathbf{e}_a, \mathbf{e}_b)/\tau)}{\sum_{c \in \mathcal{B}_b} \exp(\text{sim}(\mathbf{e}_a, \mathbf{e}_c)/\tau)}, \quad (1)$$

where $\mathcal{B} = \{\langle a_1, b_2 \rangle, \langle a_2, b_3 \rangle, \dots, \langle a_m, b_n \rangle\}$ represents the set of positive pairs and τ denotes the temperature coefficient.

Although these contrastive views are effective in improving performance (as shown in Table 2), data augmentation has a significant drawback: generating contrastive views each epoch adds extra time, significantly increasing training costs. Table 1 provides the per-epoch time consumption for the GCN-based method (e.g., LightGCN [21]) and GCL-based methods with (e.g., SGL-ED [8]). For fair comparison, all methods are tested under the same experimental setup, with details provided in Section 5.1. SGL-ED, which utilizes the graph augmentation technique to generate subgraphs, is extremely time-consuming. Additionally, most GCL-based (e.g., SimGCL) methods require repetitive graph convolution or modeling operations to obtain the contrastive views, further contributing to the high time cost. This raises an important question: *Is it possible to propose a novel contrastive learning paradigm that does not rely on data augmentation techniques, but still generates high-quality contrastive views and improves training efficiency?* While data augmentations have proven effective in generating contrastive views and enhancing performance, efficiently generating high-quality contrastive views remains a key challenge in contrastive learning.

3.2. Necessity of More Contrastive Views

To investigate the optimization mechanism of GCL-based recommendation methods, we present the number of contrastive views generated by these methods in Table 2. Specifically, the listed methods obtain different perspectives of the same node (e.g., $\mathbf{e}$) and treat each pair of generated views (e.g., $\text{view1}=\mathbf{e}'$ and $\text{view2}=\mathbf{e}''$) as a positive pair (e.g., $\langle \text{view1}, \text{view2} \rangle$) for contrastive learning. This approach allows them to leverage additional self-supervised signals to optimize model performance. As shown in the statistics, SimGCL, compared to SGL-ED, does not generate additional contrastive views but instead focuses on

Table 2

Contrastive views and performance of representative methods on two datasets. ‘User’ and ‘Item’ denote the numbers of user-based and item-based contrastive views, respectively.

Method	CL Views		Amazon-book		Yelp2018	
	User	Item	R@20	N@20	R@20	N@20
LightGCN	–	–	0.0411	0.0315	0.0639	0.0525
SGL-ED	2	2	0.0478	0.0379	0.0675	0.0555
SimGCL	2	2	0.0515	0.0414	0.0721	0.0601
NCL	4	4	0.0481	0.0373	0.0685	0.0577
CGCL	6	6	0.0483	0.0380	0.0690	0.0560
VGCL	4	4	0.0515	0.0410	0.0715	0.0587
SGCL	2	2	0.0479	0.0377	0.0715	0.0589

optimizing data augmentation strategies. In contrast, subsequent methods [37, 36, 15], shift their focus toward generating more contrastive views (*i.e.*, increasing the number of views). While these methods demonstrate improvements over baselines in their original studies, our experimental results (Table 2) reveal that the gains in performance are modest. Notably, these newer methods often fail to surpass the earlier SimGCL.

Our experimental results suggest that while generating more contrastive views can provide richer self-supervised signals, the subtle differences among these signals pose significant challenges for effective utilization. Consequently, methods that prioritize generating a larger number of contrastive views often exhibit relatively inferior performance compared to SimGCL, which with high-quality contrastive views. As discussed in the previous section, identifying and generating high-quality contrastive views remains a critical yet often overlooked challenge in contrastive learning. Furthermore, as suggested by previous studies [36], although self-supervised signals from different views demonstrate potential in alleviating data sparsity, the nuanced discrepancies between these signals significantly hinder their seamless fusion, ultimately limiting the effectiveness.

4. Methodology

In this section, we propose a Adaptive Fusion Graph Contrastive Learning (AFGCL) framework in Fig. 2. AFGCL leverages adaptive generate contrastive representation and introduces a fusion CL loss, effectively integrating self-supervised signals to alleviate the data sparsity issue.

4.1. Problem Setting

Given $\mathcal{U}(|\mathcal{U}| = M)$ and $\mathcal{I}(|\mathcal{I}| = N)$ denote the set of users and items, respectively. In recommender system [38], Graph Convolutional Networks (GCN) [19, 20, 21] are increasingly favored, which through graph convolution operations, effectively gather adaptive neighborhood information to accurately capture user preference behaviors. LightGCN [21] as currently the most popular GCN encoder, which effectively captures adaptive information through neighborhood aggregation. For efficient training, graph Laplace matrix is proposed: $\tilde{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$, where

$\tilde{\mathbf{A}} \in \mathbb{R}^{(M+N) \times (M+N)}$ denotes graph Laplacian matrix, and $\mathbf{A}$ denotes adjacency matrix, and $\mathbf{D}$ denotes diagonal matrix of $\mathbf{A}$. The point-wise Bayesian Personalisation Ranking (BPR) loss [39] is adopted to optimize model parameters:

$$\mathcal{L}_{\text{Rec}} = -\log(\text{sigmoid}(\mathbf{e}_u^\top \mathbf{e}_i - \mathbf{e}_u^\top \mathbf{e}_j)), \quad (2)$$

where $\langle u, i, j \rangle$ represents a triplet input to the model.

4.2. Adaptive Representation Fusion

From a graph signal perspective, graph propagation applies a polynomial filter over the user-item graph. Different propagation depths exhibit different spectral responses and capture collaborative information at different structural ranges. Prior studies [18] have shown that low-frequency signals characterize global collaborative structures, while high-frequency signals preserve personalized and discriminative information. Therefore, uniformly combining different propagation layers may introduce signals that are not equally beneficial to the recommendation task.

Given the normalized adjacency matrix $\tilde{\mathbf{A}}$, the representation obtained at the l -th propagation layer is defined as

$$\mathbf{E}^{(l)} = \tilde{\mathbf{A}}^l \mathbf{E}^{(0)}, \quad l = 0, 1, \dots, L, \quad (3)$$

where $\mathbf{E}^{(0)}$ denotes the initial user and item embeddings. We focus on the collaboration-enriched propagation layers $S = \{2, \dots, L\}$. To estimate their contributions to the primary recommendation task, we first construct a provisional representation using temporary coefficients:

$$\bar{\mathbf{E}}(\boldsymbol{\beta}) = \sum_{l \in S} \beta_l \mathbf{E}^{(l)}, \quad \beta_l = \frac{1}{|S|}. \quad (4)$$

During a short warm-up stage, we calculate the average descent response of the recommendation objective with respect to each propagation coefficient:

$$\vartheta_l = -\frac{1}{T} \sum_{t=1}^T \left. \frac{\partial \mathcal{L}_{\text{rec}}^{(t)}}{\partial \beta_l} \right|_{\beta_k=1/|S|}, \quad l \in S, \quad (5)$$

where T denotes the number of warm-up iterations. A larger ϑ_l indicates that increasing the contribution of the l -th propagation layer is more consistent with reducing the recommendation loss.

To transform the gradient responses into normalized initial layer weights, we formulate the following entropy-regularized optimization problem:

$$\boldsymbol{\alpha}^{(0)} = \arg \max_{\boldsymbol{\alpha} \in \Delta_S} \left(\sum_{l \in S} \alpha_l \vartheta_l + \tau_g \mathcal{H}(\boldsymbol{\alpha}) \right), \quad (6)$$

where $\Delta_S = \{\boldsymbol{\alpha} \mid \alpha_l \geq 0, \sum_{l \in S} \alpha_l = 1\}$ denotes the probability simplex, and $\mathcal{H}(\boldsymbol{\alpha}) = -\sum_{l \in S} \alpha_l \log \alpha_l$ is the entropy regularizer that prevents the weights from prematurely concentrating on a single propagation depth. By introducing a Lagrange multiplier for the simplex constraint and setting the

derivative with respect to α_l to zero, we obtain the closed-form solution as follows:

$$\alpha_l^{(0)} = \frac{\exp(\vartheta_l/\tau_g)}{\sum_{k \in S} \exp(\vartheta_k/\tau_g)}. \quad (7)$$

To retain the flexibility optimization, we use the gradient-derived solution to initialize a set of learnable layer:

$$w_l^{(0)} = \frac{\vartheta_l}{\tau_g}, \quad \alpha_l = \frac{\exp(w_l)}{\sum_{k \in S} \exp(w_k)}, \quad \mathbf{E} = \sum_{l \in S} \alpha_l \mathbf{E}^{(l)}. \quad (8)$$

Since $\text{softmax}(\mathbf{w}^{(0)}) = \boldsymbol{\alpha}^{(0)}$, the initial weights exactly follow the gradient-derived distribution. The logits $\{w_l\}$ are subsequently optimized together with the remaining model parameters, allowing the model to optimize the contributions of different propagation depths during training.

4.3. User-Item Contrastive Learning Paradigm

Following the analysis in Section 3, we urgently need to find a novel contrastive learning paradigm. Inspired by the natural language processing (NLP) [40], which generates contrastive views by applying the dropout function to input sentence. We believe that similar contrastive views also exist in graph-based recommendation tasks.

Instead of constructing positive pairs from augmented views of the same user/item sample, we argue adaptive interaction pairs as contrastive views. This perspective is rooted in the observation that, in bipartite graphs, users and items are each other's first-order neighbors. As the graph convolution (GCN) process progresses layer by layer, users and items progressively absorb information from overlapping neighborhoods, causing their representations to converge in the embedding space. Therefore, embeddings obtained from adaptive propagation layers naturally serve as semantically aligned contrastive views. Prior contrastive views originate from the same sample, necessitating that a set of views can be selected as negative sample pairs (e.g., $\sum_{j \in B} \mathbf{e}_{u_j}''$):

$$\mathcal{L}_{\text{CL}} = C(\mathbf{e}_u', \mathbf{e}_u'') + C(\mathbf{e}_i', \mathbf{e}_i''). \quad (9)$$

However, considering that the adaptive contrastive views differs from the previous contrastive views (i.e., user-item interaction pairs as a set of contrastive views), we utilize the item views and user views as negative sample pairs for the user and item, respectively. Formally, we propose the user-item contrastive learning paradigm:

$$\mathcal{L}_{\text{CL}} = C(\mathbf{e}_u, \mathbf{e}_i) + C(\mathbf{e}_i, \mathbf{e}_u), \quad (10)$$

where $\mathbf{e}_u$ and $\mathbf{e}_i$ are the user and item embedding enriched with neighborhood information, respectively.

In this section, we leverage the adaptive user-item structure as an alternative source of contrastive views, which serves as a valuable complement to traditional contrastive paradigms. This design not only simplifies the contrastive views generation process but also aligns with the fundamental modeling principles of collaborative filtering.

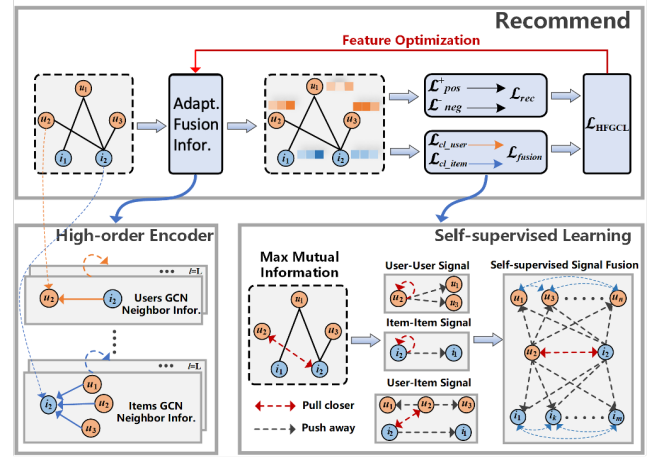


Figure 2: Optimization process of AFGCL. The recommendation objective guides the fusion of propagation signals, while the fusion contrastive objective organizes user-side and item-side supervision around each observed interaction.

4.4. Fusion of Self-supervised Signals

Existing contrastive learning methods generally construct self-supervised objectives from users and items separately. Although these objectives capture useful node-level signals, independently optimizing them may overlook the joint semantics associated with an observed user-item interaction. A straightforward strategy is to place user and item representations in a shared candidate space and directly introduce both types of samples as negatives:

$$C_{\text{Con}}(\mathbf{e}_u, \mathbf{e}_i) = - \sum_{\langle u, i \rangle \in B} \log \frac{\exp(\text{sim}(\mathbf{e}_u, \mathbf{e}_i)/\tau)}{\sum_{\mathbf{x} \in B_{ui}} \exp(\text{sim}(\mathbf{e}_u, \mathbf{x})/\tau)}, \quad (11)$$

where B_u and B_i denote the user and item representations in the current mini-batch, respectively, and $B_{ui} = B_u \cup B_i$ denotes their joint candidate set. While this strategy introduces richer negative samples, it still uses a single endpoint as the contrastive anchor. Consequently, the resulting objective mainly measures the relationships between $\mathbf{e}_u$ and the candidate representations, while the joint semantics of the observed interaction $\langle u, i \rangle$ are not explicitly represented.

To address this limitation, we explicitly construct an interaction representation that jointly characterizes the two endpoints of each observed interaction. Since representations obtained from different propagation layers describe the user-item relationship at different structural ranges, we first define a layer-specific interaction representation:

$$\mathbf{r}_{ui}^{(l)} = \mathbf{W}_s (\mathbf{e}_u^{(l)} + \mathbf{e}_i^{(l)}) + \mathbf{W}_r [\mathbf{e}_u^{(l)} \parallel \mathbf{e}_i^{(l)} \parallel (\mathbf{e}_u^{(l)} \odot \mathbf{e}_i^{(l)})] + \mathbf{b}, \quad (12)$$

where $l \in S$, $\parallel$ and $\odot$ denote concatenation and element-wise multiplication, respectively. The concatenated representations preserve the endpoint-specific information of the user and item, while the element-wise product explicitly

captures their matching pattern. $\mathbf{W}_s$, $\mathbf{W}_r$ and $\mathbf{b}$ are trainable parameters that project the combined features into the interaction representation space.

We further aggregate the interaction representations obtained from different propagation depths using the layer coefficients introduced in Section 4.2. Formally, the fused interaction representation is defined as the weighted center of the layer-specific interaction representations:

$$\mathbf{e}_{ui}^* = \arg \min_{\mathbf{z}} \sum_{l \in S} \alpha_l \|\mathbf{z} - \mathbf{r}_{ui}^{(l)}\|_2^2. \quad (13)$$

By setting the derivative of Eq. (13) with respect to $\mathbf{z}$ to zero and using $\sum_{l \in S} \alpha_l = 1$, we obtain the following closed-form solution:

$$\mathbf{e}_{ui}^* = \sum_{l \in S} \alpha_l \mathbf{r}_{ui}^{(l)}. \quad (14)$$

Here, $\mathbf{e}_{ui}^*$ represents the joint semantics of the observed interaction (u, i) . Sharing the same propagation coefficients $\{\alpha_l\}$ across node and interaction representations ensures that both are constructed from structural ranges favored by the primary recommendation objective.

Based on the fused interaction representation, we organize the user-side and item-side self-supervised signals within a unified contrastive space. Specifically, $\mathbf{e}_{ui}^*$ is used as the shared anchor, while its corresponding user and item representations form the positive set: $\mathcal{P}_{ui} = \{\mathbf{e}_u, \mathbf{e}_i\}$. The joint user-item representation set $\mathcal{B}_{ui}$ serves as the candidate space. Accordingly, the proposed fusion contrastive objective $\mathcal{C}_{\text{Fusion}} = \mathcal{C}_{ui \rightarrow u} + \mathcal{C}_{ui \rightarrow i}$ is formulated as follows:

$$\begin{aligned} \mathcal{C}_{\text{Fusion}} = & - \sum_{(u,i) \in \mathcal{B}} \left(\log \frac{\exp(\text{sim}(\mathbf{e}_{ui}^*, \mathbf{e}_u)/\tau)}{\sum_{\mathbf{x} \in \mathcal{B}_{ui}} \exp(\text{sim}(\mathbf{e}_{ui}^*, \mathbf{x})/\tau)} \right. \\ & \left. + \log \frac{\exp(\text{sim}(\mathbf{e}_{ui}^*, \mathbf{e}_i)/\tau)}{\sum_{\mathbf{x} \in \mathcal{B}_{ui}} \exp(\text{sim}(\mathbf{e}_{ui}^*, \mathbf{x})/\tau)} \right), \end{aligned} \quad (15)$$

where $\text{sim}(\cdot, \cdot)$ denotes cosine similarity, and τ is the temperature coefficient. $\mathcal{C}_{ui \rightarrow u}$ and $\mathcal{C}_{ui \rightarrow i}$ denote the interaction-to-user and interaction-to-item contrastive terms, respectively.

Unlike the endpoint-wise objective in Eq. (11), both terms in Eq. (15) share the same interaction anchor and candidate space. The first term preserves the user-side semantics associated with the interaction, while the second term retains the corresponding item-side semantics. At the same time, the joint candidate set introduces both user and item representations as negative samples, enabling the model to distinguish each observed interaction from unrelated endpoints. Therefore, the final contrastive objective of AFGCL is defined as $\mathcal{L}_{\text{CL}} = \mathcal{C}_{\text{Fusion}}$. By organizing heterogeneous self-supervised signals around the explicit interaction representation, AFGCL avoids independently combining multiple endpoint-wise objectives and integrates user-level, item-level, and interaction-level information within a unified contrastive learning framework.

Algorithm 1 Overall Training Process of AFGCL

Input: interaction matrix $\mathbf{R}$; batch size B ; propagation depth L ; candidate layers S ; warm-up iterations T ; temperatures τ_g and τ ; coefficients λ_1 and λ_2 .

Output: model parameters Θ .

- 1: Initialize model parameters Θ ;
 - 2: Initialize gradient scores $\vartheta_l = 0$ for $l \in S$;
 - 3: **for** $t = 1$ to T **do**
 - 4: Obtain layer-wise representations $\{\mathbf{E}^{(l)}\}_{l=0}^L$;
 - 5: Construct $\bar{\mathbf{E}} = \sum_{l \in S} \beta_l \mathbf{E}^{(l)}$ and calculate $\mathcal{L}_{\text{rec}}^{(t)}$;
 - 6: Update $\vartheta_l \leftarrow \vartheta_l - \frac{1}{T} \frac{\partial \mathcal{L}_{\text{rec}}^{(t)}}{\partial \beta_l}$ for each $l \in S$;
 - 7: Initialize $w_l = \vartheta_l / \tau_g$ for each $l \in S$;
 - 8: **while** AFGCL has not converged **do**
 - 9: Sample a mini-batch $\mathcal{B}$ and obtain $\{\mathbf{E}^{(l)}\}_{l=0}^L$;
 - 10: Calculate $\alpha_l = \frac{\exp(w_l)}{\sum_{k \in S} \exp(w_k)}$ and $\mathbf{E} = \sum_{l \in S} \alpha_l \mathbf{E}^{(l)}$;
 - 11: Calculate the recommendation loss $\mathcal{L}_{\text{Rec}}$;
 - 12: Construct $\mathbf{r}_{ui}^{(l)}$ and $\mathbf{e}_{ui}^*$ via Eqs. (12) and (14);
 - 13: Calculate the CL loss $\mathcal{C}_{\text{Fusion}}$ via Eq. (15);
 - 14: Calculate the overall loss $\mathcal{L}_{\text{AFGCL}}$ via Eq. (16);
 - 15: Update Θ and $\{w_l\}_{l \in S}$ using gradient descent;
 - 16: **return** Θ ;
-

4.5. Method Optimization

For the purpose of enhancing our contrastive method's suitability for the primary recommendation task, we implement a multi-task training strategy to optimize the method's parameters. The overall objective for AFGCL is defined:

$$\mathcal{L}_{\text{AFGCL}} = \mathcal{L}_{\text{Rec}} + \lambda_1 \mathcal{L}_{\text{CL}} + \lambda_2 \|\Theta\|_2^2. \quad (16)$$

where λ_1 and λ_2 denote weights for $\mathcal{L}_{\text{CL}}$ and $\|\Theta\|_2^2$, Θ denotes regularization parameter applied to $\mathbf{E}^{(0)}$. The complete training process of AFGCL is presented in Algorithm 1.

4.6. Method Analysis

4.6.1. Time Complexity Analysis

To demonstrate the efficiency of our AFGCL, we investigate the time complexity of the GCN- and GCL-based methods. Specifically, the embeddings in AFGCL are generated through graph convolution, yielding a time complexity of $\mathcal{O}(2Ld|\mathcal{E}|)$, where $|\mathcal{E}|$ denotes the number of edge, d denotes the embedding dimensions, L denotes the GCN layer. Additionally, CL loss requires fusing the self-supervised signals from user-item contrastive views, the time complexity becomes $\mathcal{O}(Bd(|S| + 4B))$.

Table 3 summarizes the time complexity of AFGCL and representative baselines, where ρ is the edge/node retention probability in SGL-ED, $|\mathcal{K}|$ denotes the number of collective-intent nodes in BIGCF and the number of clusters in NCL/VGCL, $|\mathcal{V}| = M + N$ is the number of user and item nodes, and q is the target rank for SVD in LightGCL. Augmentation-based GCL methods (e.g., SGL-ED and SGCL) require extra graph convolutions to

Table 3

Time complexity of AFGCL and several baselines.

Method	Encoding	CL Loss
NGCF	$\mathcal{O}(2Ld(\mathcal{E} + \mathcal{V}))$	–
LightGCN	$\mathcal{O}(2Ld \mathcal{E})$	–
SGL-ED	$\mathcal{O}(2(1+2p)Ld \mathcal{E})$	$\mathcal{O}(2Bd(1+B))$
SimGCL	$\mathcal{O}(6Ld \mathcal{E})$	$\mathcal{O}(2Bd(1+B))$
NCL	$\mathcal{O}(2Ld \mathcal{E} + d \mathcal{K} \mathcal{V})$	$\mathcal{O}(4Bd + 2B \mathcal{V})$
CGCL	$\mathcal{O}(2Ld \mathcal{E})$	$\mathcal{O}(6Bd + 3B \mathcal{V})$
VGCL	$\mathcal{O}(2Ld \mathcal{E} + d \mathcal{K} \mathcal{V})$	$\mathcal{O}(4Bd(1+B))$
LightGCL	$\mathcal{O}(2Ld(\mathcal{E} + q \mathcal{V}))$	$\mathcal{O}(2Bd + B \mathcal{V})$
RecDCL	$\mathcal{O}(2Ld \mathcal{E})$	$\mathcal{O}(3Bd^2 + 3B^2 + 2Bd)$
BIGCF	$\mathcal{O}(2Ld \mathcal{E} + d \mathcal{K} \mathcal{V})$	$\mathcal{O}(5Bd(1+B))$
SCCF	$\mathcal{O}(2Ld \mathcal{E})$	$\mathcal{O}(Bd^2 + B^2 + 2Bd)$
MixRec	$\mathcal{O}(2Ld \mathcal{E})$	$\mathcal{O}(4Bd(1+2B))$
SGCL	$\mathcal{O}(2(1+2p)Ld \mathcal{E})$	$\mathcal{O}(Bd(1+\mathcal{K}))$
DimCL	$\mathcal{O}(2(1+2p)Ld \mathcal{E})$	$\mathcal{O}(Bd(B+d))$
AFGCL	$\mathcal{O}(2Ld \mathcal{E})$	$\mathcal{O}(Bd(\mathcal{S} + 4B))$

generate views, increasing the overall complexity. Even augmentation-free methods may incur high cost due to dense negatives or specialized modules such as graph reconstruction in VGCL and intent modeling in BIGCF. Recent designs such as RecDCL and SCCF simplify the encoding stage by reformulating the contrastive loss, but the loss computation itself becomes more expensive and the empirical gains are often limited. In contrast, AFGCL avoids additional augmentation and specialized modeling, and achieves an efficient yet effective training procedure by fusing adaptive contrastive views with a fusion contrastive loss, leading to improved recommendation performance.

4.6.2. Theoretical Analysis

The core motivation of AFGCL is to construct high-quality representations and integrate self-supervised signals from different perspectives. As introduced in Section 4.4, the proposed fusion contrastive objective uses the interaction representation $\mathbf{e}_{ui}^*$ as a shared anchor, while the corresponding user and item representations serve as complementary positive signals in a joint candidate space.

To further explain its optimization mechanism, we analyze the gradient of $\mathcal{C}_{\text{Fusion}}$ with respect to the interaction representation $\mathbf{e}_{ui}^*$. For simplicity, we assume that all representations are ℓ_2 -normalized, such that cosine similarity can be expressed as the inner product. For an observed interaction (u, i) , the gradient is given by

$$\frac{\partial \mathcal{C}_{\text{Fusion}}}{\partial \mathbf{e}_{ui}^*} = \frac{1}{\tau} \left(2 \sum_{j \in B_{ui}} p_j \mathbf{e}_j - \mathbf{e}_u - \mathbf{e}_i \right), \quad (17)$$

where

$$p_j = \frac{\exp(\mathbf{e}_{ui}^{*\top} \mathbf{e}_j / \tau)}{\sum_{k \in B_{ui}} \exp(\mathbf{e}_{ui}^{*\top} \mathbf{e}_k / \tau)} \quad (18)$$

denotes the normalized similarity between the interaction representation and candidate $\mathbf{e}_j$. Accordingly, the gradient-descent direction can be written as

$$-\frac{\partial \mathcal{C}_{\text{Fusion}}}{\partial \mathbf{e}_{ui}^*} = \frac{1}{\tau} \left(\mathbf{e}_u + \mathbf{e}_i - 2 \sum_{j \in B_{ui}} p_j \mathbf{e}_j \right). \quad (19)$$

The first term jointly pulls the interaction representation toward its corresponding user and item representations. Therefore, instead of independently optimizing user-side and item-side objectives, the proposed loss integrates both positive signals through the same interaction anchor, thereby preserving the joint semantics of the observed interaction. The second term pushes the interaction representation away from the similarity-weighted center of the joint candidate space. In particular, candidates that are more similar to $\mathbf{e}_{ui}^*$ receive larger weights p_j and consequently produce stronger repulsive effects.

This mechanism enables the model to place greater emphasis on difficult negative samples and improves the discriminability of the learned representations. Therefore, the proposed fusion contrastive objective simultaneously encourages the alignment of the interaction representation with its corresponding user and item, while dispersing unrelated user and item representations in the joint candidate space. These two effects are closely related to the alignment and uniformity properties of contrastive representation learning [41]. Since $\mathbf{e}_{ui}^*$ is constructed from the corresponding user and item representations, the supervision is further propagated to both endpoints through the adaptive representation fusion module.

5. EXPERIMENTS

To demonstrate the effectiveness of AFGCL, we perform experimental comparisons with the state-of-the-art recommendation methods on three real datasets.

5.1. Experimental Settings

5.1.1. Datasets

In selecting datasets to validate the effectiveness of our AFGCL, we choose three datasets that are widely recognized and frequently used for benchmarking in numerous studies: Amazon-book [21], Yelp2018 [42], Tmall [43]:

- **Amazon-book:** is a subset of the Amazon review corpus focusing on implicit user–book interactions.
- **Yelp2018:** is derived from Yelp’s public dataset and contains user interactions across diverse categories.
- **Tmall:** is an e-commerce dataset released by Alibaba, consisting of anonymized user behaviors.

For all datasets, we consider all ratings ‘> 3’ as presence of interactions [21]. We filter out users with less than 10 interactions to ensure the validity of the recommendation. The details of all the datasets are shown in Table 4.

Table 4
Statistics of the datasets.

Dataset	#Users	#Items	#Interactions	Density
Amazon-book	52.6k	91.6k	2984.1k	0.06%
Yelp2018	31.7k	38.0k	1561.4k	0.13%
Tmall	47.9k	41.4k	2619.4k	0.13%

5.1.2. Baselines

BPR-MF [44] learns user/item embeddings by optimizing the BPR [39] objective; **NGCF** [19] performs graph convolution with feature transformation and nonlinear activation to propagate collaborative signals; **LightGCN** [21] simplifies NGCF by focusing on neighborhood aggregation; **SGL-ED** [8] constructs contrastive views via stochastic graph augmentations; **NCL** [37] performs neighbor-aware contrast by leveraging both structural and semantic neighbors to enrich positive signals; **SimGCL** [14] generates contrastive views by perturbing embeddings with Gaussian noise, avoiding explicit graph augmentation; **CGCL** [45] contrasts representations from different GCN layers to capture multi-hop signals while alleviating over-smoothing; **VGCL** [36] learns multi-scale contrastive objectives with variational graph reconstruction to enhance view diversity; **LightGCL** [46] synthesizes global-refined views via SVD-based graph refinement and conducts contrastive learning with the refined view; **RecDCL** [32] reduces false positives and redundancy by dual contrastive objectives and more reliable sampling strategies; **BIGCF** [29] captures individual and collective intents by bilateral intent reconstruction and intent-guided representation learning; **SCCF** [16] simplifies contrastive CF by removing GCN propagation and focusing on lightweight contrast over embeddings; **MixRec** [47] mixes multiple representation views to enhance performance; **SGCL** [15] introduces symmetry theory into graph contrastive learning and designs a symmetric contrastive objective to mitigate the negative effects of noisy augmented views; **DimCL** [30] leverages dimension-wise gradient signals to identify and suppress noisy dimensions in augmented representations; **SimGCF** [18] leverages frequency signal scaler to control representation smoothness.

5.1.3. Evaluation Indicators

To evaluate the recommendation performance of the proposed AFGCL method, we adopt two widely used Top- K ranking metrics: Recall@ K and NDCG@ K , following standard practices in recent literature [48, 14]. These metrics assess recommendation quality from two perspectives: Recall@ K measures the ability of the model to retrieve relevant items within the top- K list (*i.e.*, coverage), while NDCG@ K considers the positions of relevant items, emphasizing ranking quality. In our experiments, we report the results at $K = \{10, 20\}$, and compute the average performance across all users in the test set.

5.1.4. Hyperparameters

To ensure fair and consistent comparisons, all experiments run on a Linux system equipped with a GeForce

RTX 2080Ti GPU. We implement AFGCL in the Pytorch environment. The batch size for all methods are set to 4096, 2048 and 4096 for Amazon-book, Yelp2018 and Tmall datasets, respectively. Except for RecDCL, all methods use an embedding size of 64, while RecDCL uses 2048, as it primarily studies variations across different embedding sizes. Embeddings are initialized using the Xavier strategy. Adam is used as the optimizer by default. For methods utilizing the GCN encoder, the number of GCN layers L is chosen from $\{1, 2, 3, 4\}$. Specifically, for AFGCL, the τ is selected from $\{0.20, 0.22, 0.24, 0.26, 0.28, 0.30\}$, the regularization weights λ_1 is chosen from $\{0.1, 0.5, 1.0, 2.5\}$, the number of GCN layers L is chosen from $\{3, 4\}$, and λ_2 is fixed $1e-6$.

5.2. Overall Performance Comparisons

Here, we compare with all baselines in Table 5. The following observations are made:

- Our AFGCL achieves the most superior performance compared to all baselines on three sparse datasets. Specifically, compared to the strongest baseline, AFGCL improves *w.r.t.* NDCG@20 by 5.30%, 1.96%, and 1.60% on the Amazon-book, Yelp2018, and Tmall datasets, respectively. The experimental results are sufficient to show that AFGCL has a strong recommendation performance, enabling the provision of personalized recommendations.
- Traditional MF-based method generally underperform compared to GNN-based methods, underscoring the substantial improvements that graph structures bring to recommender systems. Among these, LightGCN serves as the encoder for most GNN-based methods due to its straightforward and efficient architectural design. AFGCL uses this encoder, and by extracting the adaptive structural information, we obtain the high-quality contrastive views thus further improving the method efficiency.
- All CL-based methods clearly outperform traditional methods, largely owing to the self-supervised signals introduced through contrastive learning. Among these methods, AFGCL achieves the best performance. This improvement is mainly attributed to its high-quality representations and its ability to effectively fuse diverse self-supervised signals.

5.3. Method Variants and Ablation Study

To examine the contribution of each component in AFGCL, we design the following variants for ablation:

- AFGCL_{w/o AF}: removes adaptive fusion strategy and aggregates all GCN layers for contrastive learning.
- AFGCL_{w/o GCN}: removes the GCN encoder and performs contrastive learning on naive embeddings.
- AFGCL_{w/o CL}: removes the CL loss and uses the adaptive fusion representation strategy.

Table 5

A comparison of proposed AFGCL against the state-of-the-art baselines. The best value is highlighted in **bold**, while the second-best value is underlined. R@ and N@ denote Recall@ and NDCG@, respectively. ‘Improv.%’ denotes the relative improvement over the best baseline. * indicates significant improvements (two-tailed paired t -test, $p < 0.05$) over the best baseline.

Method	Amazon-book				Yelp2018				Tmall			
	R@10	R@20	N@10	N@20	R@10	R@20	N@10	N@20	R@10	R@20	N@10	N@20
BPR-MF	0.0170	0.0308	0.0182	0.0239	0.0278	0.0486	0.0317	0.0394	0.0312	0.0547	0.0287	0.0400
NGCF	0.0199	0.0337	0.0200	0.0262	0.0331	0.0579	0.0368	0.0477	0.0374	0.0629	0.0351	0.0465
LightGCN	0.0228	0.0411	0.0241	0.0315	0.0362	0.0639	0.0414	0.0525	0.0435	0.0711	0.0406	0.0530
SGL-ED	0.0263	0.0478	0.0281	0.0379	0.0395	0.0675	0.0448	0.0555	0.0457	0.0738	0.0434	0.0556
SimGCL	0.0313	0.0515	0.0334	0.0414	0.0424	0.0721	0.0488	0.0601	0.0559	0.0884	0.0536	0.0674
NCL	0.0266	0.0481	0.0284	0.0373	0.0403	0.0685	0.0458	0.0577	0.0459	0.0750	0.0429	0.0553
CGCL	0.0274	0.0483	0.0284	0.0380	0.0404	0.0690	0.0452	0.0560	0.0542	0.0880	0.0510	0.0655
VGCL	0.0312	0.0515	0.0332	0.0410	0.0425	0.0715	0.0485	0.0587	0.0557	0.0880	0.0533	0.0670
LightGCL	0.0303	0.0506	0.0318	0.0397	0.0377	0.0657	0.0437	0.0539	0.0531	0.0832	0.0533	0.0637
RecDCL	0.0311	0.0525	0.0318	0.0407	0.0408	0.0690	0.0464	0.0567	0.0527	0.0853	0.0492	0.0632
BIGCF	0.0294	0.0500	0.0320	0.0398	0.0431	0.0730	0.0497	0.0603	0.0547	0.0876	0.0524	0.0664
SCCF	0.0287	0.0491	0.0294	0.0399	0.0423	0.0718	0.0489	0.0595	0.0478	0.0772	0.0453	0.0580
MixRec	<u>0.0324</u>	<u>0.0539</u>	<u>0.0349</u>	<u>0.0434</u>	<u>0.0432</u>	<u>0.0740</u>	<u>0.0498</u>	<u>0.0612</u>	<u>0.0560</u>	<u>0.0900</u>	<u>0.0537</u>	<u>0.0686</u>
SGCL	0.0281	0.0479	0.0299	0.0377	0.0422	0.0715	0.0484	0.0589	0.0544	0.0872	0.0520	0.0656
DimCL	0.0272	0.0452	0.0283	0.0367	0.0400	0.0678	0.0455	0.0569	0.0485	0.0792	0.0481	0.0600
SimGCF	0.0251	0.0434	0.0258	0.0333	0.0345	0.0669	0.0401	0.0587	0.0472	0.0773	0.0436	0.0566
AFGCL	0.0346*	0.0566*	0.0376*	0.0458*	0.0448*	0.0752*	0.0515*	0.0624*	0.0577*	0.0911*	0.0555*	0.0697*
Improv.%	6.79%	5.01%	7.74%	5.30%	3.70%	1.62%	3.41%	1.96%	3.04%	1.22%	3.35%	1.60%
p -value	3.1e-8	1.7e-8	6.1e-9	4.7e-9	4.9e-8	1.5e-9	2.0e-9	5.3e-9	5.9e-7	1.2e-7	2.2e-7	1.4e-8

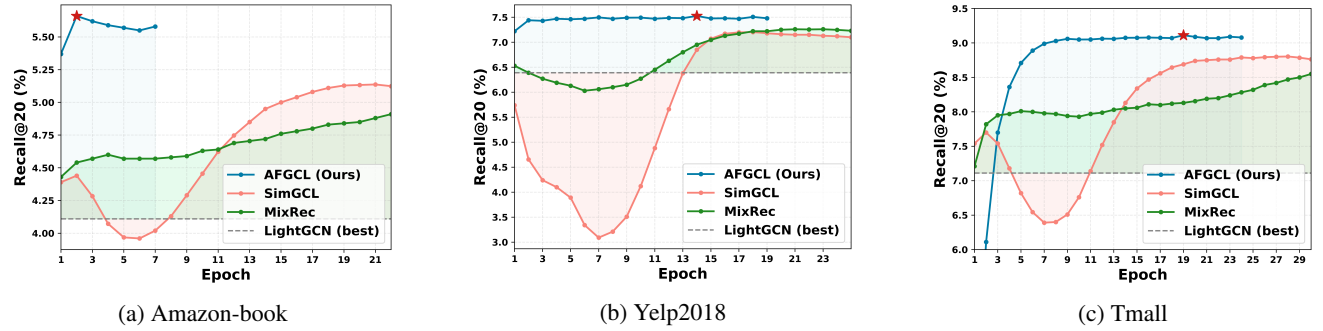


Figure 3: Training curves w.r.t. Recall@20 on three datasets (the red star indicates the best performance achieved by AFGCL).

Table 6

Performance comparison of AFGCL variants and ablation studies on three datasets.

Method	Amazon-book		Yelp2018		Tmall	
	R@20	N@20	R@20	N@20	R@20	N@20
LightGCN	0.0411	0.0315	0.0639	0.0525	0.0711	0.0530
w/o AF	0.0545	0.0432	0.0704	0.0581	0.0782	0.0590
w/o GCN	0.0539	0.0422	0.0722	0.0595	0.0870	0.0666
w/o CL	0.0372	0.0289	0.0570	0.0457	0.0580	0.0426
w/o IR	<u>0.0555</u>	<u>0.0452</u>	<u>0.0734</u>	<u>0.0612</u>	<u>0.0888</u>	<u>0.0680</u>
w/o FCL	0.0380	0.0304	0.0569	0.0471	0.0663	0.0497
AFGCL	0.0566	0.0458	0.0752	0.0624	0.0911	0.0697

- AFGCL_{w/oIR}: removes the explicit interaction representation and directly uses the fused user and item representations for contrastive learning.
- AFGCL_{w/oFCL}: replaces the proposed fusion contrastive loss with the endpoint-wise contrastive objective in Eq. (11).

Based on Table 6, we analyze the contributions of each component from follow two aspects:

5.3.1. Effectiveness of adaptive representation.

We first examine the role of the proposed adaptive fusion strategy. Compared with the full model, w/o AF shows consistent performance degradation when replacing adaptive fusion with simple aggregation over all GCN layers, indicating that treating all propagation depths equally is suboptimal. Furthermore, removing the GCN encoder in AFGCLw/o,GCN leads to inferior performance compared with the full model, indicating that neighborhood aggregation is beneficial for capturing richer collaborative signals. However, we observe that AFGCLw/o,GCN still achieves competitive performance, which highlights the effectiveness of the proposed fusion CL loss. This suggests that even with naive embeddings, the proposed contrastive objective can capture self-supervised signal, while combining it with adaptive fusion further enhances representation quality.

5.3.2. Importance of contrastive learning.

We next analyze the effectiveness of the proposed contrastive learning objective. Removing the contrastive loss in AFGCL_{w/oCL} results in a significant performance drop,

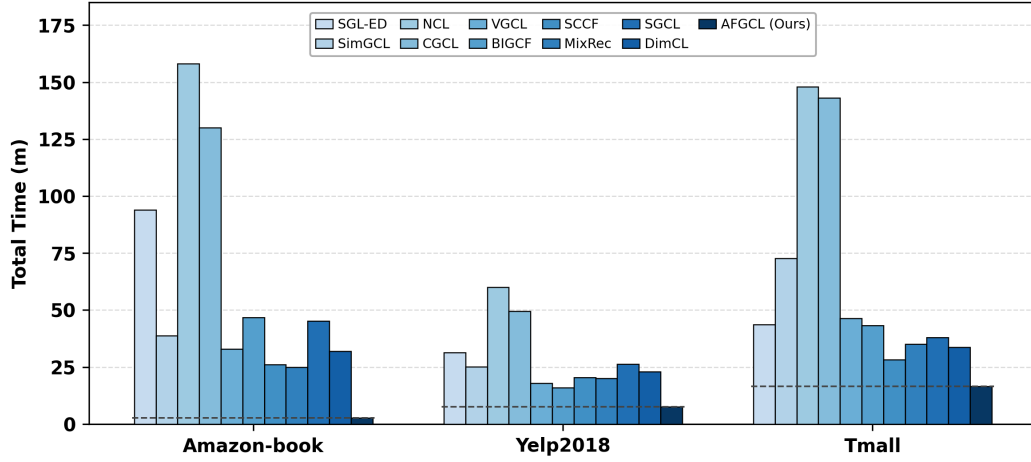


Figure 4: Total training costs comparison of AFGCL and other GCL-based methods on three datasets.

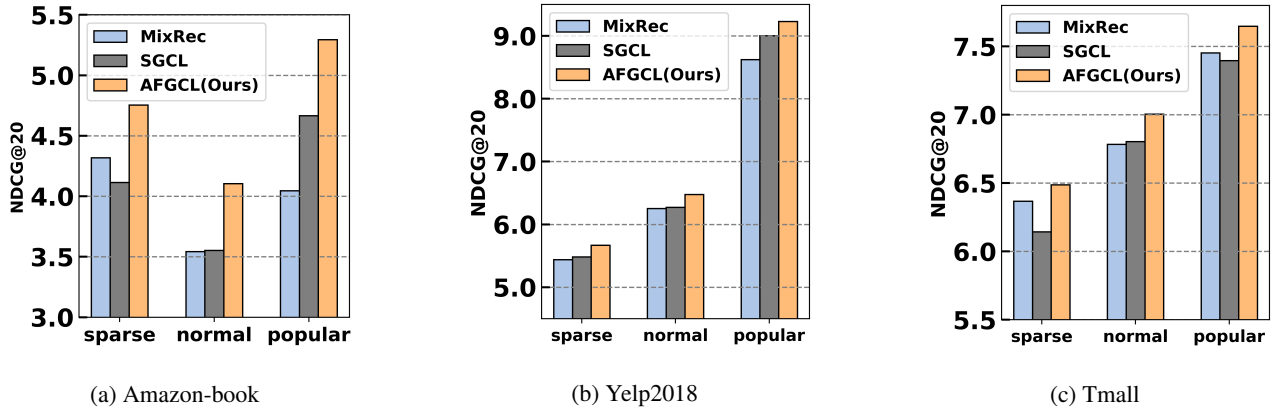


Figure 5: Performance comparison w.r.t. NDCG@20 across different user sparsity groups on three datasets.

confirming that additional self-supervised signals are important for enhancing user and item representations. We further evaluate the explicit interaction representation and the fusion contrastive objective. Both $\text{AFGCL}_{w/oIR}$ and $\text{AFGCL}_{w/oFCL}$ consistently underperform the full model. The degradation of $\text{AFGCL}_{w/oIR}$ indicates that directly contrasting user and item endpoints cannot fully capture the joint semantics of observed interactions. Meanwhile, the inferior performance of $\text{AFGCL}_{w/oFCL}$ demonstrates the advantage of using the interaction representation as a shared anchor and incorporating user-side and item-side samples within a unified contrastive space. These results verify that the interaction representation and fusion contrastive objective jointly contribute to the effectiveness of AFGCL.

Overall, the ablation results verify that the proposed framework benefits from both adaptive representation learning and well-designed contrastive objectives.

5.4. Method Efficiency Study

To assess training behavior, we plot the Recall@20 curves of AFGCL, SimGCL, and MixRec during training (Fig. 3). AFGCL reaches a stable performance within a

few early epochs, showing noticeably faster convergence than the other methods. In contrast, SimGCL and MixRec improve more slowly and present larger fluctuations at the beginning. We attribute the rapid convergence to AFGCL's fusion of self-supervised signals from multiple contrastive views, which provides more reliable supervision in early training. By comparison, SimGCL constructs views via perturbations, resulting in less stable signals at initialization, while MixRec introduces explicit intent modeling, increasing optimization complexity and slowing convergence.

We further compare the total training time of state-of-the-art methods (Fig. 4). AFGCL consistently achieves the lowest training cost on all three datasets, mainly due to its fast convergence enabled by efficient multi-view signal fusion. In contrast, augmentation-based methods such as SGL-ED and SimGCL require additional graph convolutions for generating perturbed views, which is time-consuming. For NCL and CGCL, although they avoid augmentation, the computation over large user/item sets for negatives also incurs substantial overhead. Finally, RecDCL is expected to be among the most expensive due to its large embedding

dimension (2048), even though its training-time results are not reported.

5.5. Method Sparsity Study

To evaluate AFGCL under different sparsity levels, we partition users into three groups (*sparse*, *common*, and *popular*) according to interaction frequency, following prior works [8]. The results are reported in Fig. 5.

We focus on the *sparse* group, which is prevalent in real-world recommendation where supervision is limited. On all three datasets, AFGCL consistently outperforms strong baselines (e.g., MixRec and SGCL), indicating its advantage in modeling user preferences with few interactions. This gain is largely attributed to the adaptive contrastive views, which help capture implicit user-item associations beyond immediate neighborhoods. Meanwhile, AFGCL remains stable across all groups, showing good generalization to moderately and highly active users. Notably, on Amazon-book, several methods drop in the *common* group, likely due to noisier or less consistent behaviors; AFGCL is less affected, demonstrating robustness to both sparsity and noise.

5.6. Method Robustness Study

To evaluate AFGCL under noisy interactions, we inject 10%, 20%, and 30% adversarial user-item interactions into the training data while keeping the test set unchanged. The results are reported in Fig. 6.

Across all three datasets, AFGCL consistently achieves the best performance and exhibits only moderate degradation as the noise ratio increases. In contrast, LightGCN is more sensitive to corrupted interactions due to the lack of additional self-supervision. MixRec alleviates the degradation through contrastive learning, but its perturbation-based views may become less reliable when the underlying interaction graph is already noisy. AFGCL remains more stable because its adaptive representations emphasize informative high-order collaborative signals, while the fusion contrastive objective integrates supervision from multiple relational perspectives. These results demonstrate that AFGCL can effectively reduce the influence of abnormal interactions and maintain reliable recommendation performance under different noise levels.

5.7. Method Hyperparameter Study

5.7.1. Temperature Coefficient τ

The temperature coefficient τ controls the hardness of contrastive optimization by scaling similarities between positive and negative pairs. As shown in Fig. 7(a), AFGCL is sensitive to τ , with the best performance typically achieved at a slightly larger value (around 0.2–0.3). This is mainly because AFGCL constructs more contrastive views, introducing more negatives; a higher τ alleviates over-penalization from abundant negatives and improves diverse signal integration, leading to better performance.

5.7.2. Fusion Loss Weight λ_1

The weight λ_1 balances the fusion contrastive loss and the recommendation objective. Fig. 7(b) shows that AFGCL

remains stable across a wide range of λ_1 , indicating low sensitivity. Moderate increases of λ_1 can improve performance, while overly large values may bring marginal gains and slow convergence. Hence, we adopt the empirically validated range for a good trade-off.

6. Conclusion and Future Work

This paper revisits the role of data augmentation in graph contrastive learning for recommendation and highlights its limitations. Motivated by these observations, we propose Adaptive Fusion Graph Contrastive Learning (AFGCL), which filters low-order information to construct high-order contrastive views, and further introduces a fusion contrastive loss to integrate diverse self-supervised signals. Extensive experiments on three public datasets demonstrate the effectiveness and efficiency of AFGCL. In future, we will further study how contrastive objectives produce self-supervised signals and improve their interpretability in recommender systems. We also plan to extend high-order contrastive views to other recommendation scenarios, such as sequential and social recommendation.

CRedit authorship contribution statement

Yu Zhang: Conceptualization, Methodology, Validation, Writing - Original Draft, Writing - Review & Editing. **Yiwen Zhang:** Conceptualization, Validation, Writing - Review & Editing, Funding acquisition. **Lei Sang:** Methodology, Validation, Writing - Review & Editing, Visualization. **Yi Zhang:** Validation, Software, Writing - Review. **Yun Yang:** Supervision, Writing - Review.

Declaration of competing interest

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

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No.62272001) and in part by the Australian Research Council Discovery Projects (Grant Nos. DP200102491, DP230101790) and Linkage Projects (Grant Nos. LP210301393, LP220100482).

References

- [1] C. Wu, F. Wu, Y. Huang, X. Xie, Personalized news recommendation: Methods and challenges, *ACM Transactions on Information Systems* 41 (1) (Jan. 2023). doi:10.1145/3530257.
- [2] H. Chen, Y. Wang, Y. Xu, Y. Chen, Context-aware diffusion model for sequential recommendation, *Expert Systems with Applications* 331 (2026) 133420. doi:https://doi.org/10.1016/j.eswa.2026.133420.
- [3] R. Liu, S. Wang, A. Li, Bssrec: Balance theory-based sign-aware denoising method for enhancing social recommendation, *Expert Systems with Applications* 319 (2026) 132145. doi:https://doi.org/10.1016/j.eswa.2026.132145.

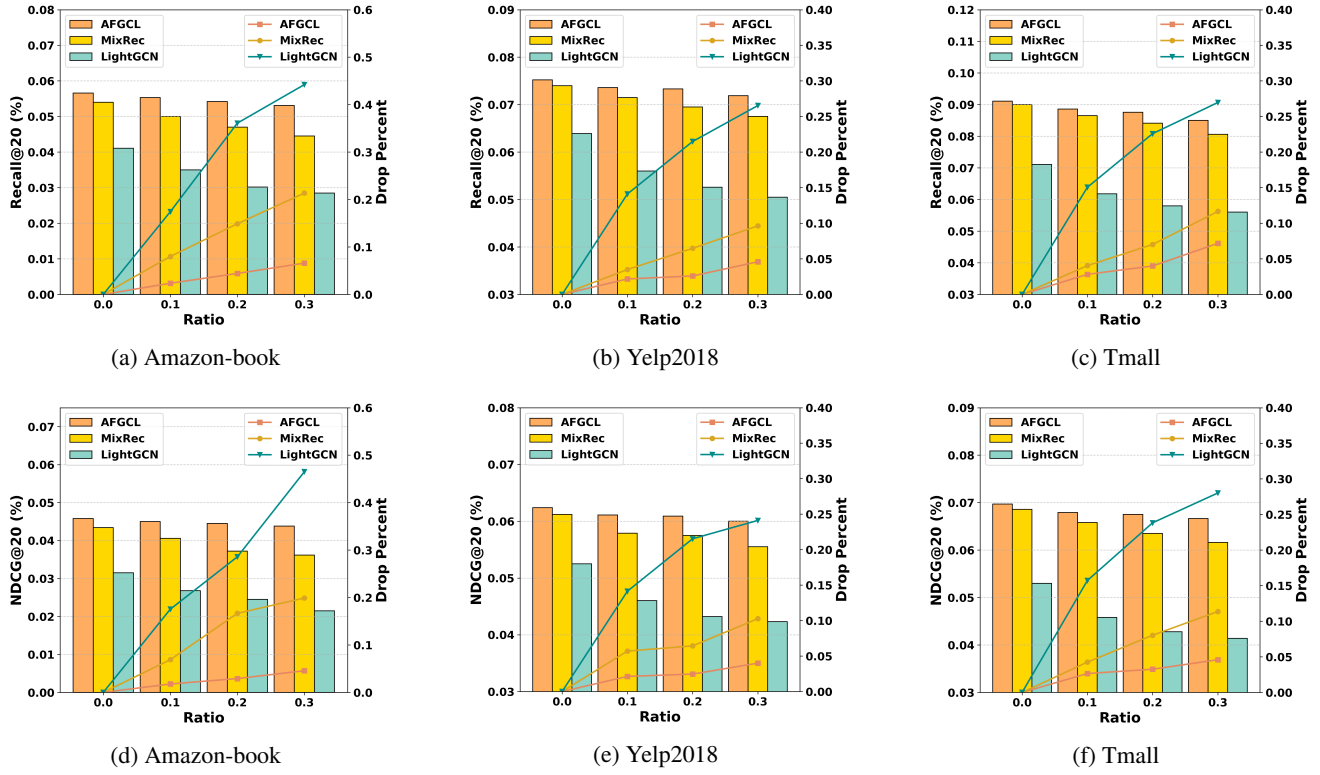


Figure 6: Performance comparison w.r.t. Recall@20 (top) and NDCG@20 (bottom) under different noise levels on three datasets.

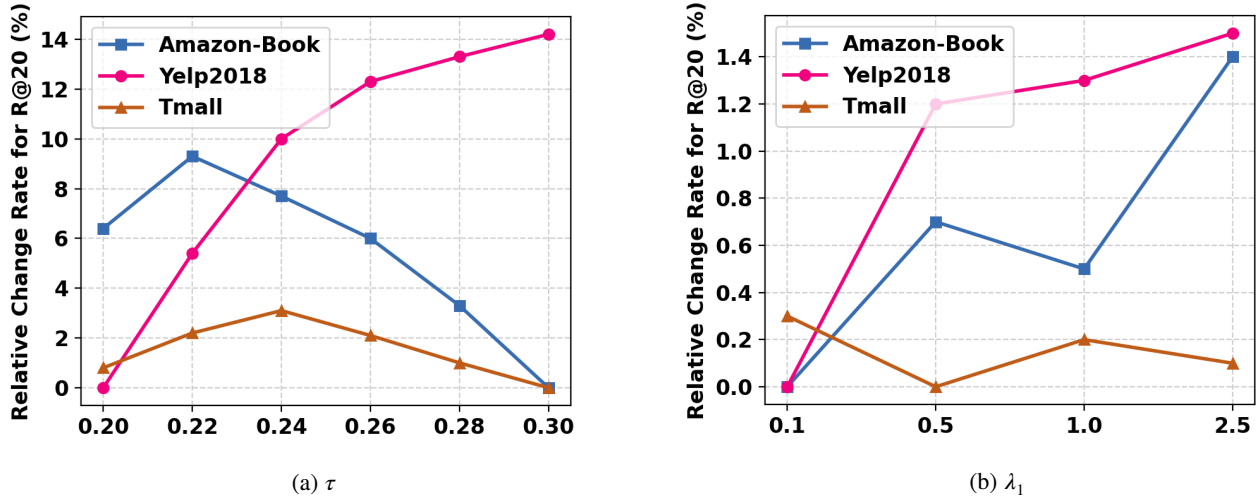


Figure 7: Hyperparameter sensitivities to (a) the temperature coefficient τ and (b) the graph contrastive regularization weight λ_1 w.r.t. Recall@20 across three datasets.

- [4] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, J. Tang, Self-supervised learning: Generative or contrastive, IEEE Transactions on Knowledge and Data Engineering 35 (1) (2023) 857–876. doi:10.1109/TKDE.2021.3090866.
- [5] L. Xia, C. Huang, C. Huang, K. Lin, T. Yu, B. Kao, Automated Self-Supervised Learning for Recommendation, in: Proceedings of the ACM Web Conference 2023, 2023, pp. 992–1002. doi:10.1145/3543507.3583336.
- [6] M. Assran, Q. Duval, I. Misra, P. Bojanowski, P. Vincent, M. Rabbat, Y. LeCun, N. Ballas, Self-supervised learning from images with a joint-embedding predictive architecture, in: IEEE Conference on Computer Vision and Pattern Recognition, 2023, pp. 15619–15629.
- [7] M. Jing, Y. Zhu, T. Zang, K. Wang, Contrastive self-supervised learning in recommender systems: A survey, ACM Transactions on Information Systems 42 (2) (Nov 2023). doi:10.1145/3627158.
- [8] J. Wu, X. Wang, F. Feng, X. He, L. Chen, J. Lian, X. Xie, Self-supervised Graph Learning for Recommendation, in: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2021, pp. 726–735. doi:10.1145/3404835.3462862.
- [9] C. Gao, X. Wang, X. He, Y. Li, Graph neural networks for recommender system, in: Proceedings of the Fifteenth ACM International

- Conference on Web Search and Data Mining, WSDM '22, 2022, p. 1623–1625. doi:10.1145/3488560.3501396.
- [10] J. Tanveer, S.-W. Lee, A. M. Rahmani, K. Aurangzeb, M. Alam, G. Zare, P. M. Alamdari, M. Hosseinzadeh, Pga-drl: Progressive graph attention-based deep reinforcement learning for recommender systems, *Information Fusion* 121 (2025) 103167. doi:https://doi.org/10.1016/j.inffus.2025.103167.
 - [11] Y. Zhao, L. Chen, X. Feng, M. Chen, J. Gong, Cross-domain recommendation via quantized disentangled generative model, *Information Fusion* 129 (2026) 104028. doi:https://doi.org/10.1016/j.inffus.2025.104028.
 - [12] M. Bayer, M.-A. Kauffhold, C. Reuter, A survey on data augmentation for text classification, *ACM Computing Surveys* 55 (7) (Dec 2022). doi:10.1145/3544558.
 - [13] X. Yang, X. Xu, W. Du, Z. Liu, J. Chen, Y. Fu, J. Hu, Diffhgcl: Task-aware diffusion-augmented heterogeneous graph contrastive learning for web api recommendation, *Expert Systems with Applications* 331 (2026) 133349. doi:https://doi.org/10.1016/j.eswa.2026.133349.
 - [14] J. Yu, H. Yin, X. Xia, T. Chen, L. Cui, Q. V. H. Nguyen, 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*, 2022, pp. 1294–1303. doi:10.1145/3477495.3531937.
 - [15] C. Zhao, E. Yang, Y. Liang, J. Zhao, G. Guo, X. Wang, Symmetric graph contrastive learning against noisy views for recommendation, *ACM Trans. Inf. Syst.* 43 (3) (May 2025).
 - [16] Y. Wu, L. Zhang, F. Mo, T. Zhu, W. Ma, J.-Y. Nie, Unifying graph convolution and contrastive learning in collaborative filtering, in: *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD '24*, 2024, p. 3425–3436. doi:10.1145/3637528.3671840.
 - [17] S. Wu, F. Sun, W. Zhang, X. Xie, B. Cui, Graph neural networks in recommender systems: A survey, *ACM Computing Surveys* 55 (5) (Dec 2022). doi:10.1145/3535101.
 - [18] F. Liu, H. Cang, H. Yuan, J. Fan, Y. Hao, F. Zhuang, G. Liu, P. Zhao, How do graph signals affect recommendation: Unveiling the mystery of low and high-frequency graph signals, in: *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1, KDD '26*, 2026, p. 831–842.
 - [19] X. Wang, X. He, M. Wang, F. Feng, T.-S. Chua, Neural Graph Collaborative Filtering, in: *Proceedings of the 42nd International ACM SIGIR Conference*, 2019, pp. 165–174. doi:10.1145/3331184.3331267.
 - [20] F. Wu, A. Souza, T. Zhang, C. Fifty, T. Yu, K. Weinberger, *Simplifying Graph Convolutional Networks*, in: *Proceedings of the 36th International Conference on Machine Learning*, Vol. 97, 2019, pp. 6861–6871. URL <https://proceedings.mlr.press/v97/wu19e.html>
 - [21] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, M. Wang, LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation, in: *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 639–648. doi:10.1145/3397271.3401063.
 - [22] K. Sharma, Y.-C. Lee, S. Nambi, A. Salian, S. Shah, S.-W. Kim, S. Kumar, A survey of graph neural networks for social recommender systems, *ACM Comput. Surv.* 56 (10) (jun 2024). doi:10.1145/3661821.
 - [23] Z. Wang, W. Wei, G. Cong, X.-L. Li, X.-L. Mao, M. Qiu, Global context enhanced graph neural networks for session-based recommendation, in: *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, p. 169–178. doi:10.1145/3397271.3401142.
 - [24] W. Fan, Y. Ma, Q. Li, Y. He, E. Zhao, J. Tang, D. Yin, Graph neural networks for social recommendation, in: *The World Wide Web Conference, WWW '19*, 2019, p. 417–426. doi:10.1145/3308558.3313488.
 - [25] Y. Zhang, Y. Zhang, Y. Zhao, S. Deng, Y. Yang, Dual variational graph reconstruction learning for social recommendation, *IEEE Transactions on Knowledge and Data Engineering* (2024). doi:10.1109/TKDE.2024.3386895.
 - [26] J. Yu, H. Yin, X. Xia, T. Chen, J. Li, Z. Huang, Self-supervised learning for recommender systems: A survey, *IEEE Transactions on Knowledge and Data Engineering* 36 (1) (2024) 335–355. doi:10.1109/TKDE.2023.3282907.
 - [27] Q. Xie, R. Tang, X. Mo, Diffusion-enhanced negative sampling in multimodal contrastive learning for recommendation, *Expert Systems with Applications* 320 (2026) 132214. doi:https://doi.org/10.1016/j.eswa.2026.132214.
 - [28] A. Samad, V. Bhatia, Fedgclrec: Federated graph contrastive learning framework for social influence recommendations, *Expert Systems with Applications* 297 (2026) 129294. doi:https://doi.org/10.1016/j.eswa.2025.129294.
 - [29] Y. Zhang, L. Sang, Y. Zhang, Exploring the individuality and collectivity of intents behind interactions for graph collaborative filtering, in: *Proceedings of the 47th International ACM SIGIR Conference, SIGIR '24*, Association for Computing Machinery, 2024, p. 1253–1262. doi:10.1145/3626772.3657738.
 - [30] C. Zhang, Q. Han, Q. Tan, S. Wang, X. Zhao, R. Chen, Dimcl: Dimension-aware augmentation in contrastive learning for recommendation, in: *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1, KDD '25*, 2025, p. 1913–1923.
 - [31] B. Zheng, J. Zhang, H. Lu, Y. Chen, M. Chen, W. X. Zhao, J.-R. Wen, 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 V.1, KDD '25*, 2025, p. 2101–2112.
 - [32] D. Zhang, Y. Geng, W. Gong, Z. Qi, Z. Chen, X. Tang, Y. Shan, Y. Dong, J. Tang, Recdcl: Dual contrastive learning for recommendation, in: *Proceedings of the ACM on Web Conference 2024, WWW '24*, 2024, p. 3655–3666. doi:10.1145/3589334.3645533.
 - [33] X. Qin, H. Yuan, P. Zhao, G. Liu, F. Zhuang, V. S. Sheng, Intent contrastive learning with cross subsequences for sequential recommendation, in: *Proceedings of the 17th ACM International Conference on Web Search and Data Mining, WSDM '24*, 2024, p. 548–556.
 - [34] Z. Cui, H. Wu, B. He, J. Cheng, C. Ma, Context matters: Enhancing sequential recommendation with context-aware diffusion-based contrastive learning, in: *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, CIKM '24*, 2024, p. 404–414.
 - [35] W. Wang, Y. Lin, M. Zhang, H. Lu, J. Zhao, J. Sun, X. Ben, P. Ren, Y. Li, Triplet contrastive learning with learnable sequence augmentation for sequential recommendation, in: *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '25*, 2025, p. 1519–1529.
 - [36] Y. Yang, Z. Wu, L. Wu, K. Zhang, R. Hong, Z. Zhang, J. Zhou, M. Wang, Generative-Contrastive Graph Learning for Recommendation, in: *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2023, pp. 1117–1126. doi:10.1145/3539618.3591691.
 - [37] Z. Lin, C. Tian, Y. Hou, W. X. Zhao, Improving graph collaborative filtering with neighborhood-enriched contrastive learning, in: *Proceedings of the ACM Web Conference 2022, WWW '22*, 2022, p. 2320–2329. doi:10.1145/3485447.3512104.
 - [38] L. Wu, X. He, X. Wang, K. Zhang, M. Wang, A survey on accuracy-oriented neural recommendation: From collaborative filtering to information-rich recommendation, *IEEE Transactions on Knowledge and Data Engineering* 35 (5) (2023) 4425–4445. doi:10.1109/TKDE.2022.3145690.
 - [39] S. Rendle, C. Freudenthaler, Z. Gantner, L. Schmidt-Thieme, BPR: Bayesian personalized ranking from implicit feedback, in: *Proceedings of the 25th Conference on Uncertainty in AI*, 2009, pp. 452–461. doi:10.48550/arXiv.1205.2618.
 - [40] T. Gao, X. Yao, D. Chen, Simcse: Simple contrastive learning of sentence embeddings (2022). arXiv:2104.08821, doi:10.48550/arXiv.2104.08821.

- [41] C. Wang, Y. Yu, W. Ma, M. Zhang, C. Chen, Y. Liu, S. Ma, Towards Representation Alignment and Uniformity in Collaborative Filtering, in: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2022, pp. 1816–1825.
- [42] L. Sang, Y. Zhang, Y. Zhang, H. Li, Y. Zhang, Towards similar alignment and unique uniformity in collaborative filtering, Expert Systems with Applications 259 (2025) 125346. doi:<https://doi.org/10.1016/j.eswa.2024.125346>.
- [43] X. Ren, L. Xia, J. Zhao, D. Yin, C. Huang, Disentangled contrastive collaborative filtering, in: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '23, 2023, p. 1137–1146. doi:[10.1145/3539618.3591665](https://doi.org/10.1145/3539618.3591665).
- [44] Y. Koren, R. Bell, C. Volinsky, Matrix Factorization Techniques for Recommender Systems, Computer 42 (8) (2009) 30–37. doi:[10.1109/MC.2009.263](https://doi.org/10.1109/MC.2009.263).
- [45] W. He, G. Sun, J. Lu, X. S. Fang, Candidate-aware graph contrastive learning for recommendation, in: Proceedings of the 46th International ACM SIGIR Conference, SIGIR '23, 2023, p. 1670–1679. doi:[10.1145/3539618.3591647](https://doi.org/10.1145/3539618.3591647).
- [46] X. Cai, C. Huang, L. Xia, X. Ren, Lightgcl: Simple yet effective graph contrastive learning for recommendation, in: The Eleventh International Conference on Learning Representations, 2023. arXiv: [2302.08191](https://arxiv.org/abs/2302.08191), doi:[10.48550/arXiv.2302.08191](https://doi.org/10.48550/arXiv.2302.08191)Focustolearnmore.
- [47] Y. Zhang, Y. Zhang, Mixrec: Individual and collective mixing empowers data augmentation for recommender systems, in: Proceedings of the ACM on Web Conference 2025, WWW '25, 2025, p. 2198–2208. doi:[10.1145/3696410.3714565](https://doi.org/10.1145/3696410.3714565).
- [48] Y. Zhang, Y. Zhang, D. Yan, S. Deng, Y. Yang, Revisiting Graph-based Recommender Systems from the Perspective of Variational Auto-Encoder, ACM Transactions on Information Systems 41 (3) (2023) 1–28. doi:[10.1145/3573385](https://doi.org/10.1145/3573385).