Self-Supervised Denoising through Independent Cascade Graph Augmentation for Robust Social Recommendation

Youchen Sun
Nanyang Technological University
Singapore
YOUCHEN001@e.ntu.edu.sg

Zhu Sun*
A*STAR Centre for Frontier AI Research; Singapore University of Technology and Design
Singapore
sunzhuntu@gmail.com

Yingpeng Du
Nanyang Technological University
Singapore
dyp1993@pku.edu.cn

Jie Zhang
Nanyang Technological University
Singapore
zhangj@ntu.edu.sg

Yew Soon Ong
A*STAR Centre for Frontier AI Research; Nanyang Technological University
Singapore
asysong@ntu.edu.sg

ABSTRACT

Social Recommendation (SR) typically exploits neighborhood influence in the social network to enhance user preference modeling. However, users' intricate social behaviors may introduce noisy social connections for user modeling and harm the models' robustness. Existing solutions to alleviate social noise either filter out the noisy connections or generate new potential social connections. Due to the absence of labels, the former approaches may retain uncertain connections for user preference modeling while the latter methods may introduce additional social noise. Through data analysis, we discover that (1) social noise likely comes from the connected users with low preference similarity; and (2) Opinion Leaders (OLs) play a pivotal role in influence dissemination, surpassing high-similarity neighbors, regardless of their preference similarity with trusting peers. Guided by these observations, we propose a novel Self-Supervised Denoising approach through Independent Cascade Graph Augmentation, for more robust SR. Specifically, we employ the independent cascade diffusion model to generate an augmented graph view, which traverses the social graph and activates the edges in sequence to simulate the cascading influence spread. To steer the augmentation towards a denoised social graph, we (1) introduce a hierarchical contrastive loss to prioritize the activation of OLs first, followed by high-similarity neighbors, while weakening the low-similarity neighbors; and (2) integrate an information bottleneck based contrastive loss, aiming to minimize mutual information between original and augmented graphs yet preserve sufficient information for improved SR. Experiments conducted on two public datasets demonstrate that our model outperforms the state-of-the-art while also exhibiting higher robustness to different extents of social noise.

CCS CONCEPTS

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

KEYWORDS

Social Recommender System; Graph Denoising; Self-Supervised Learning; Independent Cascade

ACM Reference Format:
Youchen Sun, Zhu Sun, Yingpeng Du, Jie Zhang, and Yew Soon Ong. 2024. Self-Supervised Denoising through Independent Cascade Graph Augmentation for Robust Social Recommendation. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24), August 25–29, 2024, Barcelona, Spain.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3637528.3671958

1 INTRODUCTION

Social Recommendation (SR) predicts the user's preference by leveraging both the user-item interaction network and the social network. The underlying principle of SR is known as the homophily theory [20], that is, individuals are more inclined to be influenced by those people they know or trust. However, there may exist noisy connections in social networks due to the intricate nature of users' social behavior and the low cost of making connections. For instance, users may establish connections with others for diverse reasons or even connect to others by accident. As shown in Figure 1, the preference similarity between many connected users in the social network is very low, with a majority of them lower than 0.4. Hence, these connections are likely noise and should be cautiously utilized for user preference modeling.

However, most SR approaches [4, 29, 44, 50, 52], which are mainly built upon the powerful Graph Neural Networks (GNNs) to capture the interconnections among nodes, indiscriminately employ all social connections including the noisy ones. This inevitably leads to

*Corresponding Author

KDD '24, August 25–29, 2024, Barcelona, Spain

ACM ISBN 979-8-4007-0490-1/24/08
https://doi.org/10.1145/3637528.3671958

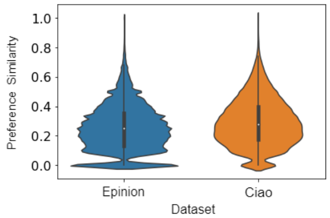

Figure 1: Distribution of preference similarity between connected users. Refer to Section 2 for detailed calculation.

sub-optimal user preference inference, consequently undermining the model's robustness. Although there are approaches designed to eliminate social noise for robust SR, they typically either filter out noisy social connections [24, 40, 41] or generate new potential connections [49–51], while unobserved, are likely to exert social influence. Due to the absence of ground truth labels, both methods exploit supervision signals from the original noisy social networks. As a result, the relevance of the preserved connections to user preference modeling remains uncertain, and the inclusion of new connections may introduce further noise.

Recently, Self Supervised Learning (SSL) [3], a paradigm that learns from unlabeled data, has achieved great success in many fields. Graph Contrastive Learning (GCL) is one of the promising directions that integrates SSL with graph representation learning [18, 47] to derive informative and robust representations for downstream tasks. The main idea is to conduct contrastive tasks among representations derived from distinct graph views generated through augmentation techniques such as edge dropout. Inspired by this, several works have integrated GCL [2, 50, 53] into the GNN-based SR, demonstrating its efficacy in addressing the network sparsity and enriching the social semantics being modeled. However, none of them explicitly considers the social noise, leaving GCL's potential to handle noisy data and enhance the robustness of SR under-explored. Although previous studies [42, 45] show that GCL with edge dropout augmentation can alleviate the noise in the user-item interaction graph, they drop each edge independently which may not be effective for denoising in the social networks.

Specifically, the connections among users in the social network are more complicated, as the social influence can cascade to users who are multi-hop away. Therefore, whether a social neighbor is noisy depends not only on the node itself but also on the indirect neighbors. Moreover, some users may be more crucial in the information propagation process. In particular, there is a group of users known as Opinion Leaders (OLs), who have a stronger ability to spread information and shape others' opinions, as highlighted by previous studies [1, 13, 28, 46]. For example, the endorsement of certain products by OLs can swiftly lead to adoption or interest development among their followers. Hence, we assert that OLs can act as a catalyst for the diffusion of social influence. This is substantiated through our data analysis in Section 2, where we illustrate the distinctive traits of OLs and unveil their paramount significance, surpassing that of high-similarity neighbors, in the influence dissemination. In light of this, two main challenges arise: (1) how to effectively consider the cascading spread of social influence and (2) how to construct the proxy contrastive tasks for effective self-supervised social denoising.

Table 1: Performance of SR under different social setups (w.r.t NDCG@10). We conduct paired t-tests at the user level between each of the last four setups with "w social", which are all significant at 1% significance level.

Dateset	Epinion		Ciao	
Model	TrustSVD	DiffNet++	TrustSVD	DiffNet++
w social	0.0191	0.0198	0.0249	0.0221
w/o social	0.0202	0.0211	0.0244	0.0214
w/o low sim	0.0205	0.0209	0.0252	0.0227
w/o high sim	0.0185	0.0195	0.0247	0.0217
w/o OLs	0.0183	0.0190	0.0244	0.0211

To tackle these challenges, we propose a novel Self-Supervised Denoising approach through Independent Cascade Graph Augmentation (SSD-ICGA) for robust SR, guided by our observations. Specifically, we adopt the Independent Cascade (IC) diffusion model [14] to simulate the cascading influence diffusion in the social network. IC traverses the original graph, activates the edges in cascade based on learned probability, and eventually generates augmented graph views with inactive edges dropped. To further effectively guide the IC process toward a denoised social graph (i.e., the inactive edges are indeed the noisy ones), we introduce i) the hierarchical contrastive task, which prioritizes the activation by OLs neighbors, followed by the neighbors with high preference similarity, while deprioritizing the low-similarity neighbors; and ii) the information bottleneck based contrastive task, which minimizes the mutual information between the representation derived from the original and augmented graphs while ensuring the augmented graph retains adequate information for an enhanced SR. The main contributions of this paper lie three-fold:

- We perform data analysis that empirically uncovers (1) the existence of social noise and (2) the hierarchy of social influence on user preference modeling among neighbors: OLs > high-similarity neighbors > low-similarity neighbors.
- We introduce an independent cascade-based graph augmentation that simulates cascading influence spread in the social network, seamlessly amplifying the impact of OLs and high-similarity neighbors while weakening that of low-similarity neighbors through a hierarchical contrastive task. Such augmentation is further integrated with an information bottleneck-based contrastive task for effective self-supervision towards a denoised social graph, thereby facilitating robust SR.
- We conduct experiments on two real-world datasets to validate the superiority of our proposed approach in terms of its prediction accuracy and robustness to social noise.

2 DATA ANALYSIS

We first conduct an exploratory analysis on two widely used datasets for SR, namely Epinion and Ciao[1], to answer the following two questions that motivate our study: (1) Do social networks contain noisy connections that adversely affect user modeling? and if so, what are the potential sources? (2) Are there any social connections that have a more significant positive impact on user modeling?

Observation 1: *Social networks contain noise, which likely originates from connections of users with low preference similarity.*

[1]The details of the datasets are introduced in Section 4.

Table 2: Statistics of Opinion Leaders. We present the mean values for each entry and perform two-sided independent t-tests to compare the two groups across all metrics, which all indicate significant difference at 1% significance level.

Dataset	**Epinion**		**Ciao**	
Social Role	OL	Ord	OL	Ord
# Consumed Item	78.9	43.2	94.1	41.4
Consumed Item Popularity	37.3	53.1	21.2	41.9
Percentage of Common Item with Truster	31.8%	4.9%	15.3%	6.4%
Social In-degree	200.7	18.9	62.8	17.6
Social Out-degree	120.1	22.7	55.7	16.3
Clustering Coefficient	0.21	0.13	0.15	0.11

In reality, we may tend to take advice from social neighbors who have common interests with us. In other words, some neighbors in the social networks might not have a social influence on the user's preference. Figure 1 illustrates the distribution of preference similarity between each pair of users in the social network. Due to the high sparsity of user-item interactions, we represent the preference of user u as a vector $[c_1, c_2, ..., c_g, ...]$, where c_g is a binary value indicating whether u has consumed any items from category g. Then, for each user pair, we calculate the Jaccard similarity between their preference vectors to measure their preference similarity. As shown in the plot, the similarity between most social connections falls below 0.4, with numerous connections exhibiting relatively low similarity, especially in Epinion. These neighbors probably exert minor social influence on the user's preferences or even be noise, considering the dissimilarity in their preferences.

To validate it, we train two representative social recommenders (matrix factorization based TrustSVD [6] and deep learning based DiffNet++ [44]) under different social setups: all social connections included ("w social"), all social connections excluded ("w/o social"), high-similarity ("w/o high sim") or low-similarity ("w/o low sim") neighbors excluded, respectively. Neighbors with preference similarity exceeding the median similarity of all neighbors are high-similarity neighbors and those falling below are low-similarity neighbors. The results are shown in Table 1. Firstly, we note that incorporating social neighbors improves the prediction performance on Ciao but harms the performance on Epinion. This could be attributed to the denser distribution of lower preference similarity (around 0) in Epinion. Secondly, the performance of "w/o low sim" surpasses that of "w social", suggesting eliminating connections with lower similarity improves performance. Thirdly, "w/o high sim" achieves worse performance than "w social", showcasing that the removal of high-similarity counterparts leads to performance decline. In summary, the above observations help verify the existence of noise in the social network, which likely originates from low-similarity social connections.

Observation 2: *Among all neighbors, Opinion Leaders (OLs) exert a stronger positive influence on user preference modeling than ordinary neighbors, regardless of their preference similarity with the target user. Accordingly, we identify a hierarchy of influence among neighbors: OLs > high-similarity neighbors > low-similarity neighbors.*

In the information diffusion process, users do not contribute equally, with some playing more crucial roles than others. For example, users may be particularly influenced by some individuals, such as OLs

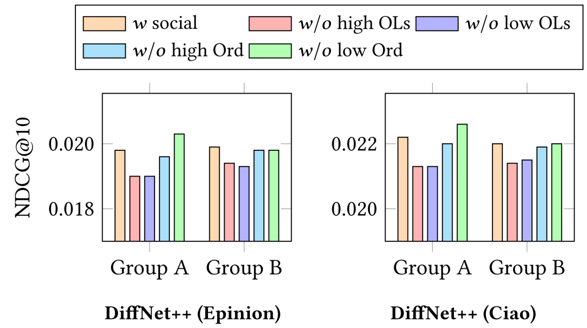

Figure 2: Comparison of the importance of OLs, high-similarity neighbors, and low-similarity neighbors.

who possess expertise and thus have a stronger ability to shape others' opinions. As highlighted by previous work [46], OLs taking up 1% of the total users are sufficient to spread information across the network. Following [46], we use PageRank [23] as an indicator of the user's importance in the social network and choose the users with the top 1% PageRank as the OLs[2].

We then explore the distinctive traits of OLs by comparing them with ordinary users ("Ord") across several metrics relevant to the recommendation context, as shown in Table 2, where several findings can be derived. Firstly, OLs consume more items that are of lower popularity on average. This is probably due to their expertise in certain fields and thereby having their preference for a wider range of products rather than following the majority. Moreover, they are trusted by more individuals than those whom they trust and their consumed items are more likely to be adopted by their neighbors, indicating that they are more trustworthy and have a stronger ability to influence their followers. Additionally, OLs' social in-degree and out-degree are much larger and their social neighbors are more closely connected (higher clustering coefficient), which facilitates a broader spread of influence. In summary, OLs are crucial in disseminating information and shaping others' opinions. Thus, considering the role of OLs would enhance user preference inference and the robustness of SR models. To support this, we train the two representative social recommenders with OL connections removed from the social network. The results are shown in the last row in Table 1, which implies a more significant negative impact than removing the high-similarity social neighbors.

Inspired by the observations that the high-similarity neighbors have stronger positive effects on user modeling than low-similarity neighbors, we further investigate whether this phenomenon extends to the OL connections. Specifically, we choose the group of users (Group A) who are connected to at least one OL and respectively remove each type of their connected neighbors: high-similarity OLs ("w/o high OLs"), low-similarity OLs ("w/o low OLs"), high-similarity ordinary neighbors ("w/o high Ord"), and low-similarity ordinary neighbors ("w/o low Ord"). We also look at the indirect impact on the rest users (Group B), whose neighbors are unaffected by the changes. As depicted in Figure 2[3], OLs exhibit consistent significance irrespective of their preference similarity to

[2]We confine the influence of OLs to users who are directly connected or reachable through multi-hop connections.

[3]Due to space limitation, we put the results of TrustSVD in Appendix A, and a similar trend can be observed as DiffNet++.

the target user, surpassing the impact of high-similarity ordinary users, followed by low-similarity ordinary users. In other words, there exists a hierarchy of influence: OL neighbor > high-similarity neighbor > low-similarity neighbor. Moreover, removing OL neighbors of users in Group A results in a performance decline in Group B whereas the removal of ordinary users has a minor effect, indicating the impact of OLs on the broader context.

3 THE PROPOSED METHODOLOGY

Overview of SSD-ICGA. We propose SSD-ICGA, a novel social recommender that achieves Self-Supervised Denoising through Independent Cascade Graph Augmentation. The overall structure is depicted in Figure 3. Technically, we first propose graph augmentation through the Independent Cascade (IC) diffusion model to consider the cascade spread of social influence. Then we introduce the hierarchical contrastive loss to amplify the role of OLs and high-similarity neighbors while de-emphasizing the low-similarity neighbors in the IC diffusion process. Finally, the graph augmentation is further integrated with an information bottleneck based contrastive task to seek efficient knowledge extraction by striking a balance between preserving relevant information and abandoning non-essential details for robust augmentation.

Problem Definition. Let $\mathcal{U} = \{u_1, u_2, ..., u_M\}$ denote the set of users and $\mathcal{I} = \{i_1, i_2, ..., i_N\}$ denote the set of items, where M is the number of users and N is the number of items. We use $\mathcal{K}$ to denote the set of OLs. Two graphs are involved: the user-item graph $\mathcal{G}_R$ and the directed social graph $\mathcal{G}_S$. The corresponding binary adjacency matrices are denoted as $\mathbf{R} = \{r_{ui}\}_{M\times N}$ and $\mathbf{S} = \{s_{uv}\}_{M\times M}$, where $r_{ui} = 1$ if user u interacts with item i, and $s_{uv} = 1$ if user u is connected with user v. Each user and item is associated with a d-dimensional initial ID embedding $\mathbf{e}_u$ and $\mathbf{e}_i$, respectively. Given $\mathcal{G}_R$ and $\mathcal{G}_S$, the objective is to obtain denoised social graph $\tilde{\mathcal{G}}_S$ for robust prediction of unobserved user-item interactions in $\mathbf{R}$.

3.1 Preference Modeling

For the preference modeling in $\mathcal{G}_R$, we follow the state-of-the-art framework LightGCN [10], which is a simple but effective GNN without redundant transformation and activation:

$$\mathbf{h}_u^{l+1} = \sum_{i\in\mathcal{N}_u^r} \frac{1}{\sqrt{|\mathcal{N}_u^r||\mathcal{N}_i^r|}}\mathbf{h}_i^l, \quad \mathbf{h}_i^{l+1} = \sum_{u\in\mathcal{N}_i^r} \frac{1}{\sqrt{|\mathcal{N}_u^r||\mathcal{N}_i^r|}}\mathbf{h}_u^l, \quad (1)$$

where $\mathcal{N}_u^r$ and $\mathcal{N}_i^r$ are the sets of neighbors of u and i in $\mathcal{G}_R$; $\mathbf{h}_u^{l+1}$ and $\mathbf{h}_i^{l+1}$ are the embeddings of u and i at layer $l+1$; $\mathbf{h}_u^0 = \mathbf{e_u}, \mathbf{h}_i^0 = \mathbf{e_i}$. After stacking L_r layers, the embeddings from each layer are combined to form the final embedding in the interaction network:

$$\mathbf{h}_u = \sum\nolimits_{l=0}^{L_r} \mathbf{h}_u^l, \quad \mathbf{h}_i = \sum\nolimits_{l=0}^{L_r} \mathbf{h}_i^l. \quad (2)$$

3.2 Social Graph Augmentation through Independent Cascade

Aggregating noisy edges in input graphs can compromise representation learning of GNNs, particularly as the layer stacks deeper. Recent works [42, 45] reveal the potential of GCL in enhancing the robustness of GNN-based recommenders to the user-item interaction noise through graph augmentation, such as edge dropout. We argue that such augmentation is insufficient to identify the noisy edges in the social network since it fails to consider the cascading spread of influence in the social network, as analyzed in Section 1. Hence, we leverage the stochastic information diffusion model [14, 16], which simulates such cascades and predicts the information flow to generate the augmented graph view.

Specifically, we adopt the widely-used information diffusion model - Independent Cascade (IC) [14], where each node in the network has two states, either active or inactive. An initial seed set is activated at time $t = 0$ and starts to activate their in-going neighbors recursively at discrete time steps. In particular, at step t, each user v that is activated at step $t-1$ will activate each in-going inactive neighbor u with probability p_{uv}. After being activated, each node has one chance to activate its neighbors and stay active until the diffusion process terminates with no more activations. This stochastic process transverses the input graph sequentially and generates augmented graphs with inactive edges removed. Different from the straightforward edge dropout to drop each edge independently [42, 45], the active or inactive edges at each time will affect the activation in subsequent steps. For example, if u is not activated by its noisy neighbors, the influence of such noisy neighbors is constrained from propagating to further hops. Conversely, the activation by valuable neighbors will facilitate further activations. Hence, IC augmentation can better capture the cascading spread of social influence and achieve denoised graph augmentation. It is formulated as:

$$\tilde{\mathcal{G}}_S = IC(\mathcal{G}_S, \mathcal{A}_0, \mathbf{P}), \quad (3)$$

where $\tilde{\mathcal{G}}_S$ is the denoised social graph generated by the diffusion process; $\mathbf{P}$ is the activation probability matrix, where each element p_{uv} represents the probability of v activating u and is learned through a multiple layer perception (MLP):

$$p_{uv} = MLP((\mathbf{W}_1\mathbf{e}_v||\mathbf{W}_2\mathbf{e}_u), \quad (4)$$

where $||$ is the concatenation operation; $\mathbf{W_1}, \mathbf{W_2} \in \mathbf{R}^{d\times d}$ are the linear transformation matrix applied to the activater and activatee respectively. $\mathcal{A}_0$ is the initial active user set at $t = 0$, where we start from a set of OLs and sample 1% users based on the PageRank score to increase randomness. In the original IC, each user can only be activated by one of its neighbors. We relax it by allowing reactivation by different neighbors, which is a more realistic reflection of the evolving diffusion process. Finally, the propagation terminates if either there is no activation or the max_iter is reached. Empirically, we set $max\_iter = 5$ according to the six degrees of separation theory [5]. The overall process is described in Algorithm 1.

3.3 Contrastive Learning for Denoised Social Graph Augmentation

IC traverses the social graph, dropping inactive edges to generate the augmented graph for robust social influence modeling. However, the recommendation objective alone is insufficient to guide the IC process toward a denoised social graph. To ensure the effectiveness of the augmentation, we devise two self-supervised contrastive tasks to facilitate an optimal social graph augmentation.

3.3.1 *Hierarchical Contrastive Loss.*

Section 2 observes an influence hierarchy among neighbors in the social network: OLs > high-similarity neighbors > low-similarity neighbors. To account

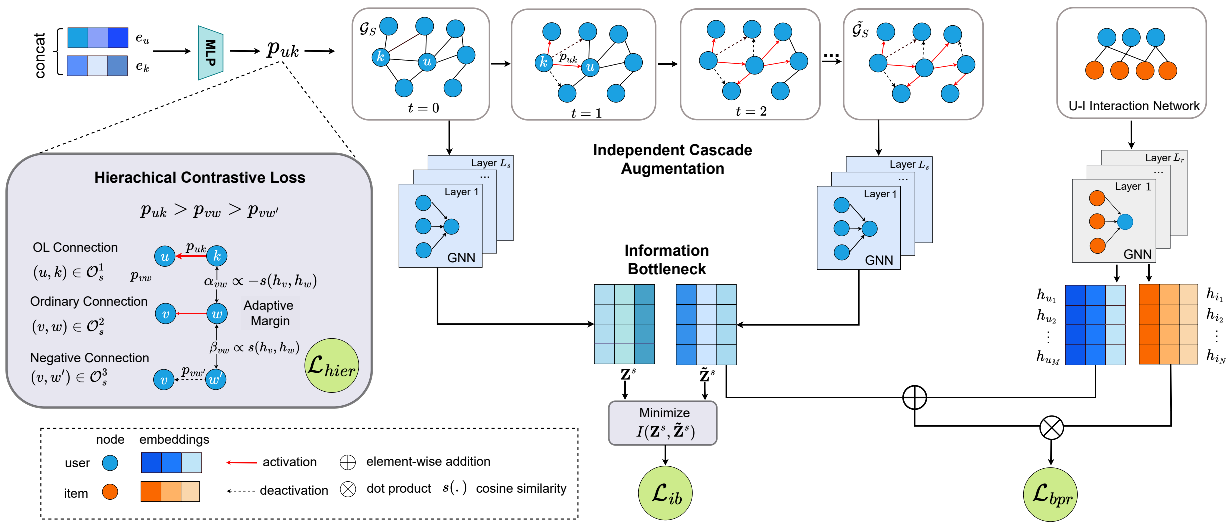

Figure 3: The overall framework of SSD-ICGA.

Algorithm 1: IC based Graph Augmentation

Data: $\mathcal{G}_S$, P, max_iter
Result: Augmented graph $\tilde{\mathcal{G}}_S$

Initialize seed set $\mathcal{A}_0$, empty graph $\tilde{\mathcal{G}}_S$, $t = 0$;
while $\mathcal{A}_t \neq \emptyset$ ***and*** $t \leq max\_iter$ **do**
  $\mathcal{A}_{t+1} = \emptyset$;
  foreach $v \in \mathcal{A}_t$ **do**
    foreach $u \in \mathcal{N}_v^s$ **do**
      Activate u at probability p_{uv};
      if *u is activated by v* **then**
        Add edge (u, v) to $\tilde{\mathcal{G}}_S$;
        $\mathcal{A}_{t+1} = \mathcal{A}_{t+1} \cup \{u\}$;
  $t = t + 1$;

for this, we impose hierarchical constraints on the activation probability within the IC process, that is, the probability of activation by OL neighbors surpasses that of high-similarity ordinary neighbors, which further exceeds that of low-similarity ordinary neighbors.

To this end, we introduce two contrastive losses [11] to enforce positive margins between the activation probability by (1) OL neighbors and ordinary neighbors; (2) ordinary neighbors and unobserved neighbors, respectively. This also indirectly prioritizes the activation by OL neighbors over the unobserved connections. The hierarchical contrastive loss is formulated as follows:

$$\mathcal{L}_{\text{hier}} = \sum\nolimits_{(u,k)\in O_1^s,(v,w)\in O_2^s} max(p_{uk} - p_{vw} + \alpha_{vw}, 0) + \sum\nolimits_{(v,w)\in O_2^s,(v,w')\in O_3^s} max(p_{vw} - p_{vw'} + \beta_{vw}, 0), \tag{5}$$

where $O_1^s = \{(u,k)|(u,k) \in \mathcal{G}_S, k \in \mathcal{K}\}$ is the set of connections to OLs; $O_2^s = \{(v,w)|(v,w) \in \mathcal{G}_S, w \notin \mathcal{K}\}$ is the set of connections to ordinary users; $O_3^s = \{(v,w')|(v,w') \notin \mathcal{G}_S, w' \notin \mathcal{K}\}$ is the set of unobserved connections. To consider the preference similarity between connected users in the hierarchy, we incorporate adaptive margin, which depends on preference similarity, into the contrastive loss. Specifically, α_{vw} is a monotone decreasing function w.r.t the preference similarity between v and w, i.e., the higher the preference similarity, the smaller the margin. In other words, if v's preference is similar to u, the probability of v activating u should be closer to that of an OL neighbor activating u. Similarly, β_{vw} is a monotone increasing function w.r.t the preference similarity, which helps ensure a larger margin with higher preference similarity. We empirically choose the tempered inverse sigmoid and tempered sigmoid as the margin function:

$$\alpha_{vw} = \frac{\eta}{1 + exp(s(\mathbf{h}_v, \mathbf{h}_w)/\theta)};\ \beta_{vw} = \frac{\eta}{1 + exp(-s(\mathbf{h}_v, \mathbf{h}_w)/\theta)}, \tag{6}$$

where $s(.)$ is the cosine similarity function; η is the scaling parameter and θ is the temperature parameter. We further apply an entropy regularization [8], a technique from reinforcement learning to promote action diversity. This regularization is applied to the activation probability to encourage more spread-out distribution, preventing it from converging to a very large or small value:

$$\mathcal{L}_{\text{ent}} = -\sum\nolimits_{(u,v)\in\mathcal{G}_S} p_{uv} \log(p_{uv}). \tag{7}$$

3.3.2 ***Information Bottleneck based Contrastive Loss.*** The hierarchical contrastive loss focuses on the relative order of the social connection activation. We further adopt a self-supervised contrastive learning scheme based on the Information Bottleneck (IB) theory to guide IC toward optimal denoised augmentation. IB [34, 35] seeks efficient knowledge extraction by striking a balance between retaining relevant information and discarding non-essential details for downstream tasks, formulated as:

$$\max\nolimits_{\mathbf{Z}} I(\mathbf{Y}, \mathbf{Z}) + (-\gamma I(\mathbf{X}, \mathbf{Z})), \tag{8}$$

where $I(,)$ is the function that measures the mutual information between two variables; X is the input data; Y is the label for the downstream task and Z is the compressed representation of X. The first term optimizes the performance of downstream tasks and the second term maximizes the amount of information removed from the original data, which are balanced by γ. This principle can be utilized to guide the augmentation by minimizing the mutual information between the original graph and the augmented graph while ensuring the retained social information in the augmented graph is sufficient for the SR task. In this way, the model can learn to drop the noisy social connections and only keep the relevant ones for the social influence modeling.

The objective of our study can be adapted as follows:

$$\min_{\Theta} \mathcal{L}_{bpr}(\mathcal{G}_R, \tilde{\mathcal{G}_S}; \Theta) + \gamma I(\mathbf{Z}^s, \tilde{\mathbf{Z}}^s), \tag{9}$$

where $\mathcal{L}_{bpr}(.)$ is the widely used BPR loss [25] for the SR task; Θ is the parameter set; $\mathbf{Z}^s$ and $\tilde{\mathbf{Z}}^s$ are the nodes representations in $\mathcal{G}_S$ and $\tilde{\mathcal{G}_S}$, learned through LightGCN alike GNN encoder:

$$\mathbf{z}_u^{l+1} = \sum\nolimits_{v \in \mathcal{N}_u^s} \frac{1}{\sqrt{|\mathcal{N}_u^s||\mathcal{N}_v^s|}} \mathbf{z}_v^l, \quad \mathbf{z}_u = \sum\nolimits_{l=0}^{L_s} \mathbf{z}_u^l \tag{10}$$

$$\tilde{\mathbf{z}}_u^{l+1} = \sum\nolimits_{v \in \mathcal{N}_u^s} \frac{p_{uv}}{\sqrt{|\mathcal{N}_u^s||\mathcal{N}_v^s|}} \tilde{\mathbf{z}}_v^l, \quad \tilde{\mathbf{z}}_u = \sum\nolimits_{l=0}^{L_s} \tilde{\mathbf{z}}_u^l, \tag{11}$$

where $\mathcal{N}_u^s$ is the set of social neighbors of u; $\mathbf{z}_u^0 = \tilde{\mathbf{z}}_u^0 = \mathbf{h}_u$; L_s is the number of stacked social layers; $\mathbf{z}_u$ and $\tilde{\mathbf{z}}_u$ are the user's final social representations in the original and augmented view, corresponding to the row vector of $\mathbf{Z}^s$ and $\tilde{\mathbf{Z}}^s$, respectively. Note that in Eq. 11, the neighborhood aggregation is based on the activation probability p_{uv}, which functions as the attention score to signify the importance of each neighbor. Following previous work, we adopt the InfoNCE [7, 22] as a measure of the mutual information and the second term in Eq. 9 is formulated as:

$$\mathcal{L}_{ib} = I(\mathbf{Z}^s, \tilde{\mathbf{Z}^s}) = \sum\nolimits_{u \in \mathcal{U}} \log \frac{exp(s(\mathbf{z}_u, \tilde{\mathbf{z}}_u)/\tau)}{\sum_{v \in \mathcal{U}} exp(s(\mathbf{z}_u, \tilde{\mathbf{z}}_v)/\tau)}, \tag{12}$$

where τ is the temperature parameter and the views of the same node form the positive pair while the views of any two different nodes form the negative pairs.

3.4 Optimization and Complexity Analysis

Opitimization. The user's representation in preference modeling and social influence modeling are merged to derive the user's final representation and we use the inner product for the prediction:

$$\bar{\mathbf{h}}_u = \mathbf{h}_u + \tilde{\mathbf{z}}_u, \quad \hat{r}_{ij} = \bar{\mathbf{h}}_u \cdot \mathbf{h}_i. \tag{13}$$

For the main SR task, the model is trained to optimize the BPR loss [25], which is formulated as:

$$\mathcal{L}_{bpr} = \sum\nolimits_{(u,i^+,i^-) \in O^r} -\ln \sigma(\hat{r}_{ui^+} - \hat{r}_{ui^-}), \tag{14}$$

where $O^r = \{(u, i^+, i^-)|(u, i^+) \in \mathcal{R}^+, (u, i^-) \in \mathcal{R}^-\}$ is the training set for item prediction; $\mathcal{R}^+$ and $\mathcal{R}^-$ are the positive and negative sample sets, respectively. Unifying the main SR loss with the two contrastive losses and the entropy regularization loss, the overall training loss is defined as:

$$\mathcal{L} = \mathcal{L}_{bpr} + \gamma \mathcal{L}_{ib} + \lambda_1 \mathcal{L}_{hier} + \lambda_2 \mathcal{L}_{ent} + \lambda_3 ||\Theta||_2^2, \tag{15}$$

where λ_1, λ_2, λ_3 control the weights of the hierarchical contrastive task, entropy regularization and L_2 regularization, respectively. The training procedure is summarized in Appendix B.

Space Complexity. Our model follows the paradigm of LightGCN to learn the node representation in both networks, which only involves the user and item embeddings of size $(M + N) \times d$. The addition parameters come from the activation probability modeling, including two transformation matrices of size $2 \times d \times d$ and the two-layer MLP of size $2 \times \frac{1}{2} \times d \times d$. Since $d \ll M, N$, our model is of comparable size to LightGCN and is lighter than most social recommenders such as DiffNet++ [44] and MHCN [52].

Table 3: Statistics of datasets.

	#Users	#Items	#Interactions	#Social Relations	Interaction Density	Social Relation Density
Epinion	22,112	79,507	517,908	355,003	0.010%	0.073%
Ciao	7,349	31,381	167,678	57,544	0.022%	0.107%

Time Complexity. The time complexity of our model mainly comes from three parts: (a) GCN aggregation; (b) hierarchical contrastive loss; and (c) IB-based contrastive loss. For (a) in both networks, the time cost is $O((L_r|\mathcal{E}_R| + L_s|\mathcal{E}_S|)d)$, where $|\mathcal{E}_R|$, $|\mathcal{E}_S|$ are the number of edges in the interaction and social networks, respectively. For (b), the computational cost for each batch is $2bd$. For (c), the calculation of InfoNCE takes $O(b(d + Md))$ in each batch, where b is batch size. The complexity can be reduced by performing contrast learning within the batch and the time cost is $O(b(d + bd))$. Overall, since $L_s, L_r, d, b \ll M, |\mathcal{E}_R|, |\mathcal{E}_S|$, the time cost scales linearly with the user and network size.

4 EXPERIMENTS AND ANALYSIS

4.1 Experimental Settings

4.1.1 ***Datasets.*** We conduct experiments on two commonly used SR datasets, Epinion and Ciao [32], both containing user-item interaction and social networks. Following state-of-the-arts [10, 39], we only retain the ratings that are greater than 3 and assign them a value of 1, and the rest are negative samples and assigned a value of 0. Moreover, we filter out users and items with less than two interactions. The detailed statistics of the two datasets are shown in Table 3. Following [10, 44], all datasets are randomly split into 8:1:1 for training, validation, and testing.

4.1.2 ***Baselines.*** We compare SSD-ICGA with 12 state-of-the-art models across three different classes, including (1) the matrix factorization based methods BPR [25], SBPR [54] and TrustSVD [6]; (2) the GNN-based methods LightGCN [10], DiffNet++ [44], MHCN [52], GoRec [37], DcRec [43], DSL [38] and DMJP [29]; and (3) the denoising SR models ESRF [49] and GDMSR [24]. We do not compare with baselines such as SoRec [19], GraphRec [4], SEPT [50] and IF-BPR [48] since previous work [24, 44, 52] have validated their superiority over them. In particular, **BPR** is the classic matrix factorization-based method that optimizes BPR loss; **SBPR** extends BPR by considering the ranking of items consumed by social neighbors; **TrustSVD** models the influence of first-order social neighbors based on SVD++; **LightGCN** learns the node embeddings through a simplified GNN; **DiffNet++** uses the attention mechanism to model the high-order influence in interaction and social networks; **MHCN** conducts hypergraph convolution on multiple motif-based hypergraphs and integrates SSL for training; **GoRec** elicits the opinions from influential nodes and integrates them into the GNN; **DcRec** conducts contrastive learning between the social and interaction networks for knowledge transfer; **DSL** leverages SSL to effectively transfer social signals into user modeling; **DMJP** learns the disentangled representation in social and interaction networks for mutual enhancement; **ESRF** unifies the social neighborhood generation and denoising in an adversarial training setting; **GDMSR** proposes to denoise the social network that can be adapted to any SR models and we implement MHCN following the original paper.

4.1.3 ***Evaluation Metrics.*** We adopt two widely used metrics, Recall@K and NDCG@K to evaluate the performance of all models. Specifically, Recall@K measures whether the target is successfully ranked in the top-K list and NDCG@K focuses on the rank of the target item in the list. Following [10, 39], we perform item ranking on all non-interacted items for each user instead of the sampled item sets to ensure an unbiased evaluation process.

4.1.4 ***Implementation Details.*** For a fair comparison, all methods are trained to optimize BPR loss with Adam [15] as the optimizer; the batch size is fixed as 1024; the learning rate is 0.001 and the latent embedding dimension is set to 64. Other essential hyperparameters are tuned using grid search based on the performance on the validation set. We train each method for 500 epochs and adopt the early stop strategy such that the training process terminates if the performance on the validation set does not improve for 50 consecutive epochs. For SSD-ICGA[4], the MLPs use the two-layer structure; γ is searched in $\{e^{-1}, e^{-2}, e^{-3}, e^{-4}\}$; λ_1 and λ_2 are searched in $\{0.01, 0.1, 0.5, 1, 5, 10\}$; λ_3 is searched in $\{e^{-1}, e^{-2}, e^{-3}, e^{-4}, e^{-5}\}$; τ is searched in $\{0.1, 0.3, 0.5, 1\}$. Due to space limitation, the best parameter settings for all methods are provided in Appendix C for reproducibility [30, 31].

4.2 Results and Analysis

4.2.1 ***Overall Performance Comparison.*** The performance of all methods on the two datasets is reported in Table 4. 'Improv%' indicates the relative improvement of SSD-ICGA (in bold) over the strongest baseline (underlined). Several major findings can be noted. **Firstly**, SSD-ICGA consistently performs the best on both datasets. This is attributed to three aspects: (a) it achieves graph augmentation through the IC model, which captures the cascading spread of social influence; (b) it amplifies the roles of OL and high-similarity neighbors while de-emphasizing the low-similarity neighbors in the information diffusion process via a hierarchical contrast task; and (c) it further integrates the information bottleneck based GCL for effective self-supervised denoising. **Secondly**, GNN-based SR models (e.g., DMJP and DSL) outperform the MF-based methods (e.g., SPR and TrustSVD) significantly, validating the power of GNNs in handling graph structure data. Moreover, SSL-enhanced SR models outperform those without SSL, which verifies the effectiveness of SSL for enhanced SR. **Lastly**, LightGCN, as a strong baseline, outperforms most SR models, possibly because most SR models directly use the social network with potential noise. It is further confirmed by the fact that GDMSR outperforms LightGCN since it essentially implements MHCN on a denoised social graph. This verifies the existence of social noise, its harm on social influence modeling, and the value of fusing denoised social networks.

4.2.2 ***Performance w.r.t Different Sparsity Level.*** To evaluate the models' performance at various sparsity levels, we rank all users in ascending order based on their number of interactions and then divide them into four groups of equal size. The performance of SSD-ICGA and the two best-performing baselines, GDMSR and DSL, under each group are reported in Figure 4, where SSD-ICGA consistently outperformed other baselines across all sparsity levels. This underscores the effectiveness of incorporating denoised social signals, even under scenarios where interaction data are limited.

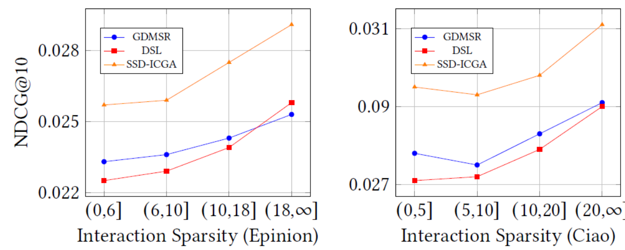

Figure 4: Performance w.r.t different sparsity level.

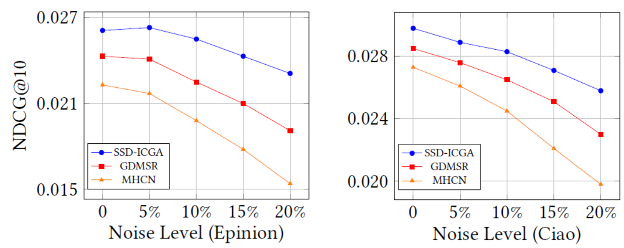

Figure 5: Performance w.r.t different level of social noise.

4.2.3 ***Robustness to Social Noise.*** To substantiate SSD-ICGA's robustness to the social noise, we contaminate the social network by injecting different proportions of fake social connections (i.e., 5%,10%,15%, and 20% of the total social connections) and the user-item interaction graph remains unchanged. We compare SSD-ICDA with the strongest baseline GDMSR (i.e., MHCN with denoised social graph) and its backbone MHCN. The performance under different noise levels is reported in Figure 5. Several observations are noted. **(1)** Overall, the performance of all models declines to varying degrees as more noise is added, which demonstrates SR's vulnerability to social noise. **(2)** Both SSD-ICGA and GDMSR defeat MHCN, showcasing better performance and smaller relative drops as the ratio of injected noise increases, suggesting the effectiveness of denoising for robust SR. **(3)** SSD-ICGA consistently achieves the highest accuracy and exhibits the smallest relative accuracy drops with social noise injected. Moreover, the gap between SSD-ICGA and other methods widens as the noise level increases. This indicates its stronger robustness to social noise, which can be attributed to the IC-based augmentation integrated with the two contrastive tasks. **(4)** The performance of SSD-ICGA on Epinion initially increases with a relatively low level of noise injected, showcasing its ability to extract useful information from the noise to enhance SR. However, escalating noise levels lead to a performance decline, as the negative impact of social noise becomes dominant.

4.2.4 ***Ablation Study.*** We conduct ablation studies to validate the efficacy of key components in SSD-ICGA. The performance is presented in Table 5, where several findings are noted.

Effect of IC-based Graph Augmentation. The IC model generates the augmented social view, which is used as the denoised graph for robust social influence modeling. It considers the activation of

[4]Our code is available at https://github.com/sunyc123r/SSD-ICGA.git.

Table 4: Performance comparison on NDCG@K (N@K) and Recall@K (R@K). We perform t-tests between the strongest baseline (underlined) and SSD-ICGA at the user level, which is statistically significant with p-value < 0.001.

Dataset	Metric	BPR	SBPR	TrustSVD	LightGCN	DiffNet++	MHCN	GoRec	DcRec	DMJP	ESRF	DSL	GDMSR	SSD-ICGA	Improv%
Epinion	N@10	0.0177	0.0164	0.0191	0.0233	0.0198	0.0223	0.0205	0.0213	0.0229	0.0203	0.0231	0.0243	**0.0261**	7.4%
	R@10	0.0304	0.0301	0.0327	0.0381	0.0339	0.0369	0.0347	0.0358	0.0373	0.0341	0.0377	0.0403	**0.0436**	8.2%
	N@20	0.0221	0.0202	0.0241	0.0291	0.0250	0.0271	0.0263	0.0280	0.0286	0.0260	0.0289	0.0307	**0.0331**	7.8%
	R@20	0.0457	0.0448	0.0477	0.0551	0.0492	0.0525	0.0514	0.0537	0.0541	0.0517	0.0565	0.0583	**0.0640**	9.8%
Ciao	N@10	0.0167	0.0188	0.0249	0.0271	0.0221	0.0273	0.0257	0.0263	0.0277	0.0261	0.0272	0.0282	**0.0298**	5.7%
	R@10	0.0274	0.0291	0.0411	0.0431	0.0378	0.0431	0.0422	0.0429	0.0440	0.0423	0.0429	0.0437	**0.0471**	7.0%
	N@20	0.0210	0.0237	0.0298	0.0327	0.0265	0.0335	0.0313	0.0329	0.0341	0.0325	0.0339	0.0345	**0.0370**	7.2%
	R@20	0.0418	0.0440	0.0602	0.0633	0.0567	0.0649	0.0616	0.0637	0.0668	0.0631	0.0649	0.0661	**0.0732**	9.6%

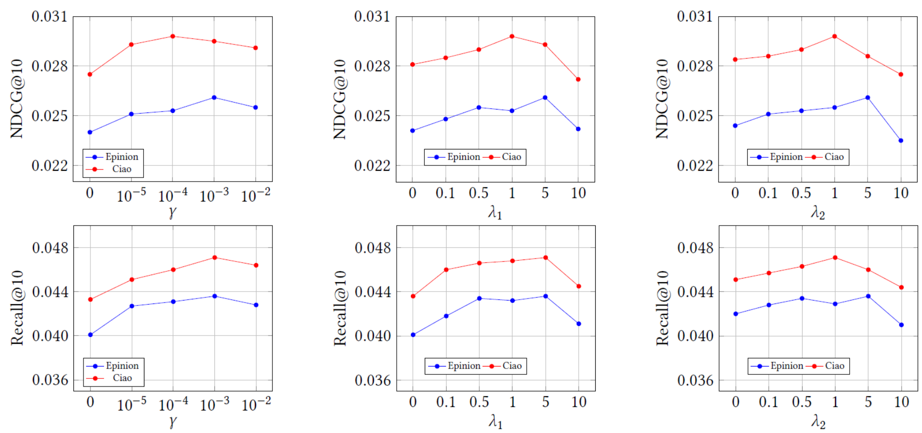

Figure 6: The impacts of key hyper-parameters.

Table 5: Ablation studies of key components in SSD-ICGA.

Dataset	Epinion				Ciao			
Metric	N@10	R@10	N@20	R@20	N@10	R@10	N@20	R@20
SSD-ICGA	**0.0261**	**0.0436**	**0.0331**	**0.0640**	**0.0298**	**0.0471**	**0.0370**	**0.0732**
SSD-ED	0.0252	0.0420	0.0318	0.0614	0.0289	0.0460	0.0356	0.0707
w/o OL hier	0.0254	0.0423	0.0321	0.0618	0.0287	0.0448	0.0350	0.0702
w/o hier	0.0241	0.0401	0.0303	0.0593	0.0281	0.0436	0.0343	0.0681
max MI	0.0255	0.0426	0.0323	0.0617	0.0293	0.0458	0.0359	0.0714
w/o IB	0.0240	0.0403	0.0295	0.0588	0.0275	0.0438	0.0338	0.0675

Table 6: Effect of IB on model's robustness w.r.t NDCG@10.

Dataset	Epinion			Ciao		
Metric	original	*w* noise	drop	original	*w* noise	drop
SSD-ICGA	0.0261	0.0231	11.5%	0.0298	0.0258	13.4%
max MI	0.0255	0.0217	15.0%	0.0293	0.0244	16.7%
w/o IB	0.0240	0.0195	18.7%	0.0275	0.0214	22.2 %

each edge in cascade, which better mimics the information diffusion process. To examine its effectiveness, we compare its variant, which generates the augmented view by dropping out each edge independently based on the attention score (shortened as "SSD-ED"). The performance decline observed in SSD-ED compared with SSD-ICGA helps verify the effectiveness of IC in improving the augmentation quality, thereby benefiting the SR task.

Effect of Hierarchical Contrastive Loss. It is used to prioritize the activation by OLs and high-similarity neighbors in the information diffusion process. To examine its effect, we remove the first term in Eq. 5 ("*w/o* OL hier") and further disable this module by removing the whole loss ("*w/o* hier"). The performance of "*w/o* OL hier" is better than "*w/o* hier", but both are defeated by SSD-ICGA, showcasing the importance of the hierarchical contrastive task in guiding information diffusion.

Effect of Information Bottleneck. Unlike most work [43, 52] that maximizes the mutual information between augmented views, we perform IB-based GCL to minimize the mutual information between the original and augmented views while maximizing the relevance of the augmented view to the SR task. To examine its effect, we compare with two variants: "*max* MI" maximizing the mutual information, and "*w/o* IB" not performing GCL. The result in Table 5 reveals that "*max* MI" outperforms "*w/o* IB", while both are defeated by SSD-ICGA. Moreover, we also examine their robustness to the social noise in two settings: (1) "original" denoting the original social network; and (2) "*w* noise" meaning 20% social noise injected as described in Section 4.2.3. The results in Table 6 show that SSD-ICGA is more robust than the two variants, achieving the best performance in both settings and experiencing the least relative performance drops when noise is injected. The above findings verify the effectiveness of the IB-based contrastive task in extracting relevant social signals for robust SR.

*4.2.5 **Hyper-parameter Sensitivity Analysis**.* We now investigate the impact of essential parameters on SSD-ICGA, including γ, λ_1 and λ_2. In particular, the three parameters control the importance of information bottleneck based contrastive loss, hierarchical contrastive loss, and entropy regularization. The results are plotted

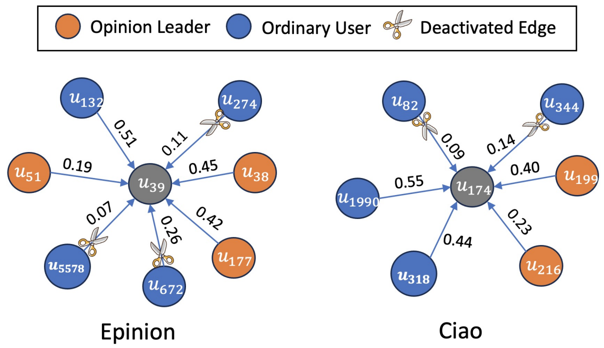

Figure 7: Case Study of Trustee's Activation Probability.

in Figure 6, where we notice that the model performance w.r.t all three parameters exhibit similar trends. Generally, as the values of the parameters increase, the performance initially ascends, attains an optimal level, and eventually declines. On both datasets, the best setting for γ is around 0.01; and the optimal values for λ_1 and λ_2 are around [1, 5]. This underscores the importance of setting proper values to guarantee a more accurate SR.

4.2.6 ***Case study.*** To assess the efficacy of our independent cascade-based graph augmentation, we perform case studies on u_{39} from Epinion and u_{174} from Ciao by visualizing their activation status by each neighbor. The results are given in Figure 7, where the weight indicates the cosine similarity of preference embeddings of each connected user pair. Several observations are made: (1) the deactivated edges are mostly low-similarity ones, e.g., u_{274}, u_{5578} in Epinion and u_{82}, u_{344} in Ciao; (2) Edges involving OLs are kept despite some having low preference similarity with the target user, e.g. u_{51} in Epinion and u_{216} in Ciao. This behavior aligns with the objective of hierarchical contrastive loss, which regularizes the hierarchy of social influence, prioritizing OLs over high-similarity neighbors and high-similarity neighbors over low-similarity neighbors. The effectiveness of such hierarchical regularization has been validated in section 4.2.4.

5 RELATED WORK

GNNs for Social Recommendation (SR). GNNs [9, 36], as powerful graph representation learning methods, have found widespread applications in recommender systems to model the complex interaction between users and item [10, 33, 39]. GNNs also empower social recommenders to model social influence and information diffusion within the social network, including GraphRec [4], DiffNet++ [44], DMJP [29] and DSL [38]. Besides, there are also works that combine GNNs and hypergraphs or heterogeneous graphs to enrich the semantics being modeled and enhance representation learning, such as MHCN [52], SEPT [50] and HGCL [2]. However, GNNs are sensitive to the quality of the input graphs. Most existing studies fail to account for potential noise within social networks while indiscriminately considering all social connections. This compromises the robustness of SR models and undermines their recommendation performance.

Robust Social Recommendation (SR). SR models' vulnerability to noise has been shown in prior studies [49, 52] due to users' intricate social behavior, resulting in the presence of weak ties or noisy social connections. Some attempts to mitigate this issue employ attention mechanism to characterize the importance of each neighbor like GraphRec [4] and DiffNet++ [44]. Others have resorted to differentiating social influence by learning fine-grained disentangled representations such as DMJP [29] and DSR [27]. However, these methods still learn user preferences on the entire graph where weak ties or noisy edges may undermine the quality. Thus, some works turn to acquire a cleaner social graph for the SR task, which either filters the noisy social connections [17, 24, 40, 41] or generates new social connections [49–51]. However, without ground truth labels, both approaches rely on supervision signals derived from the original noisy social networks. Consequently, the relevance of preserved connections to user preference modeling remains uncertain, and incorporating new connections may exacerbate noise levels. Contrarily, our method denoises the social network through self-supervised graph contrastive learning without labels.

Self Supervised Learning in Social Recommendation (SR). Self Supervised Learning (SSL), first introduced in the field of computer vision and natural language processing [21, 26], learns from unlabelled data through pretext tasks and data augmentation. Later works [18, 45] apply SSL in the graph domain to enrich graph representation learning. The common strategy is to perform contrastive learning [12, 45] between different augmented views created through, e.g., edge dropout, node dropout, or feature shuffling. The same idea has been extended to GNN-based recommenders to alleviate the data sparsity issue and enhance representation learning, such as SGL [45] and CGI [42]. Recent works like MHCN [52], SEPT [50], and HGCL [2] have extended the application to SR. Specifically, SSL is mainly used to contrast between social-related channels to enrich the social semantics and ease the data sparsity issue. The potential of SSL in enhancing the robustness of SR models to social noise is, however, under-explored.

6 CONCLUSION

In this work, we propose a novel denoising social recommender, SSD-ICGA, that achieves self-supervised social denoising for a robust social recommendation. Specifically, we utilize the Independent Cascade (IC) model to simulate the influence diffusion in the social network and generate the augmented social graph for robust social influence modeling. Through data analysis, we find that (1) social noise likely originates from low-similarity neighbors; (2) there exists a hierarchy of social influence among neighbors: Opinion Leaders (OLs) > high-similarity neighbors > low-similarity neighbors. Guided by these observations, we integrated the IC augmentation with a hierarchical contrastive loss and an information bottleneck based contrastive loss for effective self-supervision toward a denoised social view. Extensive experiments on two public datasets demonstrate the effectiveness and robustness of SSD-ICGA.

ACKNOWLEDGMENTS

This work is partially supported by the MOE AcRF Tier 1 funding (RG13/23), A*STAR Center for Frontier Artificial Intelligence Research, and the Data Science and Artificial Intelligence Research Center, School of Computer Science and Engineering at Nanyang Technological University, Singapore.

REFERENCES

[1] Seyed Mojtaba Hosseini Bamakan, Ildar Nurgaliev, and Qiang Qu. 2019. Opinion Leader Detection: A methodological review. *Expert Systems with Applications* 115 (2019), 200–222.

[2] Mengru Chen, Chao Huang, Lianghao Xia, Wei Wei, Yong Xu, and Ronghua Luo. 2023. Heterogeneous Graph Contrastive Learning for Recommendation. In *Proceedings of the 16th ACM International Conference on Web Search and Data Mining (WSDM)*. 544–552.

[3] Linus Ericsson, Henry Gouk, Chen Change Loy, and Timothy M Hospedales. 2022. Self-Supervised Representation Learning: Introduction, advances, and challenges. *IEEE Signal Processing Magazine* 39, 3 (2022), 42–62.

[4] Wenqi Fan, Yao Ma, Qing Li, Yuan He, Eric Zhao, Jiliang Tang, and Dawei Yin. 2019. Graph Neural Networks for Social Recommendation. In *Proceedings of the 19th International Conference on World Wide Web (WWW)*. 417–426.

[5] John Guare. 2016. Six Degrees of Separation. In *The Contemporary Monologue: Men*. Routledge, 89–93.

[6] Guibing Guo, Jie Zhang, and Neil Yorke-Smith. 2015. Trustsvd: Collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, Vol. 29.

[7] Michael Gutmann and Aapo Hyvärinen. 2010. Noise-Contrastive Estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings, 297–304.

[8] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. Soft Actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International Conference on Machine Learning (ICML)*. PMLR, 1861–1870.

[9] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive Representation Learning on Large Graphs. *Advances in Neural Information Processing Systems (NeurIPS)* 30 (2017).

[10] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 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 (SIGIR)*. 639–648.

[11] Elad Hoffer and Nir Ailon. 2015. Deep Metric Learning using Triplet Network. In *Similarity-Based Pattern Recognition: 3rd International Workshop (SIMBAD)*. Springer, 84–92.

[12] Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. 2020. A Survey on Contrastive Self-Supervised Learning. *Technologies* 9, 1 (2020), 2.

[13] Elihu Katz. 1957. The Two-step Flow of Communication: An up-to-date report on an hypothesis. *Public Opinion Qquarterly* 21, 1 (1957), 61–78.

[14] David Kempe, Jon Kleinberg, and Éva Tardos. 2003. Maximizing the Spread of Influence through a Social Network. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*. 137–146.

[15] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[16] Mei Li, Xiang Wang, Kai Gao, and Shanshan Zhang. 2017. A Survey on Information Diffusion in Online Social Networks: Models and methods. *Information* 8, 4 (2017), 118.

[17] Chun-Yi Liu, Chuan Zhou, Jia Wu, Yue Hu, and Li Guo. 2018. Social recommendation with an essential preference space. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[18] Yixin Liu, Ming Jin, Shirui Pan, Chuan Zhou, Yu Zheng, Feng Xia, and S Yu Philip. 2022. Graph Self-Supervised Learning: A survey. *IEEE Transactions on Knowledge and Data Engineering (TKDE)* 35, 6 (2022), 5879–5900.

[19] Hao Ma, Haixuan Yang, Michael R Lyu, and Irwin King. 2008. Sorec: Social Recommendation Using Probabilistic Matrix Factorization. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM)*. 931–940.

[20] Miller McPherson, Lynn Smith-Lovin, and James M Cook. 2001. Birds of a Feather: Homophily in Social Networks. *Annual Review of Sociology* (2001), 415–444.

[21] Abdelrahman Mohamed, Hung-yi Lee, Lasse Borgholt, Jakob D Havtorn, Joakim Edin, Christian Igel, Katrin Kirchhoff, Shang-Wen Li, Karen Livescu, Lars Maaløe, et al. 2022. Self-Supervised Speech Representation Learning: A review. *IEEE Journal of Selected Topics in Signal Processing* (2022).

[22] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation Learning with Contrastive Predictive Coding. *arXiv preprint arXiv:1807.03748* (2018).

[23] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The PageRank Citation Ranking : Bringing Order to the Web. In *The Web Conference*. https://api.semanticscholar.org/CorpusID:1508503

[24] Yuhan Quan, Jingtao Ding, Chen Gao, Lingling Yi, Depeng Jin, and Yong Li. 2023. Robust Preference-Guided Denoising for Graph-based Social Recommendation. In *Proceedings of the 32th International Conference on World Wide Web (WWW)*. 1097–1108.

[25] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618* (2012).

[26] Madeline C Schiappa, Yogesh S Rawat, and Mubarak Shah. 2023. Self-Supervised Learning for Videos: A survey. *Comput. Surveys* 55, 13s (2023), 1–37.

[27] Xiao Sha, Zhu Sun, and Jie Zhang. 2021. Disentangling Multi-Facet Social Relations for Recommendation. *IEEE Transactions on Computational Social Systems (TCSS)* (2021).

[28] Youchen Sun. 2023. Denoising Explicit Social Signals for Robust Recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*. 1344–1348.

[29] Youchen Sun, Zhu Sun, Xiao Sha, Jie Zhang, and Yew Soon Ong. 2023. Disentangling Motives behind Item Consumption and Social Connection for Mutually-enhanced Joint Prediction. In *Proceedings of the 17th ACM Conference on Recommender Systems (RecSys)*. 613–624.

[30] Zhu Sun, Hui Fang, Jie Yang, Xinghua Qu, Hongyang Liu, Di Yu, Yew-Soon Ong, and Jie Zhang. 2022. DaisyRec 2.0: Benchmarking Recommendation for Rigorous Evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2022).

[31] Zhu Sun, Di Yu, Hui Fang, Jie Yang, Xinghua Qu, Jie Zhang, and Cong Geng. 2020. Are we evaluating rigorously? Benchmarking Recommendation for Reproducible Evaluation and Fair Comparison. In *Proceedings of the 14th ACM Conference on Recommender Systems (RecSys)*. 23–32.

[32] Jiliang Tang, Huiji Gao, Huan Liu, and Atish Das Sarma. 2012. eTrust: Understanding Trust evolution in an Online World. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*. 253–261.

[33] Shisong Tang, Qing Li, Xiaoteng Ma, Ci Gao, Dingmin Wang, Yong Jiang, Qian Ma, Aoyang Zhang, and Hechang Chen. 2022. Knowledge-based temporal fusion network for interpretable online video popularity prediction. In *Proceedings of the ACM Web Conference 2022*. 2879–2887.

[34] Naftali Tishby, Fernando C Pereira, and William Bialek. 2000. The Information Bottleneck Method. *arXiv preprint physics/0004057* (2000).

[35] Naftali Tishby and Noga Zaslavsky. 2015. Deep Learning and the Information Bottleneck Principle. In *2015 IEEE Information Theory Workshop (ITW)*. IEEE, 1–5.

[36] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph Attention Networks. *arXiv preprint arXiv:1710.10903* (2017).

[37] Jianling Wang, Kaize Ding, Ziwei Zhu, Yin Zhang, and James Caverlee. 2020. Key Opinion Leaders in Recommendation Systems: Opinion elicitation and diffusion. In *Proceedings of the 13th International Conference on Web Search and Data Mining (WSDM)*. 636–644.

[38] Tianle Wang, Lianghao Xia, and Chao Huang. 2023. Denoised self-augmented learning for social recommendation. *arXiv preprint arXiv:2305.12685* (2023).

[39] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 165–174.

[40] Xin Wang, Steven CH Hoi, Martin Ester, Jiajun Bu, and Chun Chen. 2017. Learning Personalized Preference of Strong and Weak Ties for Social Recommendation. In *Proceedings of the 26th International Conference on World Wide Web (WWW)*. 1601–1610.

[41] Xin Wang, Wei Lu, Martin Ester, Can Wang, and Chun Chen. 2016. Social Recommendation with Strong and Weak Ties. In *Proceedings of the 25th ACM International on conference on Information and Knowledge Management*. 5–14.

[42] Chunyu Wei, Jian Liang, Di Liu, and Fei Wang. 2022. Contrastive Graph Structure Learning via Information Bottleneck for Recommendation. *Advances in Neural Information Processing Systems (NeurIPS)* 35 (2022), 20407–20420.

[43] Jiahao Wu, Wenqi Fan, Jingfan Chen, Shengcai Liu, Qing Li, and Ke Tang. 2022. Disentangled Contrastive Learning for Social Recommendation. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management (CIKM)*. 4570–4574.

[44] Le Wu, Junwei Li, Peijie Sun, Richang Hong, Yong Ge, and Meng Wang. 2020. Diffnet++: A neural influence and interest diffusion network for social recommendation. *IEEE Transactions on Knowledge and Data Engineering (TKDE)* (2020).

[45] Lirong Wu, Haitao Lin, Cheng Tan, Zhangyang Gao, and Stan Z Li. 2021. Self-Supervised Learning on Graphs: Contrastive, Generative, or Predictive. *IEEE Transactions on Knowledge and Data Engineering (TKDE)* (2021).

[46] Yang Yang, Jie Tang, Cane Leung, Yizhou Sun, Qicong Chen, Juanzi Li, and Qiang Yang. 2015. Rain: Social Role-aware Information Diffusion. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, Vol. 29.

[47] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. 2020. Graph Contrastive Learning with Augmentations. *Advances in Neural Information Processing Systems (NeurIPS)* 33 (2020), 5812–5823.

[48] Junliang Yu, Min Gao, Jundong Li, Hongzhi Yin, and Huan Liu. 2018. Adaptive Implicit Friends Identification over Heterogeneous Network for Social Recommendation. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM)*. 357–366.

[49] Junliang Yu, Min Gao, Hongzhi Yin, Jundong Li, Chongming Gao, and Qinyong Wang. 2019. Generating Reliable Friends via Adversarial Training to Improve Social Recommendation. In *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 768–777.

[50] Junliang Yu, Hongzhi Yin, Min Gao, Xin Xia, Xiangliang Zhang, and Nguyen Quoc Viet Hung. 2021. Socially-Aware Self-Supervised Tri-Training for Recommendation. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2084–2092.

[51] Junliang Yu, Hongzhi Yin, Jundong Li, Min Gao, Zi Huang, and Lizhen Cui. 2020. Enhancing Social Recommendation with Adversarial Graph Convolutional Networks. *IEEE Transactions on Knowledge and Data Engineering (TKDE)* 34, 8 (2020), 3727–3739.

[52] Junliang Yu, Hongzhi Yin, Jundong Li, Qinyong Wang, Nguyen Quoc Viet Hung, and Xiangliang Zhang. 2021. Self-Supervised Multi-channel Hypergraph Convolutional Network for Social Recommendation. In *Proceedings of the 30th International Conference on World Wide Web (WWW)*. 413–424.

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

[54] Tong Zhao, Julian McAuley, and Irwin King. 2014. Leveraging Social Connections to Improve Personalized Ranking for Collaborative Filtering. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management (CIKM)*. 261–270.

A DATA ANALYSIS

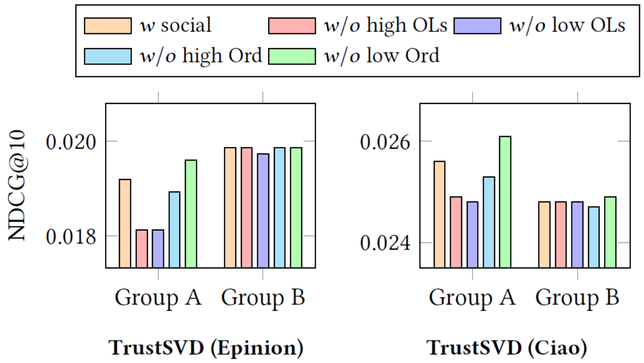

Figure 8: Comparison of the importance of OLs, high-similarity neighbors, and low-similarity neighbors.

In Section 2, we investigate the hierarchy of influence among the OLs, high-similarity neighbors, and low-similarity neighbors, and report the result on the DiffNet++ model in Figure 2. Figure 8 presents the result on another baseline, TrustSVD, where similar trends can be observed.

B ALGORITHM OF SSD-ICGA TRAINING

The training process of SSD-ICGA is given in Algorithm 2.

Algorithm 2: SSD-ICGA Training

Data: $\mathcal{G}_S$, $\mathcal{G}_R$, $\mathcal{A}_0$, batch_size
Result: Model parameters $\boldsymbol{\Theta}$

```
Initialize all parameters;
repeat
    foreach (u, i⁺) ∈ G_R do
        Sample one negative item i⁻ →O^r ∪ (u, i⁺, i⁻);
        Sample (u, k) →O_1^s ∪ (u, k);
        Sample (v, w) →O_2^s ∪ (v, w);
        Sample (v, w') →O_3^s ∪ (v, w');
        Calculate P via Eq. 4;
        Generate G̃_S = IC(G_S, A_0, P) via Algorithm 1;
        Calculate z_u and z̃_u via Eq. 10 and Eq. 11;
        Calculate h_u, h_i via Eq. 2;
        Calculate h̄_u via Eq. 13;
        Calculate L_bpr, L_hier and L_ib via Eq. 14, Eq. 5, and
          Eq. 12;
        Accumulate batch loss by Eq. 15;
        if |O^r| > batch_size then
            Take gradient descent and update Θ;
until converged;
```

C OPTIMAL PARAMETER SETTINGS

Table 7 reports the optimal settings of essential hyper-parameters for all methods compared in the experiment, for reproducibility.

Table 7: Optimal settings of essential hyper-parameters for all approaches.

	Parameter	Epinion	Ciao	Description
BPR	λ_1	0.01	0.001	$l2$ regularization
SBPR	λ_1	0.001	0.001	l_2 regularization
TrustSVD	λ_1	0.01	0.001	l_2 regularization
	γ	0.9	0.9	balance
LightGCN	L_r	2	3	number of interaction layers
	λ_1	0.001	0.001	$l2$ regularization
DiffNet++	λ_1	0.001	0.001	l_2 regularization
	K	3	3	number of stacking layer
MHCN	β	0.01	0.05	balance
	λ_1	0.01	0.0001	l_2 regularization
	L	2	3	GNN depth
GoRec	λ_1	0.01	0.001	l_2 regularization
	L	3	3	number of stacking layer
DcRec	λ_1	0.01	0.01	contrastive loss weight
	λ_2	0.001	0.01	contrastive loss weight
DMJP	P	4	4	#intention facets
	Q	4	4	#relation facets
	λ_1	0.001	0.01	independence regularization
ESRF	L	3	3	# of stacking layer
	γ	0.001	0.0001	balance
DSL	L	3	2	# of propagation layer
	λ_2	0.0001	0.0001	SSL regularization
GDMSR	α	0.5	0.5	balance
	γ	0.2	0.5	adaptive denoising parameter