DivGCL: A Graph Contrastive Learning Model for Diverse Recommendation

Wenwen Gong[1], Yangliao Geng[2], Dan Zhang[3], Yifan Zhu[4], Xiaolong Xu[5], Haolong Xiang[5], Amin Beheshti[6], Xuyun Zhang[6], Lianyong Qi[7,8]*

[1]School of Information and Electrical Engineering, China Agricultural University, China
[2]School of Computer Science & Technology, Beijing Jiaotong University, China
[3]Department of Computer Science and Technology, Tsinghua University, China
[4]School of Computer Science, Beijing University of Posts and Telecommunications, China
[5]School of Software, Nanjing University of Information Science and Technology, China
[6]School of Computing, Macquarie University, Australia
[7]College of Computer Science and Technology, China University of Petroleum (East China), China
[8]State Key Lab. for Novel Software Technology, Nanjing University, China
wenweng@cau.edu.cn, gengyla@bjtu.edu.cn, zd21@mails.tsinghua.edu.cn, yifan_zhu@bupt.edu.cn,
{xlxu, hlxiang}@nuist.edu.cn, {xuyun.zhang, amin.beheshti}@mq.edu.au, lianyongqi@upc.edu.cn

Abstract

Graph Contrastive Learning (GCL), as a primary paradigm of graph self-supervised learning, spurs a fruitful line of research in tackling the data sparsity issue by maximizing the consistency of user/item embeddings between different augmented views with random perturbations. However, diversity, as a crucial metric for recommendation performance and user satisfaction, has received rather little attention. In fact, there exists a challenging dilemma in balancing accuracy and diversity. To address these issues, we propose a new **G**raph **C**ontrastive **L**earning model for **Div**ersifying recommendations (DivGCL). Inspired by the excellence of the determinant point process (DPP), DivGCL adopts a DPP likelihood-based loss function to achieve an ideal trade-off between diversity and accuracy, optimizing it jointly with the advanced Gaussian noise-augmented GCL objective. Extensive experiments on four popular datasets demonstrate that DivGCL surpasses existing approaches in balancing accuracy and diversity, with an improvement of 23.47% at T@20 (abbreviation for trade-off metric) on ML-1M.

1 Introduction

With the advent of the era of information overload, recommender systems, as effective means to provide users with the most useful information from massive data, including time series data (Fan et al. 2023, 2022), have attracted much attention and become irreplaceable (Zhang et al. 2019; Zhao et al. 2021; He et al. 2017; Fan Wang 2024; Wang et al. 2025). With the prevalence of graph neural networks (GNN), a typical pipeline of GNN-based recommendations (Zhang et al. 2023; Chen et al. 2024b) first model user-item interaction data as a bipartite graph and then generate node embeddings by aggregating ones from their own neighbors, which yields outstanding accuracy performance. Especially, graph contrastive learning (GCL) has achieved remarkable success in the field of recommendation for tackling the data sparsity issue, since its powerful ability to extract self-supervised signals from the original bipartite graph.

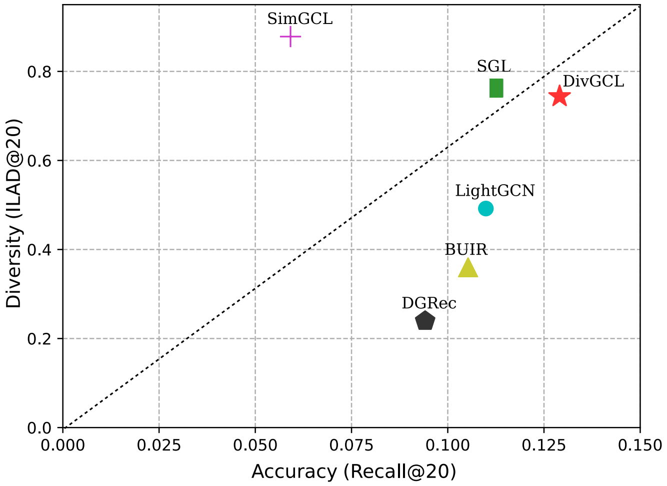


Figure 1: The motivation of DivGCL. Existing accuracy vs. diversity of current methods on Beauty dataset.

However, GCL-based recommendation models often favor popular items and pursue the recommendation accuracy, leading to unsatisfactory diversity (Lin et al. 2022; Lee et al. 2021; Yu et al. 2022; Yang et al. 2021; Wang et al. 2022; Zhang et al. 2024; Chen et al. 2024c,a). This will lead to a large redundancy in the recommendation list and then hurt user satisfaction and enterprise profits. Thus, accuracy is no longer the only goal pursued by researchers, then, how to obtain more diverse recommendation results has attracted more and more attention from scholars (Yang et al. 2023; Zheng et al. 2021). In fact, a well-developed recommender model often be comprehensively evaluated from distinct perspectives, e.g. diversity. Early tremendous endeavors at diversification mainly target the downstream post-processing stage after the recommendation candidate stage. The two stages are independent of each other, i.e., diversification signals are interrupted and manifest in the upstream optimization of the

---

*Corresponding Author.

recommendation process, which will be prone to produce suboptimal results. Hence, a natural question arises: *How to inject diversification signals into end-to-end GCL-based models for maximizing the utility of recommender systems?*

To address this issue, we investigated comprehensive comparisons with and without diversification signals, respectively. As illustrated in Figure 1, surprised to recognize that the performance in accuracy comes at an obvious sacrifice while diversification signals are present, e.g., LightGCN, DGRec, and BUIR. This means that the imbalance between accuracy and diversity is highly tricky (Peng et al. 2024). A consequence of this may be that recommended items contain many less popular items, which may not always be preferred by the user. Therefore, the recommendation models must balance the correlation between the user preferences for items and diversity. Considering this point, a follow-up question then emerges: *Can we emancipate graph learning from only accuracy-promoted direction and steer in a direction driven by accuracy and diversity to strive for an ideal tradeoff between them?*

In light of the above questions, we investigate in depth how to better balance diversity and accuracy by injecting diverse signals into an end-to-end GCL model and propose a novel **g**raph **c**ontrastive **l**earning model for **div**erse recommendations (DivGCL). On one hand, DivGCL improves the InfoNCE objective by introducing Gaussian noise and comparing the view enhanced only once with the original image to learn more informative and evenly distributed node representations. On the other hand, DivGCL designs a determinant point process (DPP) likelihood-based loss to better balance the diversity and accuracy of representations, while jointly optimizing it with advanced GCL-wise objectives. Extensive experiments conducted on four public datasets validate the encouraging superiority of our DivGCL in comparison with the state-of-the-art GNN-based and GCL-based models and the indispensability of two components in DivGCL. The main contributions of the work include:

- We formulate the item recommendation objective from a set selection perspective and then design a DPP likelihood-based loss for the accuracy and diversity-promoting recommendation.
- We improve and simplify noise-augmented objective by introducing Gaussian noise, and then develop an end-to-end GCL model for the diversity-promoting recommendation, referred to as DivGCL.
- We perform extensive experiments on four public datasets, demonstrating the strength of DivGCL compared with several state-of-the-art approaches. In contrast, DivGCL achieves significant performance improvements up to 18.49%, 24.39%, and 23.47% on ML-1M in terms of Recall@20, NDCG@20 and T@20.

2 Related Work

Diversified Recommendation

The extensive diversification research in recommendations offline can be mainly divided into two categories: post-processing, and learning to rank (LTR). The pioneering work on post-processing approaches is Maximal Marginal Relevance (MMR) (Carbonell and Goldstein 1998) and MSD (Borodin et al. 2017), in which the post-processing module is appended after the generation of recommended candidates. Subsequently, heuristics methods (Qin and Zhu 2013; Ashkan et al. 2015; Sha, Wu, and Niu 2016) often adopt the form of a submodular objective function on top of a scheme similar to marginal relevance. With follow-up studies, recently, DPP (Kulesza and Taskar 2012) promises to promote diversity while maintaining accurate performance for the recommender system, which allows many researchers to strive to replace the heuristics in post-processing based mothods (Chen, Zhang, and Zhou 2017, 2018; Wilhelm et al. 2018; Liu et al. 2019; Gan et al. 2020; Qiao et al. 2020; Liu et al. 2020). However, the imperfection of such methods gradually emerged because the re-rank mechanism for post-processing is independent of the generation phase of candidates and leads to suboptimal recommendation results. Then, most practitioners attempt to obtain an optimal ranking for each user by way of learning, namely Learning-To-Rank (LTR), which aims to directly generate a recommendation sequence for each user instead of predicting the score value for each candidate item. For example, Cheng et al. (Cheng et al. 2017) proposed a learning-based collaborative filtering algorithm (DCF) to seek more satisfactory diversity while maintaining accuracy. However, for such LTR methods, it is pretty tricky to build an appropriate listwise dataset. Thus, whether it is a post-processing strategy or an LTR method, the correlation between items is ignored, resulting in unsatisfactory results.

Moreover, an end-to-end diversified recommendation model DGCN (Zheng et al. 2021) is developed simultaneously considering neighbor exploring rebalanced, category-enhanced negative sampling and adversarial learning, which perfectly couples the candidate generation and post-processing stage. Furthermore, Yang et al. (Yang et al. 2023) propose DGRec, diversifying the GNN-based recommender system by directly improving the embedding of users/items in well-designed diversified retrieval generation. Although these studies can achieve effective end-to-end recommendations, one drawback is that a category of items is required, which is not friendly enough for datasets like Yelp2018 and then limits their feasibility.

Graph Contrastive Learning in Recommendation

Recently, inspired by the powerful success of contrastive learning in CV and NLP fields (Chen et al. 2020; He et al. 2020; Grill et al. 2020), the most promising research integrates contrastive learning (CL) (Wu et al. 2021; Lee et al. 2021; Zhou et al. 2021; Yu et al. 2022, 2023; Lin et al. 2022; Wang et al. 2022) into graph-based models for recommendation by constructing self-supervised tasks, mitigating the tricky issue of high sparsity in observed positively-related interactions (Liu et al. 2021). Motivated by MoCo (He et al. 2020), a self-supervised framework BIUR (Lee et al. 2021) is proposed based on an asymmetric structure to capture user/item informative representations only on positively-related instances to deal with the key challenges in one-class collaborative filtering (OCCF) problem. Based on pioneering Siamese networks (Chen and He 2021), SelfCF (Zhou et al. 2023) simplifies BUIR and augments the output embeddings generated in observed positive-only instances instead of perturbing the raw input data. Furthermore, elaborately-designed SGL (Wu et al. 2021) and SimGCL (Yu et al. 2022) perform graph structure-level data and representation-level data augmentation by conducting random dropout operations and adding random noise to enhance accuracy, respectively. Recently, an effective DirectAU (Wang et al. 2022) that directly optimizes a new learning objective function from the perspective of alignment and uniformity, which works better than state-of-the-art methods.

Although such works produce relatively ideal performance in accuracy, none of them considers recommendation results from the perspective of diversity, resulting in limited benefits. Therefore, we will study the feasibility of DPP-based objectives and perform a joint CL-optimized objective to optimize the final learning objective.

3 Method

In this section, we develop a graph contrastive learning framework for diverse recommendations, referred to as DivGCL. As illustrated in Figure 2, DivGCL is mainly characterized by two recommendation-optimized objectives: an advanced CL-based objective focusing on an accuracy-targeted recommendation for mining more informative and evenly-distributed node representations, and a tradeoff-targeted objective that focuses on the accuracy and diversity-promoting recommendation to drive more diverse representations.

DPP Likelihood-based Loss

DPP, as a probabilistic model, has the powerful ability to capture the negative correlations among various items. Assume that discrete $\mathcal{I}$ denotes a ground set of N items $\{1, 2, \ldots, N\}$, a DPP $\mathcal{P}$ (Kulesza and Taskar 2010) on $\mathcal{I}$ models a probability distribution over the power set of $\mathcal{I}$, $2^{\mathcal{I}}$. An **L-ensemble** can define a DPP via a $N \times N$ positive semi-definite kernel matrix $\mathbf{L}$, such that for any set $Y \subseteq \mathcal{I}$:

$$\mathcal{P}_L(Y) \propto \det(\mathbf{L}_Y), \tag{1}$$

where $\det(\cdot)$ represents the determinant of a matrix. $\mathbf{L}_Y = [\mathbf{L}_{i,j}]_{i,j \in Y}$ is obtained by indexing by the rows and columns of $\mathbf{L}$ that corresponds to elements in Y, then Eq. (1) can be rewritten by taking two items i and j for example:

$$\mathcal{P}_L(Y)(\{i,j\}) \propto \begin{vmatrix} \mathbf{L}_{i,i} & \mathbf{L}_{i,j} \\ \mathbf{L}_{j,i} & \mathbf{L}_{j,j} \end{vmatrix} = \mathbf{L}_{i,i}\mathbf{L}_{j,j} - \mathbf{L}_{i,j}\mathbf{L}_{j,i}, \tag{2}$$

which implies that a large $\mathbf{L}_{j,i}$ reduces the probability of i and j appearing together if we regard the entries of the kernel $\mathbf{L}$ as measurements of similarity between pairs of elements in $\mathcal{I}$. In line with our goals, the repulsive correlations between pairs of items are modeled, and then $\mathcal{P}_L$ favors sets of items with ideal high-quality while diversity. Furthermore, as depicted in previous work (Kulesza and Taskar 2011, 2012), $\mathbf{L}_{i,j} = \hat{y}_i \phi_i \phi_j^\top \hat{y}_j$ holds, in which scalars $\hat{y}_i$ and $\hat{y}_j$ describe predicted preferences toward items i, j and row vector $\phi_i \in \mathbb{R}^{1 \times D}$ ($\|\phi_i\|_2 = 1$) represents the normalized feature representation of item i. Formally, we can compute a kernel matrix for user u:

$$\mathbf{L}_u = \mathrm{Diag}(\hat{y}_u) \cdot S_u \cdot \mathrm{Diag}(\hat{y}_u), \tag{3}$$

where $\mathbf{S} = [\mathbf{S}_{i,j}]_{i,j \in \mathcal{I}}$ denotes a $N \times N$ similarity matrix measuring the similarity between pairwise items ($S_{i,j} = \phi_i \phi_j^\top \in [-1, 1]$). In this way, a DPP can perfectly balance the two objectives of high accuracy and diversity by modeling the quality and similarity into a unified model.

Based on the above analysis, we formulate the item recommendation objective from a set selection perspective, i.e., regarding the items that a user interacts with (positive items) as a DPP-distributed set, which can be written as:

$$\mathcal{L}_{dpp} = \sum_{u=1}^{M} \frac{-\log\det\left(\mathbf{L}_{u_{T_u}}\right)}{-\log\det\left(\mathbf{L}_{u_{\tilde{T}_u}}\right)}. \tag{4}$$

Where M represents the number of users; $T_u, \tilde{T}_u$ are considered as observable positive and unobservable negative samples for user u. Considering the expensive computational cost of calculating the personalized kernel $\mathcal{L}_u$ for each user and only a small subset of items relevant to a specific user, if we consider the entire item set and calculate personalized kernels for all users in practical computation, it would require excessive space complexity and computational cost. To alleviate this problem, based on the DPP theory mentioned above, DivGCL divides the probability based on user personal kernels into quality and diversity parts according to Eq. (2) and Eq. (3), by rewriting the probability of the positive item subset in Eq.(4) as:

$$\log\det\left(\mathbf{L}_{u_{T_u}}\right) = \sum_{i \in T_u} \log\left(\hat{y}_{u,i}^2\right) + \log\det\left(\mathbf{S}_{u_{T_u}}\right). \tag{5}$$

Similarly, the log-likelihood of the negative item set can also be calculated. Before calculating the log-likelihood of the negative item set, DivGCL defines $\mathbf{L}_{u_{\tilde{T}_u}}$ for denoting the DPP kernel matrix of negative sample pairs, formally:

$$\mathbf{L}_{u_{\tilde{T}_u}} = \mathrm{Diag}(\hat{y}_{u_{\tilde{T}_u}}) \cdot S_{u_{\tilde{T}_u}}^{-1} \cdot \mathrm{Diag}(\hat{y}_{u_{\tilde{T}_u}}), \tag{6}$$

then, through equation derivation, the probability of the negative item subset in Eq. (4) can be rewritten as:

$$\log\det\left(\mathbf{L}_{u_{\tilde{T}_u}}\right) = \sum_{i \in \tilde{T}_u} \log\left(\hat{y}_{u,i}^2\right) - \log\det\left(\mathbf{S}_{u_{\tilde{T}_u}}\right). \tag{7}$$

Plugging (5) and (7) into (4) can yield the final form of our diverse DPP loss:

$$\mathcal{L}_{dpp} = \sum_{u=1}^{M} \frac{\sum_{i \in T_u} \log\left(\hat{y}_{u,i}^2\right) + \log\det\left(\mathbf{L}_{u_{T_u}}\right)}{\sum_{j \in \tilde{T}_u} \log\left(\hat{y}_{u,j}^2\right) - \log\det\left(\mathbf{L}_{u_{\tilde{T}_u}}\right)}. \tag{8}$$

In this way, jointly learning two objectives focusing on different aspects provides new insights into promoting diversity in recommendations. Please note that in some applications, cardinality constraints are usually applied to T_u to return a fixed-size subset with the highest probability, then obtaining

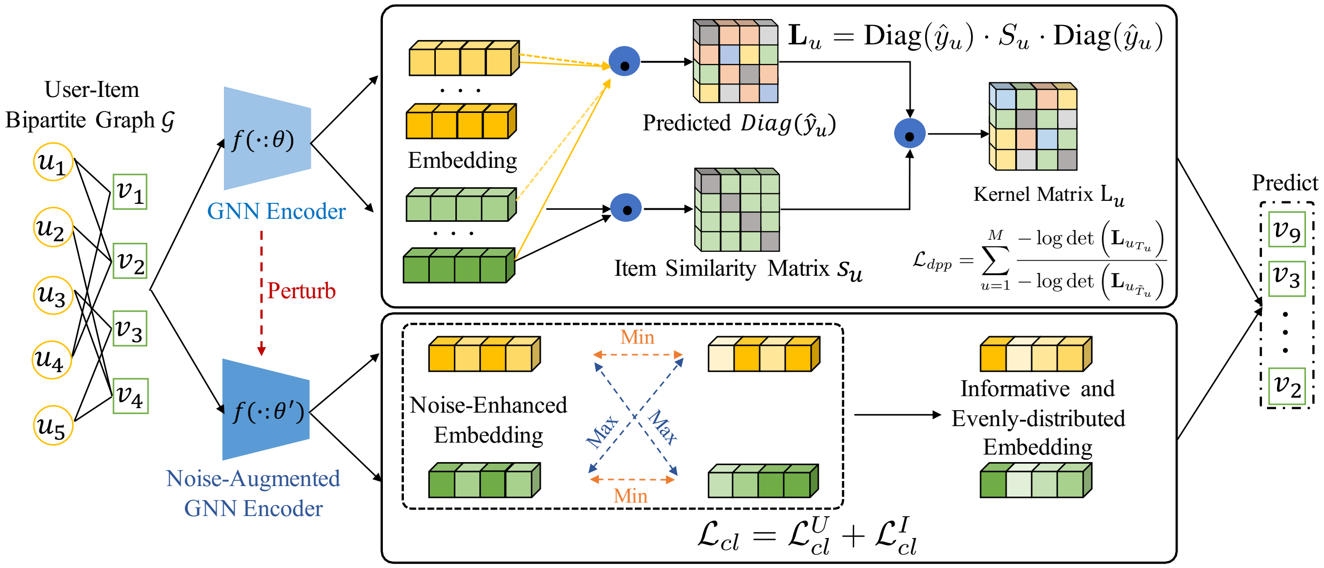


Figure 2: Illustration of the proposed DivGCL, a novel framework of graph contrastive learning for diverse recommendation beyond accuracy-diversity.

a k-DPP (Kulesza and Taskar 2012). In our experimental sets, we select all of positive samples interacted by a user while randomly selecting the corresponding number of negative items of the user for computing loss function. In this setting, a user's preference for an item and the diversity factor among items are considered.

Advanced Noise-based Data Augmentation

Basic GCL-based objective. Through the analysis of previous work (Yu et al. 2022), we can conclude that by contrasting node embeddings of two augmented views, not directly involving the embeddings from the main-view, i.e., the original graph contrastive recommendation models can reach a better performance.

Advanced GCL-based objective. However, Xia et al. put forward that the amount of information represented by the main-view is quite important (Xia et al. 2022). With this in mind, we enhance the performance of contrastive learning by introducing main-view and utilizing node representations of one main-view and one augmented-view, which is also experimentally efficient. Moreover, we innovatively use Gaussian noise to obtain augmented-view. Formally, the CL-loss for the user side is:

$$\mathcal{L}_{cl}^{U} = \sum_{u \in \mathcal{U}} -\log \frac{\exp(s(z_u, z_u^{'})/\tau)}{\sum_{v \in \mathcal{U}, v \neq u} \exp(s(z_u, z_v^{'})/\tau)}, \tag{9}$$

where z, z' denotes user/item embedding of main-view and augmented-view, and the cosine similarity function $s(z, z')$ measures the similarity between any two vectors. $\tau > 0$ is the hyper-parameter representing temperature. Analogously, the CL-loss for the item side $\mathcal{L}_{cl}^{I}$ can be derived, then the loss of contrastive learning task is computed as $\mathcal{L}_{cl} = \mathcal{L}_{cl}^{U} + \mathcal{L}_{cl}^{I}$. Finally, the loss function for jointly learning schema with trade-off hyperparameters α and λ is obtained as:

$$\mathcal{L} = \mathcal{L}_{cl} + \alpha \mathcal{L}_{dpp} + \lambda \parallel \Theta \parallel_2^2, \tag{10}$$

in which, Θ,α and λ are hyperparameters. In practice, we select all interacted items as a user's positive items set, i.e., T_u, and randomly sample the same set of negative elements from those with which the user has not interacted, i.e., $\tilde{T}_u$, then use Adam to optimize the joint loss.

Complexity Analysis

Let $|\mathcal{U}|$, $|\mathcal{I}|$ and $|\mathcal{R}|$ mean the number of user, item, and edges in graph $\mathcal{G}$. B and d denotes the batch size and the embedding size, and N represent the number of nodes in a batch. s represents the number of epochs and l is the number of GCN layers. Note that we train on a sampling of users. The training process is shown in Algorithm 1. The complexity of DivGCL includes three parts:

Algorithm 1: Training Process of DivGCL

```
Input: $\mathcal{G}, \mathcal{U}, \mathcal{I}, \mathcal{R}$.
Output: Top-$K$ diversified recommendation list $L$.
for each user $u$ in a mini-batch do
    Obtain positive samples $(u, i) \in \mathcal{R}$ Obtain an
      equal number of negative samples $(u, j) \notin \mathcal{R}$ by
      random sampling
    Initialize $\mathbf{e}_u^{(0)}, \mathbf{e}_i^{(0)}$ using Xavier initialization,
      $\forall u \in \mathcal{U}, \forall i \in \mathcal{I}$
    Generate $\mathbf{e}_u$ and $\mathbf{e}_i$ through LightGCN encoder
    Normalize $\mathbf{e}_u$: $\mathbf{e}_u = \frac{\mathbf{e}_u}{||\mathbf{e}_u||}$; $\mathbf{e}_i$: $\mathbf{e}_i = \frac{\mathbf{e}_i}{||\mathbf{e}_i||}$
    Calculate DPP likelihood-based loss using Eq. (8)
    Calculate contrastive loss using Eq. (9)
    Calculate total loss using Eq. (10)
```

- Encoding. The time complexity of the graph convolution encoder LightGCN is $O(|\mathcal{R}|sld\frac{|\mathcal{U}|}{B})$.
- Evaluating $\mathcal{L}_{cl}$. As formulated in Eq. (9), only user-item positive pairs are used. The complexity of the numerator and denominator for the user side and item side are both $O(s(B\frac{|\mathcal{R}|}{|\mathcal{U}|}d + B\frac{|\mathcal{R}|}{|\mathcal{U}|}Nd))$.
- Evaluating $\mathcal{L}_{dpp}$. The complexity of the numerator in Eq. (8) is similar to that of the denominator, and we only analyze the numerator. The negative samples equal to the number of positive ones are required. For the quality and diversity parts, the complexity of is linear to $\log(Bd\frac{|\mathcal{R}|}{|\mathcal{U}|})$ and $\frac{|\mathcal{R}|}{|\mathcal{U}|}d^2$, respectively.

Therefore, the total complexity of DivGCL is directly related to d, i.e., $O(sB(d\frac{|\mathcal{R}|}{|\mathcal{U}|}(\frac{l|\mathcal{U}|^2}{BB} + N) + \log(Bd\frac{|\mathcal{R}|}{|\mathcal{U}|})))$. d is generally controllable and not too large, thus the efficiency of the DivGCL can be guaranteed.

4 Experiments

Experimental Settings

Datasets. To verify the performance of DivGCL, we perform comprehensive experiments on four widely used public datasets with different domains, including MovieLens-1M (ML-1M), Beauty, AMiner, and Yelp2018. As for the Yelp2018 dataset, we follow the split ratio of datasets in SimGCL, i.e., the ratio 7:1:2 for the training set, validation set, and test set. For Beauty and ML-1M datasets, we do the same segmentation and only retain the users with interaction records greater than 10, where the main category is selected as the final category. More specific statistics of the datasets are presented in Table 1. The "-" in Table 1 and Table 2 indicates that the dataset lacks category information, so no comparison with the DGRec (Yang et al. 2023) is performed.

Dataset	ML-1M	Beauty	AMiner	Yelp2018
#Users	6,040	8,159	5,340	31,668
#Items	3,629	5,862	14,967	38,048
#Inter	836,478	98,566	163,084	1,561,406
Avg/user	138.49	12.08	30.54	49.31
Avg/item	230.50	16.81	10.90	41.65
Category	41	18	-	-
Avg/Category	143.07	325.67	-	-
Density	3.82%	0.21%	0.21%	0.13%
Sparsity	96.18%	99.79%	99.79%	99.87%

Table 1: Datasets and their description.

Baselines and Evaluation Metrics. We evaluate DivGCL by comparing the performance with accuracy-driven and diversity-driven GNN baselines, represented by SGL (Wu et al. 2021), SimGCL (Yu et al. 2022), DGRec (Yang et al. 2023) and so on. Here note the latest work on recommendation balance in diversity recommendation model (Peng et al. 2024) is not compared due to two reasons: 1) It introduces user utility that is inconsistent with DivGCL as research motivation and evaluation metric; 2) it is application-oriented and lacks evaluation on public datasets. We perform different values of K in the range of $\{10, 20, 50\}$ on evaluated experiments, and only present the results of $K = 20$ for simplicity. Note that the tradeoff metric is referred to as T.

Implementation Settings. All our experimental comparisons are performed on an Ubuntu server with hardware settings (18.04.4 LTS server with Intel(R) Xeon(R) Gold 6240 16-Core Processor, and GeForce RTX 3090 GPU) and software settings (Python 3.9.7 PyTorch). Specifically, for fair comparisons, we implement all experiments in the framework in SimGCL (Yu et al. 2022). In particular, the training batch here is sampled according to the user. If Recall@20 on the validation dataset does not increase for the next 10 epochs during the training process, early stopping is used. For all models, we employ Xavier initialization to initialize the parameters and Adam optimizer with a learning rate $1e^{-3}$ as the default optimizer. The backbone in DivGCL is a 2-layer LightGCN that propagates the user- item interactions. As for the setting of hyper-parameters in all baselines, we set them as the value suggested by their original work.

Overall Performance Comparison

We begin with a comparison of DivGCL with other baselines, and the overall recommendation performance of the proposed DivGCL is presented in Table 2. The training batch size on the ML-1M, Beauty, AMiner, and Yelp2018 datasets is set to 256, 64, 256, and 128. The improvement is abbreviated as %Imp, which is calculated through the relative improvement over the sub-optimal model, i.e., SimGCL. Accordingly, our findings are organized as below:

- Overall, DivGCL outperforms all competitive methods in terms of T@20 representing the tradeoff between accuracy and diversity, as indicated by the statistical results in Table 2. Specifically, DivGCL achieves a relatively-ideal improvement over sub-optimal approach SimGCL or SGL w.r.t. T@20 by 23.47%, 2.78%, 2.20%, 9.16% or 9.34% on ML-1M, Beauty, Yelp2018, and AMiner, respectively. This forcefully demonstrates the excellence of DivGCL in balancing diversity and accuracy.
- DivGCL shows great power in accuracy over other competing methods. Remarkably, DivGCL increase by 18.49% and 24.39%, 5.46% and 6.2%, 6.52% and 5.43%, 10.97% and 12.56% in Recall@20 and NDCG@20 on ML-1M, Beauty, Yelp2018, and AMiner. However, interestingly, DivGCL is not optimal in terms of ILAD@20 and ILMD@20 metrics on all four datasets. Besides, all baseline methods achieve superior accuracy, while diversity is significantly affected, and vice versa. From a diverse perspective, our proposal is slightly inferior to SimGCL in terms of the ILAD@20 and ILMD@20 metrics on Beauty dataset. Note here that although we conduct research on diversified recommendations, our goal is to enhance the tradeoff of recommendations instead of the improvement in a single indicator. Given the performance increase gained in Recall and NDCG, such slight inferiority in ILAD and ILMD is still perfectly

Dataset	Metric	Method							%Imp
		BPR-MF	LightGCN	SGL	BUIR	SimGCL	DGRec	DivGCL	
ML-1M	Recall@20	5.59±0.0082	6.03±0.0008	1.91±0.0017	6.11±0.0015	5.82±0.0005	3.56±0.0005	**7.24±0.0013**	18.49%
	NDCG@20	2.11±0.0025	2.51±0.0002	1.16±0.0011	2.84.±0.0020	3.28±0.0012	1.61±0.0003	**4.08±0.0007**	24.39%
	ILAD@20	4.15±0.0060	0.22±0.0001	**92.63±0.0004**	3.94±0.0047	80.59±0.0070	12.53±0.0004	85.10±0.0006	-
	ILMD@20	1.26±0.0018	0.41±0.0001	**71.07±0.0013**	0.36±0.0015	21.25±0.008	0.37±0.0001	24.24±0.0027	-
	T@20	3.16±0.0035	0.34±0.0058	3.02±0.0027	2.92±0.0410	8.31±0.0018	2.94±0.0003	**10.26±0.0016**	**23.47%**
Beauty	Recall@20	9.48±0.0016	11.58±0.0021	12.26±0.0006	10.83±0.0008	5.35±0.0016	9.73±0.0098	**12.93±0.0019**	5.46%
	NDCG@20	4.95±0.0008	6.10±0.0007	6.61±0.0009	5.51±0.0005	3.03±0.0007	4.65±0.0041	**7.02±0.0011**	6.2%
	ILAD@20	63.24±0.0085	54.57±0.0348	75.12±0.0166	39.02±0.0079	**88.58±0.0055**	25.50±0.1545	73.64±0.0101	-
	ILMD@20	28.58±0.0072	18.79±0.0163	35.42±0.0121	9.49±0.0036	**42.72±0.0050**	2.92±0.0614	23.85±0.0069	-
	T@20	12.48±0.0021	14.24±0.0051	16.11±0.0021	12.22±0.0015	7.87±0.0019	6.63±0.0011	**16.57±0.0026**	**2.78%**
AMiner	Recall@20	16.05±0.0036	8.86±0.0019	9.74±0.0027	13.41±0.0014	15.57±0.0049	-	**17.81±0.0052**	10.97%
	NDCG@20	7.97±0.0014	4.39±0.0012	4.66±0.0015	6.41±0.0004	8.04±0.0016	-	**9.05±0.0019**	12.56%
	ILAD@20	51.76±0.0134	23.11±0.0129	**83.96±0.0106**	27.24±0.0073	71.36±0.0197	-	67.91±0.0020	-
	ILMD@20	12.49±0.0065	2.55±0.0022	**35.82±0.0311**	1.50±0.0006	4.42±0.0033	-	5.68±0.0012	-
	T@20	17.49±0.0037	8.74±0.0029	12.85±0.0031	11.73±0.0015	17.99±0.0026	-	**19.67±0.0037**	**9.34%**
Yelp2018	Recall@20	5.20±0.0002	5.76±0.0002	6.14±0.0012	4.64±0.0006	6.44±0.0002	-	**6.86±0.0002**	6.52%
	NDCG@20	4.25±0.0002	4.72±0.0002	5.04±0.0012	3.78±0.0005	5.34±0.0001	-	**5.63±0.0001**	5.43%
	ILAD@20	22.28±0.0131	22.49±0.0011	55.94±0.0075	6.92±0.0037	58.35±0.1976	-	**58.63±0.0013**	-
	ILMD@20	7.81±0.0062	4.95±0.0002	**25.52±0.0114**	0.25±0.0002	15.74±0.0086	-	17.42±0.0006	-
	T@20	7.19±0.0009	6.62±0.0003	9.83±0.0014	3.87±0.0014	8.63±0.0003	-	**10.73±0.0002**	**9.16%**

Table 2: Overall comparison of recommendation performance with representative baseline on four datasets (% is omitted). Specifically, the best results and the sub-optimal results are highlighted in **bold** and in underline, respectively. We average the experimental results repeated with 5 random seeds and report the improvement over the suboptimal baseline (% Improve).

Model	ML-1M					Beauty				
	Recall@20	NDCG@20	ILAD@20	ILMD@20	T@20	Recall@20	NDCG@20	ILAD@20	ILMD@20	T@20
SimGCL	5.82	3.28	80.59	21.25	8.31	5.35	3.03	88.58	42.72	7.87
DivGCL	**7.24**	**4.08**	**85.10**	**24.24**	**10.26**	**12.93**	**7.02**	**73.64**	**23.85**	**16.57**
-DPP	7.10	3.97	85.16	24.34	10.05	12.59	6.92	73.85	24.12	16.27
-CL	5.78	2.98	38.02	11.11	7.43	7.41	4.03	70.59	17.16	10.12
%Imp	24.40%	12.03%	5.60%	14.07%	23.47%	141.68%	131.68%	-	-	100.81%

Table 3: Analysis on ablation study of DivGCL on ML-1M and Beauty.

acceptable. We attribute the superiority to two aspects: (1) Through powerful noise-based graph augmentation, DivGCL is able to better embed more informative representations of users or items; (2) benefiting from DPP, DivGCL can generate more relevant while diverse items.

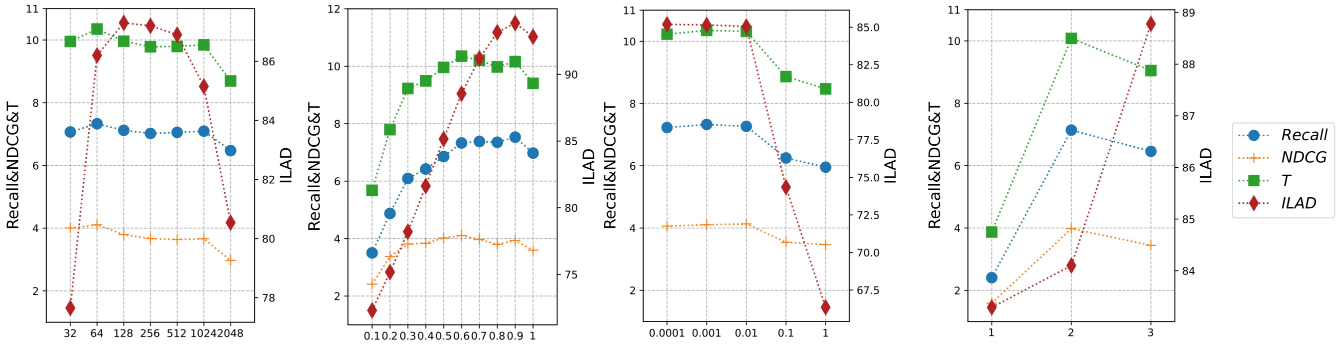
- A baseline method does not perform well on all datasets. For example, except for DivGCL, SimGCL is the best method on ML-1M and AMiner but is slightly inferior to SGL on the other three datasets. We also found that accuracy has a more obvious impact on the tradeoff indicator.

Ablation Study

As shown in Table 3, in this section, we explored the effect of two components of the DivGCL on representative datasets (ML-1M and Beauty) datasets. In particular, "-CL" denotes the final results of DivGCL without the noise-based data augmentation (i.e., only the Eq. (8)) . "-DPP" stands for the model by removing the DPP likelihood-based loss designed in this work (i.e., only the Eq. (9)).

Effect of simple noise-based data augmentation. Based on the statistical results in Table 3, we can find that (1) there is a slight reduction when we remove the "-DPP" module. Numerically, the reported results on the Beauty dataset decrease from 12.93 and 7.02 to 12.59 and 6.92 in Recall@20 and NDCG@20, respectively. This demonstrates that although noise-based data augmentation is simple, it has indeed made a dramatically significant contribution to DivGCL, which is an important finding of the work. That is, to some extent, noise-based data enhancement can ensure that the diversity is ideal while ensuring considerable accuracy. This is attributed to that Gaussian noise-based data augmentation can make the representation more uniform. (2) Although the reported results of "-DPP" on the ML-1M and Beauty datasets have decreased in Recall@20 and NDCG@20, it is still better than SimGCL on all metrics, which shows the benefits of data augmentation that reinforces the interaction between users and items. (3) Besides, according to two indicators of diversity, i.e., ILAD and ILMD, "-DPP" increased instead. This again illustrates the imbalance between diversity and accuracy. However, the final balance metric T@20 lags behind DivGCL.

Effect of DPP likelihood-based objective. From Table 3, we have some observations: (1) In contrast, when we remove the "-CL" module, the performance of DivGCL is reduced



(a) Embedding size *d* **(b) Perturbation strength *η*** **(c) Weight *α*** **(d) The number of convolution layers *l***

Figure 3: Impact analysis of DivGCL on ML-1M dataset with different hyper-parameter settings, including the embedding size d, perturbation strength η, weight α, and the number of graph convolution layers l.

even more, which demonstrates that DPP aims to balance diversity and accuracy rather than to complete the recommendation task. (2) Moreover, "-CL" suffers from a decrease in both diversity and accuracy, Moreover, this method suffers from a decrease in both diversity and accuracy, which is sufficient to prove that the DPP likelihood-based objective can play a significant role when used together with the Gaussian noise-based objective. (3) On some individual datasets, e.g. Beauty, the accuracy of only DPP is still better than SimGCL, which in a sense demonstrates the importance of user-item interactions either in alignment or uniformity.

Hyperparameter Analysis

Impact of the embedding size d. Due to the consideration between memory cost and performance, we tune the embedding size among $\{32, 64, 128, 256, 512, 1024\}$. Typically, the embedding size in most Collaborative Filtering (CF) methods is set to 64 (Zhu et al. 2023). Consistently, Figure 3. (a) shows the model performs optimally when d is 64, which indicates that a dimension that is too small may result in insufficient mining of high-order information, while a dimension that is too large may lead to overfitting.

Impact of the perturbation strength η. We vary η in $[0-1]$ in steps of 0.1 to investigate the influence of η. From Figure 3. (b), we can seen that as η grows in the range of $[0.1 - 0.6]$, each metric increases, which proves the effectiveness of adding perturbation to representations. Specifically, within a certain range, the stronger the perturbation, the more useful information the model can learn. However, accuracy-wise metrics, i.e., Recall@20 and NDCG@20, and balance metric T@20 gradually decrease as η further increases, which shows that excessive η can lead to the destruction of original information, preventing DivGCL from capturing more signal between main view and augmented view. Therefore, the optimal η on the ML-1M is 0.6.

Impact of the weight α. As depicted in Figure 3. (c), as α gradually increases, both accuracy and balance metrics increase, while the diversity metric ILAD@20 continues to decrease. In specific, the fact that α rises from 0.0001 to 0.001 leads to a small decrease in diversity while a relatively high increase in accuracy, which ultimately produces increased T@20. However, diversity sharply decreases when α increases from 0.01 to 1, leading to a decrease in T@20. Similar to η, it can be concluded that increasing α within a certain range can work well. Therefore, the performance of DivGCL on ML-1M is optimal when α is equal to 0.01.

Impact of the number of graph convolution layers l. As l arises, the Recall@20, NDCG@20, and T@20 show a trend of increasing first and then decreasing from Figure 3. (d). With l increasing from 1 to 2, the Recall, NDCG, and T increase, indicating the importance of aggregating high-order neighborhood information. However, these three metrics all decrease with l increasing to 3, while ILAD@20 significantly increases. This demonstrates that more convolution layers make the embedding vectors of users and items homogeneous, resulting in excessive diversity or even a sharp decline in final recommendations. Therefore, DivGCL resorts to a graph encoder with two convolutional layers as its backbone, allowing it to fully utilize high-order neighborhood signals while avoiding oversmoothing of embeddings.

5 Conclusion

In this paper, we first point out the tricky issue of unbalanced accuracy-diversity in diverse recommendations and then investigate how accuracy-targeted GCL recommendations work well. As a solution, we propose DivGCL for achieving a balance between diversity and accuracy. Specifically, we develop a novel DPP likelihood-based loss by constructing a DPP kernel and then optimize it jointly with an advanced category-free GCL objective by Gaussian noise-based data augmentation. Extensive experiments on four public datasets show DivGCL can strike a trade-off beyond accuracy-diversity for diversification recommendations. In the future, we aim to conduct a promising research by performing an effective fusion of pre-trained language model and GNNs for more real-world diverse recommendations.

Acknowledgements

This work was supported by the China Postdoctoral Science Foundation (2023M731951), the Natural Science Foundation of Shandong Province (No. ZR2023MF007), National Natural Science Foundation of China (No. 62372242), State Key Laboratory of New Software Technology Open Project (No. KFKT2024B50). Dr. Xuyun Zhang is the recipient of an ARC DECRA (project No. DE210101458) funded by the Australian Government.

References

Ashkan, A.; Kveton, B.; Berkovsky, S.; and Wen, Z. 2015. Optimal Greedy Diversity for Recommendation. In *IJCAI*, volume 15, 1742–1748.

Borodin, A.; Jain, A.; Lee, H. C.; and Ye, Y. 2017. Max-sum diversification, monotone submodular functions, and dynamic updates. *ACM Transactions on Algorithms*, 13(3): 1–25.

Carbonell, J.; and Goldstein, J. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *SIGIR*, 335–336.

Chen, L.; Zhang, G.; and Zhou, H. 2017. Improving the diversity of top-N recommendation via determinantal point process. In *LSRS Workshop*, 3.

Chen, L.; Zhang, G.; and Zhou, H. 2018. Fast greedy map inference for determinantal point process to improve recommendation diversity. In *NeurIPS*, 5627–5638.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *ICML*, 1597–1607.

Chen, W.; Liu, Y.; Zhang, Z.; Zhuang, F.; and Zhong, J. 2024a. Modeling Adaptive Inter-Task Feature Interactions via Sentiment-Aware Contrastive Learning for Joint Aspect-Sentiment Prediction. In *AAAI*, volume 38, 17781–17789.

Chen, W.; Wu, Y.; Zhang, Z.; Zhuang, F.; He, Z.; Xie, R.; and Xia, F. 2024b. FairGap: Fairness-aware Recommendation via Generating Counterfactual Graph. *ACM Transactions on Information Systems*, 42(4): 1–25.

Chen, W.; Yuan, M.; Zhang, Z.; Xie, R.; Zhuang, F.; Wang, D.; and Liu, R. 2024c. FairDgcl: Fairness-aware Recommendation with Dynamic Graph Contrastive Learning. *arXiv preprint arXiv:2410.17555*.

Chen, X.; and He, K. 2021. Exploring simple siamese representation learning. In *CVPR*, 15750–15758.

Cheng, P.; Wang, S.; Ma, J.; Sun, J.; and Xiong, H. 2017. Learning to recommend accurate and diverse items. In *WWW*, 183–192.

Fan, W.; Wang, P.; Wang, D.; Wang, D.; Zhou, Y.; and Fu, Y. 2023. Dish-TS: A General Paradigm for Alleviating Distribution Shift in Time Series Forecasting. In *AAAI*, volume 37, 7522–7529.

Fan, W.; Zheng, S.; Yi, X.; Cao, W.; Fu, Y.; Bian, J.; and Liu, T.-Y. 2022. DEPTS: Deep Expansion Learning for Periodic Time Series Forecasting. In *ICLR*.

Fan Wang, W. L. T. F. X. L. Y. T. L. Q. X. Z., Chaochao Chen. 2024. CE-RCFR: Robust Counterfactual Regression for Consensus-Enabled Treatment Effect Estimation. In *SIGKDD*.

Gan, L.; Nurbakova, D.; Laporte, L.; and Calabretto, S. 2020. Enhancing Recommendation Diversity using Determinantal Point Processes on Knowledge Graphs. In *SIGIR*, 2001–2004.

Grill, J.-B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P.; Buchatskaya, E.; Doersch, C.; Avila Pires, B.; Guo, Z.; Gheshlaghi Azar, M.; et al. 2020. Bootstrap your own latent-a new approach to self-supervised learning. *NeurIPS*, 33: 21271–21284.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 9729–9738.

He, X.; Liao, L.; Zhang, H.; Nie, L.; Hu, X.; and Chua, T.-S. 2017. Neural collaborative filtering. In *WWW*, 173–182.

Kulesza, A.; and Taskar, B. 2010. Structured determinantal point processes. *NeurIPS*, 23: 1171–1179.

Kulesza, A.; and Taskar, B. 2011. Learning determinantal point processes. *UAI*, 419–427.

Kulesza, A.; and Taskar, B. 2012. Determinantal Point Processes for Machine Learning. *Machine Learning*, 5(2-3): 123–286.

Lee, D.; Kang, S.; Ju, H.; Park, C.; and Yu, H. 2021. Bootstrapping user and item representations for one-class collaborative filtering. In *SIGIR*, 317–326.

Lin, Z.; Tian, C.; Hou, Y.; and Zhao, W. X. 2022. Improving graph collaborative filtering with neighborhood-enriched contrastive learning. In *WWW*, 2320–2329.

Liu, W.; Su, J.; Chen, C.; and Zheng, X. 2021. Leveraging distribution alignment via stein path for cross-domain cold-start recommendation. *Advances in Neural Information Processing Systems*, 34: 19223–19234.

Liu, Y.; Xiao, Y.; Wu, Q.; Miao, C.; Zhang, J.; Zhao, B.; and Tang, H. 2020. Diversified interactive recommendation with implicit feedback. In *AAAI*, volume 34, 4932–4939.

Liu, Y.; Zhang, Y.; Wu, Q.; Miao, C.; Cui, L.; Zhao, B.; Zhao, Y.; and Guan, L. 2019. Diversity-promoting deep reinforcement learning for interactive recommendation. *ICCSE*, 132–139.

Peng, K.; Raghavan, M.; Pierson, E.; Kleinberg, J.; and Garg, N. 2024. Reconciling the accuracy-diversity trade-off in recommendations. In *WWW*, 1318–1329.

Qiao, M.; Yu, J.; Liu, T.; Wang, X.; and Tao, D. 2020. Diversified Bayesian Nonnegative Matrix Factorization. In *AAAI*, volume 34, 5420–5427.

Qin, L.; and Zhu, X. 2013. Promoting diversity in recommendation by entropy regularizer. In *IJCAI*, 2698–2704.

Sha, C.; Wu, X.; and Niu, J. 2016. A framework for recommending relevant and diverse items. In *IJCAI*, volume 16, 3868–3874.

Wang, C.; Yu, Y.; Ma, W.; Zhang, M.; Chen, C.; Liu, Y.; and Ma, S. 2022. Towards Representation Alignment and

Uniformity in Collaborative Filtering. In *SIGKDD*, 1816–1825.

Wang, F.; Chen, C.; Liu, W.; Lei, M.; Chen, J.; Liu, Y.; Zheng, X.; and Yin, J. 2025. DR-VAE: Debiased and Representation-enhanced Variational Autoencoder for Collaborative Recommendation. In *AAAI*.

Wilhelm, M.; Ramanathan, A.; Bonomo, A.; Jain, S.; Chi, E. H.; and Gillenwater, J. 2018. Practical diversified recommendations on youtube with determinantal point processes. In *CIKM*, 2165–2173.

Wu, J.; Wang, X.; Feng, F.; He, X.; Chen, L.; Lian, J.; and Xie, X. 2021. Self-supervised graph learning for recommendation. In *SIGIR*, 726–735.

Xia, J.; Wu, L.; Chen, J.; Hu, B.; and Li, S. Z. 2022. Simgrace: A simple framework for graph contrastive learning without data augmentation. In *WWW*, 1070–1079.

Yang, L.; Wang, S.; Tao, Y.; Sun, J.; Liu, X.; Yu, P. S.; and Wang, T. 2023. DGRec: Graph Neural Network for Recommendation with Diversified Embedding Generation. In *WSDM*, 661–669.

Yang, Y.; Wu, L.; Hong, R.; Zhang, K.; and Wang, M. 2021. Enhanced graph learning for collaborative filtering via mutual information maximization. In *SIGIR*, 71–80.

Yu, J.; Xia, X.; Chen, T.; Cui, L.; Hung, N. Q. V.; and Yin, H. 2023. XSimGCL: Towards extremely simple graph contrastive learning for recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 36(2): 913–926.

Yu, J.; Yin, H.; Xia, X.; Chen, T.; Cui, L.; and Nguyen, Q. V. H. 2022. Are graph augmentations necessary? simple graph contrastive learning for recommendation. In *SIGIR*, 1294–1303.

Zhang, D.; Geng, Y.; Gong, W.; Qi, Z.; Chen, Z.; Tang, X.; Shan, Y.; Dong, Y.; and Tang, J. 2024. RecDCL: Dual Contrastive Learning for Recommendation. In *WWW*, 3655–3666.

Zhang, D.; Zhu, Y.; Dong, Y.; Wang, Y.; Feng, W.; Kharlamov, E.; and Tang, J. 2023. ApeGNN: Node-Wise Adaptive Aggregation in GNNs for Recommendation. In *WWW*, 759–769.

Zhang, S.; Yao, L.; Sun, A.; and Tay, Y. 2019. Deep learning based recommender system: A survey and new perspectives. *ACM computing surveys*, 52(1): 1–38.

Zhao, W. X.; Mu, S.; Hou, Y.; Lin, Z.; Chen, Y.; Pan, X.; Li, K.; Lu, Y.; Wang, H.; Tian, C.; et al. 2021. Recbole: Towards a unified, comprehensive and efficient framework for recommendation algorithms. In *CIKM*, 4653–4664.

Zheng, Y.; Gao, C.; Chen, L.; Jin, D.; and Li, Y. 2021. DGCN: Diversified recommendation with graph convolutional networks. In *WWW*, 401–412.

Zhou, C.; Ma, J.; Zhang, J.; Zhou, J.; and Yang, H. 2021. Contrastive learning for debiased candidate generation in large-scale recommender systems. In *SIGKDD*, 3985–3995.

Zhou, X.; Sun, A.; Liu, Y.; Zhang, J.; and Miao, C. 2023. Selfcf: A simple framework for self-supervised collaborative filtering. *ACM Transactions on Recommender Systems*, 1(2): 1–25.

Zhu, Y.; Cong, F.; Zhang, D.; Gong, W.; Lin, Q.; Feng, W.; Dong, Y.; and Tang, J. 2023. Wingnn: Dynamic graph neural networks with random gradient aggregation window. In *SIGKDD*, 3650–3662.