International Journal on Semantic Web and Information Systems
Volume 20 • Issue 1

Web Semantic-Based Robust Graph Contrastive Learning for Recommendation via Invariant Learning

Wengui Dai, College of Computer and Information Science, Southwest University, China*
Yujun Wang, College of Computer and Information Science, Southwest University, China

ABSTRACT

The use of contrastive learning (CL) in recommendation has advanced significantly. Recently, some works use perturbations in the embedding space to obtain enhanced views of nodes. This makes the representation distribution of nodes more even and then improve recommendation effectiveness. In this article, the authors provide an explanation on the role of added noises in the embedding space from the perspective of invariant learning and feature selection. Guided by this thinking, the authors devise a more reasonable method for generating random noises and put forward web semantic based robust graph contrastive learning for recommendation via invariant learning, a novel graph CL-based recommendation model, named RobustGCL. RobustGCL, randomly zeros the values of certain dimensions in the noise vectors at a fixed ratio. In this way, RobustGCL can identify invariant and variant features and then learn invariant and variant representations. Tests on publicly available datasets show that our proposed approach can learn invariant representations and achieve better performance.

KEYWORDS

Contrastive learning, Invariant learning, Graph collaborative filtering, Recommendation system

WEB SEMANTIC-BASED ROBUST GRAPH CONTRASTIVE LEARNING FOR RECOMMENDATION VIA INVARIANT LEARNING

The research and development of information technology has assisted lives in various aspects, such as education, healthcare, and transportation (Hu et al., 2022; Xu et al., 2021; Zhou et al., 2022; Deveci et al., 2023; Mohammed et al., 2022; Appati et al., 2022; Rajput et al., 2022; Tripathi & Kumar, 2022; Liu et al., 2022; Gupta et al., 2023; Alakbarov, 2022; Roy et al., 2022). In the age of data explosion, recommendation systems play a significant role (Li et al., 2022; Xiao et al., 2022; George & Lal, 2021; Zhang et al., 2023). It is important for collaborative recommendation to learn high-quality representations. The introduction of graph convolution network (GCN) (Hamilton, Ying & Leskovec, 2017; Kipf & Welling, 2016) enhances representations by offering a comprehensive method of integrating multi-hop neighbors into node representations (Berg, Kipf & Welling, 2017; He et al., 2020; Wang et al., 2019; Ying et al., 2018). Unfortunately, the following limitations affect GCN-based recommendation models:

DOI: 10.4018/IJSWIS.337962
*Corresponding Author

sparse supervision signal (Bayer et al., 2017; He & McAuley, 2016), skewed data distribution (Clauset, Shalizi & Newman, 2009; Milojević, 2010) and noises in interactions (Wang et al., 2021). Fortunately, contrastive learning techniques have been proven to be able to address the aforementioned issues in other fields because it can extract generic characteristics from big amounts of unlabeled data and generalize representations in a self-supervised manner (Chen et al., 2020; Gidaris, Singh & Komodakis, 2018; Oord, Li & Vinyals, 2018, Devlin et al., 2018; Lan et al., 2019).

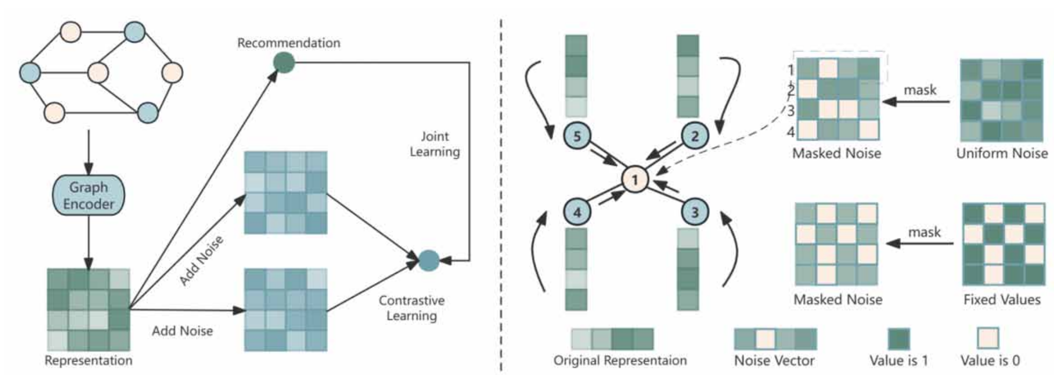
Naturally, introducing contrastive learning into recommendation models to address the aforementioned issues is a great idea. An increasing number of studies have gained significant success by applying contrastive learning to the recommendation (Lin et al., 2022; Wu et al., 2021; Yu et al., 2022a; Yu et al., 2022b; Zhou et al., 2020). It is worth mentioning that some works (Yu et al., 2022a; Yu et al., 2022b) use embedding perturbation to enhance the robustness of recommendation by directly adding random uniform noises to the original representations In recommendation models based on graph contrastive learning, which is less time-consuming but more efficient. These methods speculate that a more even representation distribution in a certain scope can enhance the capacity for generalization while maintaining the intrinsic qualities of nodes. Their experiment proved the correctness of this hypothesis. We believed that continuing to study along this line of thought is very promising.

Invariant learning is based on the invariance principle of causality, leveraging the invariant features of observed data from different environments, and ignoring spurious relationships. To capture representations with invariant prediction capacity across environments is the aim of invariant learning (Wang et al., 2022; Zhang et al., 2023). In theory, it can achieve guaranteed generalization in the case of distribution deviation, and has achieved great success in practice. InvPref (Wang et al., 2022) assumes the observed user actions are determined jointly by invariant tendency, which is the true tendency and variant preference influenced by environments.

The model structure of RobustGCL is shown in the left subfigure, while the specific approach of adding noises is shown in the right subfigure. Considering the messages propagated to the center node -- Node 1, RobustGCL aggregates these messages to it, and then obtains the original representation of Node 1. After that, the random noise vector, whose values of certain dimensions are zeros, will be imposed on the current original representation. The color green indicates a value of one and lower transparency indicates lower values.

In this paper, inspired by InvPref (Wang et al., 2022), the authors provide an explanation of the role of added noises in the embedding space from the perspective of invariant learning and feature selection. This paper proposes Web Semantic-Based Robust Graph Contrastive Learning for Recommendation via Invariant Learning (RobustGCL), which learns invariant representations by selecting representation-level features. From the perspective of invariant learning and feature selection, we consider that the added random noise vectors play the role of representation-level

Figure 1. The framework of RobustGCL model



feature selection and then find the invariant representation-level feature. The length of added noise vectors is very small and there are many dimensions in the noise vectors. Therefore, the values of each dimension are extremely small, and even the values in some dimensions are infinitely close to zeros. In our hypothesis, if the values of certain dimensions of the noise vectors are zeros, we believe that these dimensions are primarily used for learning invariant representation-level features, which mainly reflect the information of the node itself. In contrast, the rest of the dimensions, whose values are big enough, are primarily used for learning variant representation-level features mainly reflecting the environmental information. The environmental information of a node comes from other nodes related to it. For item nodes with high popularity, their representations should mainly come from their own information, while reducing information from the environment. For item nodes with low popularity, the importance of information from the node itself decreases, while that from the environment increases.

Based on the above assumptions, we devise a new way to generate random noises. Comprehensive test outcomes on actual datasets amply verify the efficacy of our approach, which boosts resilience against noisy interactions and recommendation accuracy, particularly for long-tail items. At the same time, we also demonstrated the rationality of our hypothesis through a series of extended experiments. In SimGCL, the experimental results reveal that, in comparison to Gaussian noise, the addition of uniform noise yields the most favorable outcomes. However, in our experiments, the performance of incorporating Gaussian noise can surpass that of uniform noise when specific dimensions in the Gaussian noise are modified to zeros. Subsequent experiments have demonstrated that similar operations on uniform noise can also enhance performance. Notably, the results indicate that performance improvement is achievable as long as the proportion of zeros is appropriately managed, irrespective of the type of noise. Furthermore, through the visualization of item embeddings learned by different models, a conspicuous color blocking phenomenon is observed. This suggests significant differences in certain features between popular and unpopular products. The below summarizes this work's efforts:

- We propose RobustGCL, devising a new way to generate random noises and proving its effectiveness on publicly available datasets.
- We explain the role of the added noises in recommendation based on graph contrastive learning from the perspective of invariant learning and feature selection.
- To show that the suggested RobustGCL makes sense, we run in-depth tests on publicly available datasets.

RELATED WORKS

Graph for Recommendation

It is crucial to learn vector representation (embeddings) of users and items with high quality in modern recommendation systems. Early method map gathered features that characterize nodes to gain their embeddings (Rendle et al., 2012). It is not possible for these methods to fully utilize latent collaboration signals in user-item interactions. NGCF (Wang et al., 2019) effectively injects the collaborative signal in low-order and high-order connectivities into the embedding process in an explicit manner by propagating embeddings on user-item bipartite graph. However, LightGCN (He et al., 2020) finds that nonlinear activation, as well as feature transformation in NGCF, which are components of standard GCN, is useless and even worse for graph recommendation because nodes in the user-item interaction graph merely contain an ID as input. LightGCN is a simple yet powerful recommendation model that just incorporates the GCN's most vital components, straightforward weighted sum aggregator.

Graph Contrastive Learning for Recommendation

Many graph contrastive learning models (Lin et al., 2022; Wu et al., 2021; Yu et al., 2022a; Yu et al., 2022b) employ LightGCN as the graph encoder because of its simple structure and effectiveness. To optimize model parameters, existing works usually adopt the pairwise Bayesian Personalized Ranking (BPR) loss (Rendle et al., 2012) as the main supervised learning task. It requires an observable interaction's prediction score to be higher than the prediction score of its unobserved counterparts. The representations in graph augmentation-based contrastive learning models are assumed to be invariant to perturbations in the partial structure. SGL (Wu et al., 2021) leverages perturbation of graph structure to obtain augmented views and InfoNCE (Hjelm et al., 2018) for contrastive learning. After obtaining the enhanced views of nodes, SGL treats the views of the same node in different dropout-based graph augmentation as the positive pairs, and the views of any different nodes as the negative pairs. Positive pair consistency is encouraged by the contrastive learning loss, whereas negative pair discrepancies are widened. However, SimGCL (Yu et al., 2022b) demonstrates through experimentation that graph augmentations are not necessary and instead directs the focus to the embedding space in order to achieve a more uniform representation distribution. At every layer, varied random uniform noises are applied to the present node embeddings. For each node, adding different random noises can obtain different augmented views.

Invariant Learning

Invariant Learning assumes that observed data originate from multiple environments, and the distribution of data in different environments is not entirely the same. Obtaining representations with invariant predictive capacity across environments is the goal of Invariant Learning. In theory, it can achieve guaranteed generalization in the case of distribution deviation and has succeeded in practice (Wang et al., 2022; Zhang et al., 2023). InvPref (Wang et al., 2022) assumes that the observed user actions are determined jointly by invariant tendency, which is the true tendency, and variant preference influenced by environments. Inspired by InvPref, we propose RobustGCL learning invariant representations by selecting representation-level features. RobustGCL sets the values of all dimensions of the noise vectors to ones and then randomly resets the values of certain dimensions to zeros at a fixed ratio. We believe that if the values of certain dimensions of the noise vectors are zeros, these dimensions are primarily used for learning invariant representation-level features and reflecting the information of the node itself. In contrast, the rest of dimensions whose values are big enough are used for learning variant representation-level features, mainly reflecting the environmental information.

METHOD

We introduce the proposed RobustGCL in three parts in this section. To begin with, we introduce the base graph collaborative filtering method, outputting the original representations for supervised learning and the final representations for recommendation. Next, the previous noise-based data augmentation for contrastive learning and a devised way of noise-based data augmentation in RobustGCL are introduced. Finally, to integrate the primary supervised task with the self-supervised work, a multi-task learning technique is used. The overall framework of RobustGCL and the way of data augmentation are depicted in Figure 1.

Preliminaries

A set is represented by capital letters, and an element of the associated set is represented by lowercase letters. In particular, the set of users is represented by U , and the set of items by I . The node set $\mathcal{V}$ includes all users and items. The edge set $\mathcal{E}$ represents all observed interactions. $E^{(0)} \in R^{|\mathbb{V}| \times d}$

is the initialized random node embeddings with d-dimensions and $A \in R^{|\mathbb{V}| \times |\mathbb{V}|}$ is the matrix of normalized undirected adjacency.

Graph Encoder

The neighborhood aggregation scheme on graph $G = \left(\mathcal{V}, \mathcal{E}\right)$ is of vital importance in GNN-based recommendation. Light Graph Convolution (LGC) in LightGCN is a common operation in graph contrastive learning methods because of its simplicity and efficiency. It can be defined as Equation 1:

$$e_u^{(l+1)} = \sum_{i \in \mathcal{N}_u} \frac{1}{\sqrt{|\mathcal{N}_u|}\sqrt{|\mathcal{N}_i|}} e_i^{(l)}, e_i^{(l+1)} = \sum_{u \in \mathcal{N}_i} \frac{1}{\sqrt{|\mathcal{N}_i|}\sqrt{|\mathcal{N}_u|}} e_u^{(l)}, \tag{1}$$

where $e_u^{(l)}$ and $e_i^{(l)}$ are respectively the enhanced embedding of user u and item i generated from l layers, e_u^0 and e_i^0 is the ID embeddings(trainable parameters), $\mathcal{N}_u$ indicates the set of items interacted by user u while $\mathcal{N}_i$ indicates the set of users interacting with item i. The term $\frac{1}{\sqrt{|\mathcal{N}_u|}\sqrt{|\mathcal{N}_i|}}$ can prevent the embeddings scale from growing with the graph convolution procedure. The final representations can be defined as Equation 2:

$$e_u = \sum_{l=0}^{L} \frac{1}{\left(L+1\right)} e_u^{(l)}, e_i = \sum_{l=0}^{L} \frac{1}{\left(L+1\right)} e_i^{(l)}, \tag{2}$$

where e_u and e_i stand for the final representations of user u and item i. The value of $e_u^\top e_i$ is adopted to forecast how likely u would choose i. The job of optimizing model parameters is typically framed in existing works as supervised learning. The BPR loss is a typical decision, which can be represented as Equation 3:

$$L_{BPR} = \sum_{(u,i,v) \in S} - log\left(\sigma\left(e_u^\top e_i - e_u^\top e_v\right)\right), \tag{3}$$

where $S = \{\left(u,i,v\right) \mid \left(u,i\right) \in \mathcal{E}, \left(u,v\right) \notin \mathcal{E}, v \in I\}$ is the training data.

Contrastive Learning with Noise-Based Augmentation

SimGCL has experimentally proved that the decisive factor of boosting recommendation performance is a more even representation distribution and proposed a new strategy for obtaining enhanced views. Specifically, for data augmentation, SimGCL directly incorporates random noises into the original representations. Following previous works (Yu et al., 2022a; Yu et al., 2022b), we focus on how to generate more suitable random noises and attempt to explain what role random noises play in the training process.

Inspired by InvPref (Wang et al., 2022), we believe that it is reasonable to understand the introduction of random noises from the perspective of invariant learning and feature selection. From this perspective, the reason why directly adding random noises to the original representations can obtain a more even representation distribution is that the values of some dimensions in random noises are close to zeros, which plays a role in feature selection. As a result, we directly set the values of certain dimensions in the random noises to zeros.

Formally, in consideration of a node v and its original representation e_v from graph encoder, the representation-level data augmentations can be carried out as Equation 4:

$$e_v^{'} = e_v + sign\left(e_v\right) \odot f\left(\Delta_v, k\right), \tag{4}$$

where the elements in the original random noise vectors $\Delta_v \sim N\left(0,1\right)$ and the range of values is from 0 to 1; $f\left(\Delta_v, k\right)$ resets the values smaller than k in the original random noise vectors to zero and makes sure that all added random noise vectors have the same length ϵ, which is small in order to prevent a large deviation of e_v. At the same time, we force the lengths of representations to remain unchanged after adding random noise vectors.

It is worth noting that the added random noises for every node representation are different. At each layer, different random noise vectors whose values on some dimensions are zeros are added to the present node embeddings. The final node representations are acquired by Equation 5:

$$E = \frac{1}{L+1}\sum_{l=1}^{L} E^{(l)}, E^{(l)} = AE^{(l-1)} + sign\left(E^{(l-1)}\right) \odot f\left(\Delta^{(l-1)}, k\right). \tag{5}$$

It should be noted that while calculating the final representations, the input embeddings $E^{(0)}$ is skipped, which can lead to a performance improvement.

By adding the random noise vectors whose values of certain dimensions are zeros to the original representations, some dimensions of values are disturbed, while others are accurate. For one node i ($i \in I$), each added random noise vector generates an augmented representation. Positive pairs are defined as the augmented views of the same node i ($e_i^{'}, e_i^{''}$), while negative pairs are defined as the augmented views of any different nodes. Formally, InfoNCE is used as the contrastive loss. The losses in classic recommendation can be defined as Equation 6:

$$\mathcal{L}_{cl}^{user} = \sum_{u \in U} -log\frac{exp\left(\left(\mathrm{e}_u^{'}\right)^{\top}\mathrm{e}_u^{''} / \tau\right)}{\sum_{v \in U} exp\left(\left(e_u^{'}\right)^{\top} e_v^{'} / \tau\right)}, \mathcal{L}_{cl}^{item} = \sum_{i \in I} -log\frac{exp\left(\left(\mathrm{e}_i^{'}\right)^{\top}\mathrm{e}_i^{''} / \tau\right)}{\sum_{v \in I} exp\left(\left(e_i^{'}\right)^{\top} e_v^{''} / \tau\right)}, \tag{6}$$

where $\tau > 0$ is a hyper-parameter.

Multi-Task Learning Scheme

To optimize the classic recommendation task with the self-supervised learning task and then improve the model's performance, a multi-task learning scheme is applied. It can be defined as Equation 7:

$$L = \mathcal{L}_{BPR} + \lambda_0\left(\mathcal{L}_{cl}^{user} + \mathcal{L}_{cl}^{item}\right) + \lambda_1 \left|\left|\Theta\right|\right|_2^2, \tag{7}$$

where Θ is the set of all training parameters, λ_0 and λ_1 are the hyper-parameters. Finally, Adam is used to optimize the joint loss.

EXPERIMENTS

Datasets

We use Yelp2018 dataset (He et al., 2020), Douban-Book dataset (Yu et al., 2021) and Amazon-Book dataset (Wu et al., 2021) in our experiments. Yelp dataset is widely used as a benchmark dataset in recommendation system studies. In the experiments, we chose the first version of Yelp dataset, Yelp2018. Douban-Book dataset is a public dataset containing different kinds of raw information, such as ratings, reviews, item details, and user profiles. Douban Book is a Chinese website allowing Internet users to express their opinions and comments about books. Amazon Dataset is also widely used as a benchmark dataset in recommendation system studies. A subset of Amazon dataset, Books is adopted in our experiments. To make it suitable for the Top-K recommendation, following the same strategy described in the literature (Wu et al., 2021; Yu et al., 2022b), we only retain items with a rating greater than 3 in Douban-Book, adopting a 1–5 rating system. Table 1 presents the information about datasets.

Baselines

We compare RobustGCL with the following contrastive learning models. To ensure fairness, we implement all models with SELFRec[1]and the code[2] is released.

1. **LightGCN (He et al., 2020):** It is a simple yet powerful recommendation model that just incorporates the GCN's most vital components, making it more suitable for graph recommendation. Graph contrastive learning models commonly adopt LightGCN as their backbone because of its simplicity and efficiency.
2. **SGL (Wu et al., 2021):** It supplements the traditional supervised task with self-discrimination as an additional auxiliary self-supervised task. This method enhances node representations and then improves the recommendation accuracy.
3. **SimGCL (Yu et al., 2022b):** It uses perturbations in the embedding space to obtain the enhanced views of nodes, achieving significant performance improvement and being less time-consuming.

Parameter Settings

We generate the initial embeddings with the Xavier initialization. The embedding size is 64, the L_2 regularization coefficient is 0.0001, and the batch is 2048, which are common in many papers (He et al., 2020; Wu et al., 2021; Yu et al., 2022b). The value of A is set to 0.2 for contrastive learning and the ϵ in SimGCL and RobustGCL is set to 0.1. Adam optimizes all the models with the learning rate of 0.001.

Evaluating Indicators

To assess the performance of models, Recall@20 and NDCG@20 are used. Following previous works, the full-ranking strategy (Zhao et al., 2020) is adopted. Recall refers to the proportion of successfully recommended items by the recommended algorithm among all true related items. It can be defined as Equation 8:

Table 1. Statistical information about datasets

Attribute	#Users	#Items	#Interactions	Density
Yelp2018	31,668	38,048	1,561,406	0.00130
Douban-Book	13,024	22,347	792,062	0.00272
Amazon-Book	52,643	91,599	2,984,108	0.00062

$$Recall = \frac{Correctly\ recommended\ items}{Total\ relevant\ items} \tag{8}$$

NDCG is used to measure the similarity between the ranking of recommendation results and the actual ranking. It can be defined as Equation 9:

$$NDCG = \frac{DCG}{IDCG} \tag{9}$$

where DCG Represents the cumulative gain of loss, IDCG Represents the cumulative gain of loss under ideal conditions.

Comparing Performance

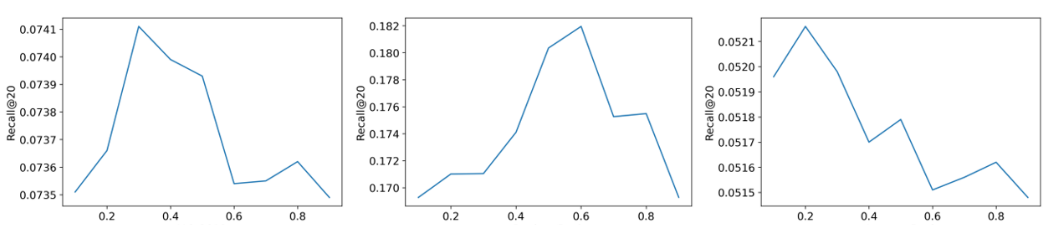
Comparing Performance Overall

Table 2 presents the comparative results with competing methods. To enhance clarity, optimal outcomes are highlighted in **bold**, while the runner-up is *italicized*. It is evident that RobustGCL consistently outperforms other baseline methods across all three datasets. In the context of random noise vectors, varying proportions of zeros yield different outcomes. The relevant experimental results are depicted in Figure 2. Notably, an optimal value of 0.3 is observed for Yelp2018, 0.2 for Amazon-Book, and 0.6 for Douban-Book. This suggests that the information contributing to the final node

Table 2. Overall comparison

Dataset	Douban-Book		Yelp2018		Amazon-Book	
Model	**Recall**	**NDCG**	**Recall**	**NDCG**	**Recall**	**NDCG**
LightGCN	0.1501	0.1282	0.0639	0.0525	0.0410	0.0318
SGL-ND	0.1626	0.1450	0.0644	0.0528	0.0440	0.0346
SGL-ED	0.1732	0.1551	0.0675	0.0555	0.0478	0.0379
SGL-RW	0.1730	0.1546	0.0667	0.0547	0.0457	0.0356
SGL-WA	0.1705	0.1525	0.0671	0.0550	0.0466	0.0373
SimGCL	0.1772	0.1583	0.0721	0.0601	0.0515	0.0414
XSimGCL	*0.0179*	*0.1591*	*0.0723*	*0.0604*	*0.0518*	*0.0416*
RobustGCL	**0.1820**	**0.1612**	**0.0740**	**0.0610**	**0.0521**	**0.0419**

Figure 2. Performance comparison with different k



representations in Douban-Book primarily stems from the node itself rather than the surrounding nodes. Conversely, the primary sources for generating final node representations in Yelp2018 and Amazon-Book are the relative nodes. Additionally, the running time is illustrated in Figure 3, with the experiment conducted on a GeForce RTX 2080Ti GPU. Figure 3 indicates that the running time of RobustGCL is comparable to that of SimGCL.

RobustGCL With Different Type of Noises

In order to further substantiate the hypothesis that random noises contribute to the model's ability to discern dimensions for learning invariant and variant features, we conducted a dedicated experiment. Initially, we standardized all dimensions in the random noises to ones, subsequently adjusting some dimensions to zeros in proportion. The experimental findings from Yelp2018 and Douban-Book are presented in Table 3, where numerical values denote the proportion of zeros in random noises. For instance, RobustGCL-f-40% signifies that 40% of zeros and 60% of ones constitute the added random noises. The results of the experiment reveal that this form of random noise effectively yields positive outcomes, indicating that the introduced noises within the embedding space indeed contribute to the recognition of both invariant and variant features.

Figure 3. Running time

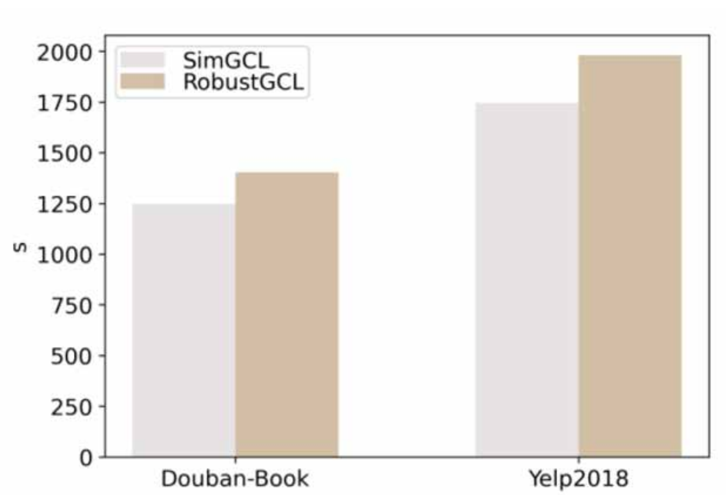


Table 3. Experiment of RobustGCL in invariant feature learning

Dataset	Yelp2018		Douban-Book	
Model	**Recall**	**NDCG**	**Recall**	**NDCG**
SimGCL	0.0721	0.0601	0.1772	0.1583
RobustGCL-f-40%	0.0738	0.0607	0.1804	0.1594
RobustGCL-f-50%	0.0735	0.0609	0.1805	0.1596
RobustGCL-f-60%	0.0737	0.0610	0.1809	0.1606
RobustGCL-f-70%	**0.0741**	**0.0610**	0.1813	0.1605
RobustGCL-f-80%	0.0738	0.0608	**0.1817**	**0.1609**
RobustGCL-f-90%	0.0735	0.0606	0.1810	0.1597

Additionally, we assessed the impact of various types of noise on RobustGCL, encompassing positive uniform noises, Gaussian noises. In contrast to SimGCL, these added noises proportionally render some dimensions as zeros. The outcomes are detailed in Table 4, where RobustGCL-u represents RobustGCL with uniform noises sampled from $U(0,1)$, RobustGCL-p denotes RobustGCL with positive uniform noises devoid of the sign of learned embeddings, and RobustGCL-g signifies RobustGCL with standard Gaussian noises. Observing the experimental results, it is evident that irrespective of the noise type, performance improvement is attainable as long as the proportion of zeros is appropriately managed.

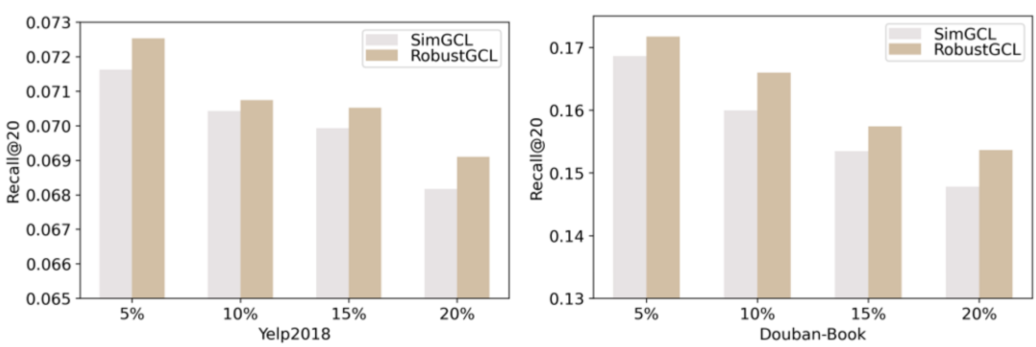
Robustness to Noisy Interactions

Experiments were conducted to assess the robustness of RobustGCL in the presence of noisy interactions. For this purpose, the training set was deliberately contaminated by introducing varying proportions of negative user-item interactions, while ensuring that the test set remained unaffected. The outcomes are illustrated in Figure 4. It is apparent that the inclusion of noisy interactions has a detrimental effect on the models' performance. However, even under these conditions, RobustGCL consistently outperforms SimGCL in terms of overall performance across different proportions of noisy data. This indicates that, through feature selection, RobustGCL can effectively discern invariant and variant representations, originating from environmental information and the node itself, respectively. Consequently, RobustGCL exhibits a heightened capacity to mitigate the impact of noisy data, as it can rely more heavily on invariant features for predictive modeling.

Table 4. Comparing the performance of RobustGCL with different types of noises

Dataset	Yelp2018		Douban-Book	
Model	**Recall**	**NDCG**	**Recall**	**NDCG**
SimGCL	0.0721	0.0601	0.1772	0.1583
RobustGCL-u	0.0735	0.0606	0.1818	0.1610
RobustGCL-p	0.0738	0.0608	0.1816	0.1603
RobustGCL-f	**0.0741**	**0.0610**	0.1817	0.1609
RobustGCL-g	0.0740	**0.0610**	**0.1820**	**0.1612**

Figure 4. Model performance in relation to the noise ratio



Long-Tail Recommendation

To assess whether RobustGCL enhances its ability to mitigate popularity bias, three subsets are derived from the test set based on the popularity of each item, with the constraint that the training set remains unchanged. Eighty percent of the items with the fewest interactions are labeled as "unpopular," five percent of the items with the highest interactions are labeled as "popular," and the remaining items are labeled as "normal." Experiments are conducted to validate the Recall@20 value contributed by each subgroup (the sum of values from the three subsets represents the overall Recall@20 value). Figure 5 presents the results, clearly indicating that RobustGCL outperforms LightGCN in recommending long-tail items. In the context of recommending popular items, RobustGCL shows a marginal improvement compared to LightGCN. However, when it comes to recommending normal and unpopular items, RobustGCL exhibits a significant enhancement over LightGCN. In the case of Yelp, RobustGCL demonstrates a 40% improvement in normal items and a 50% improvement in unpopular items. Similarly, in platforms like Douban-Book, RobustGCL showcases a 60% improvement in normal items and a 50% improvement in unpopular items.

The bar stands for Recall@20, while the percentage of performance improvement is indicated by the line.

To gain deeper insights into the benefits brought by RobustGCL and to validate the proposed perspective, Figure 6 presents the learned item embeddings, illustrating the impact of the proposed method on representation learning. For comparison, we randomly selected 32 unpopular items and 32 popular items. The upper part provides a visualization of the representation of popular items, while the lower part does the same for unpopular items. To facilitate observation, we averaged the values of adjacent dimensions.

Figure 5. Comparing performance across various item groups

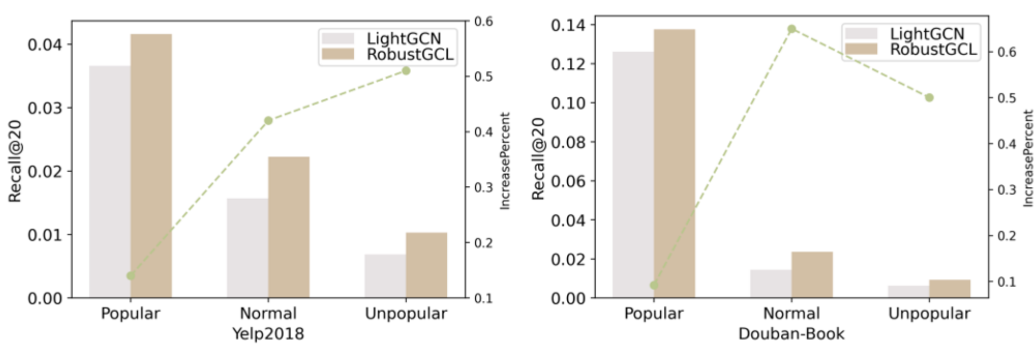
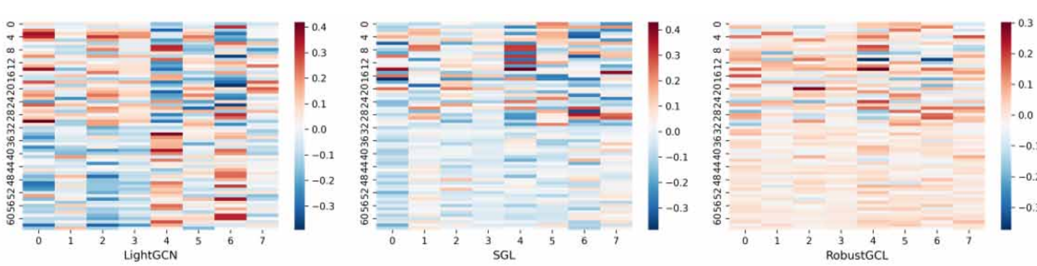


Figure 6. Visualization of item embeddings from Yelp2018



It is evident that the embeddings from LightGCN exhibit distinct color segmentation. Upon further comparing the color segmentation areas of popular and unpopular item embeddings in LightGCN, regions that are prominently red in popular item embeddings appear blue in unpopular item embeddings, and vice versa. Observing and comparing the embeddings of unpopular items learned by LightGCN, SGL, and RobustGCL, respectively, it is notable that the color blocks gradually become less pronounced in RobustGCL, indicating a more uniform distribution of embedding values.

Visualization of Embeddings

We hypothesize that popular and unpopular products may exhibit significant differences in certain features, such as sales volume, user reviews, etc. The various dimensions of the neural network can independently learn these distinctive features, leading to improved discrimination between popular and unpopular products. For instance, one dimension may specialize in learning sales volume, while another may focus on user reviews. Our methodology holds the potential to acquire representations that emphasize uniformity, allowing the model to allocate less attention to specific noisy features and concentrate more on the general characteristics of the data. This has the effect of enhancing the model's robustness and stability when confronted with minor variations in the input data.

Prospect

RobustGCL is presently trained using offline data. Online recommendation is a crucial area of focus, and RobustGCL aims to incorporate processing modules for streaming data. Additionally, our model has the potential to be extended to diverse recommendation scenarios, including, but not limited to, multi-model recommendation and conversation recommendation.

CONCLUSION

This paper provides a hypothesis on the role of added noises in the embedding space from the perspective of invariant learning and feature selection. Then the authors design several experiments to prove the validity of the hypothesis. Based on this hypothesis, the authors devise a more reasonable method for generating random noises and put forward a novel graph contrastive learning recommendation model named RobustGCL. RobustGCL randomly zeros the values of certain dimensions in the noise vectors at a fixed ratio. In this way, RobustGCL can identify invariant and variant features and then learn invariant and variant representations. Tests on publicly available datasets show that our approach can learn invariant representations and achieve better performance.

CONFLICT OF INTEREST

The authors declare no conflicts of interest.

FUNDING STATEMENT

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

AUTHOR NOTE

The data used to support the findings of this study are included within the article.

Correspondence concerning this article should be addressed to Wengui Dai, College of Computer and Information Science, Southwest University, Chongqing, Chongqing, 400715, China. Email: dwengui2020@163.com

REFERENCES

Alakbarov, R. (2022). An optimization model for task scheduling in mobile cloud computing. [IJCAC]. *International Journal of Cloud Applications and Computing*, *12*(1), 1–17. doi:10.4018/IJCAC.297102

Appati, J. K., Nartey, P. K., Yaokumah, W., & Abdulai, J. D. (2022). A systematic review of fingerprint recognition system development. [IJSSCI]. *International Journal of Software Science and Computational Intelligence*, *14*(1), 1–17. doi:10.4018/IJSSCI.300358

Bayer, I., He, X., Kanagal, B., & Rendle, S. (2017). A generic coordinate descent framework for learning from implicit feedback. *In: Proceedings of the 26th International Conference on World Wide Web*, (pp. 1341–1350). ACM. doi:10.1145/3038912.3052694

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. *In: International conference on machine learning*, (pp. 1597–1607). PMLR.

Clauset, A., Shalizi, C. R., & Newman, M. E. (2009). Power-law distributions in empirical data. *SIAM Review*, *51*(4), 661–703. doi:10.1137/070710111

Deveci, M., Gokasar, I., Pamucar, D., Zaidan, A. A., Wen, X., & Gupta, B. B. (2023). Evaluation of cooperative intelligent transportation system scenarios for resilience in transportation using type-2 neutrosophic fuzzy vikor. *Transportation Research Part A, Policy and Practice*, *172*, 103666. doi:10.1016/j.tra.2023.103666

George, G., & Lal, A. M. (2021). A personalized approach to course recommendation in higher education. [IJSWIS]. *International Journal on Semantic Web and Information Systems*, *17*(2), 100–114. doi:10.4018/IJSWIS.2021040106

Gupta, B. B., Gaurav, A., Panigrahi, P. K., & Arya, V. (2023). Analysis of artificial intelligence-based technologies and approaches on sustainable entrepreneurship. *Technological Forecasting and Social Change*, *186*, 122152. doi:10.1016/j.techfore.2022.122152

Hamilton, W., Ying, Z., & Leskovec, J. (2017). Inductive representation learning on large graphs. *Advances in Neural Information Processing Systems*, •••, 30.

He, R., & McAuley, J. (2016). Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. *In: Proceedings of the 25th international conference on world wide web*, (pp. 507–517). IEEE. doi:10.1145/2872427.2883037

He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., & Wang, M. (2020). 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*, (pp. 639–648). IEEE. doi:10.1145/3397271.3401063

Hu, B., Gaurav, A., Choi, C., & Almomani, A. (2022). Evaluation and comparative analysis of semantic web-based strategies for enhancing educational system development. [IJSWIS]. *International Journal on Semantic Web and Information Systems*, *18*(1), 1–14. doi:10.4018/IJSWIS.302895

Li, P., Li, T., Wang, X., Zhang, S., Jiang, Y., & Tang, Y. (2022). Scholar recommendation based on high-order propagation of knowledge graphs. [IJSWIS]. *International Journal on Semantic Web and Information Systems*, *18*(1), 1–19. doi:10.4018/IJSWIS.313190

Lin, Z., Tian, C., Hou, Y., & Zhao, W. X. (2022). Improving graph collaborative filtering with neighborhood-enriched contrastive learning. *In: Proceedings of the ACM Web Conference 2022*, (pp. 2320-2329). ACM. doi:10.1145/3485447.3512104

Liu, R. W., Guo, Y., Lu, Y., Chui, K. T., & Gupta, B. B. (2022). Deep network-enabled haze visibility enhancement for visual iot-driven intelligent transportation systems. *IEEE Transactions on Industrial Informatics*, *19*(2), 1581–1591. doi:10.1109/TII.2022.3170594

Milojević, S. (2010). Power law distributions in information science: Making the case for logarithmic binning. *Journal of the American Society for Information Science and Technology*, *61*(12), 2417–2425. doi:10.1002/asi.21426

Mohammed, S. S., Menaouer, B., Zohra, A. F. F., & Nada, M. (2022). Sentiment analysis of COVID-19 tweets using adaptive neuro-fuzzy inference system models. [IJSSCI]. *International Journal of Software Science and Computational Intelligence*, *14*(1), 1–20. doi:10.4018/IJSSCI.300361

Rajput, R. K. S., Goyal, D., Pant, A., Sharma, G., Arya, V., & Rafsanjani, M. K. (2022). Cloud data centre energy utilization estimation: Simulation and modelling with idr. [IJCAC]. *International Journal of Cloud Applications and Computing*, *12*(1), 1–16. doi:10.4018/IJCAC.311035

Roy, H., Islam, M. S., Haque, S., & Riyad, M. (2022). Electronic waste management scenario in Bangladesh: Policies, recommendations, and case study at Dhaka and Chittagong for a sustainable solution. *Sustainable Technology and Entrepreneurship*, *1*(3), 100025. doi:10.1016/j.stae.2022.100025

Tripathi, G., Kumar, R. (2022). A heuristic-based task scheduling policy for qos improvement in cloud. *International Journal of Cloud Applications and Computing (IJ-CAC), 12*(1), 1–22.

Wang, W., Feng, F., He, X., Nie, L., & Chua, T. S. (2021). Denoising implicit feedback for recommendation. *In: Proceedings of the 14th ACM international conference on web search and data mining*, (pp. 373–381). ACM. doi:10.1145/3437963.3441800

Wang, X., He, X., Wang, M., Feng, F., & Chua, T. S. (2019). Neural graph collaborative filtering. *In: Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*, (pp. 165–174). ACM.

Wang, Z., He, Y., Liu, J., Zou, W., Yu, P. S., & Cui, P. (2022). Invariant preference learning for general debiasing in recommendation. *In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, (pp. 1969–1978). ACM. doi:10.1145/3534678.3539439

Wu, J., Wang, X., Feng, F., He, X., Chen, L., Lian, J., & Xie, X. (2021). Self-supervised graph learning for recommendation. *In: Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, (pp. 726–735). ACM.

Xiao, J., Liu, X., Zeng, J., Cao, Y., & Feng, Z. (2022). Recommendation of healthcare services based on an embedded user profile model. [IJSWIS]. *International Journal on Semantic Web and Information Systems*, *18*(1), 1–21. doi:10.4018/IJSWIS.313198

Xu, Z., He, D., Vijayakumar, P., Gupta, B., & Shen, J. (2021). Certificateless public auditing scheme with data privacy and dynamics in group user model of cloud-assisted medical wsns. *IEEE Journal of Biomedical and Health Informatics*. PMID:34788225

Ying, R., He, R., Chen, K., Eksombatchai, P., Hamilton, W. L., & Leskovec, J. (2018). Graph convolutional neural networks for web-scale recommender systems. *In: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 974–983. doi:10.1145/3219819.3219890

Yu, J., Yin, H., Gao, M., Xia, X., Zhang, X., & Viet Hung, N. Q. (2021). Socially-aware self-supervised tri-training for recommendation. *In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, (pp. 2084–2092). ACM.

Yu, J., Yin, H., Xia, X., Chen, T., Cui, L., & Nguyen, Q. V. H. (2022b). 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*, (pp. 1294–1303). ACM.

Zhang, A., Zheng, J., Wang, X., Yuan, Y., & Chua, T. S. (2023). Invariant collaborative filtering to popularity distribution shift. *In: Proceedings of the ACM Web Conference 2023*, (pp. 1240–1251). ACM. doi:10.1145/3543507.3583461

Zhang, Q., Guo, Z., Zhu, Y., Vijayakumar, P., Castiglione, A., & Gupta, B. B. (2023). A deep learning-based fast fake news detection model for cyber-physical social services. *Pattern Recognition Letters*, *168*, 31–38. doi:10.1016/j.patrec.2023.02.026

Zhao, W. X., Chen, J., Wang, P., Gu, Q., & Wen, J. R. (2020). Revisiting alternative experimental settings for evaluating top-n item recommendation algorithms. *In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, (pp. 2329–2332). ACM. doi:10.1145/3340531.3412095

Zhou, K., Wang, H., Zhao, W. X., Zhu, Y., Wang, S., Zhang, F., Wang, Z., & Wen, J. R. (2020). S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization. *In: Proceedings of the 29th ACM international conference on information & knowledge management*, (pp. 1893–1902). ACM.

Zhou, Z., Li, Y., Li, J., Yu, K., Kou, G., Wang, M., & Gupta, B. B. (2022). Gan-Siamese network for cross-domain vehicle re-identification in intelligent transport systems. *IEEE Transactions on Network Science and Engineering*.

ENDNOTES

[1] https://github.com/Coder-Yu/SELFRec
[2] https://github.com/xinchen2/RobustGCL

Wengui Dai, postgraduate, graduated from Southwest University in 2024. Worked in college of computer and information science. His research interests include sequential recommendation, graph recommendation, natural language processing.

Yujun Wang, Ph.D., graduated from Shinshu University, Japan in 2003. He is currently an associate professor at the college of computer and information science, Southwest University. He has been engaged in the long-term research and development of mechatronics integrated products both domestically and internationally.