PIRGL: Prior and Implicit Reliability Graph Learning for Social Recommendation

Yunqi Han[1], Jidong Wang[1], Yunjie Guo[1], Yifan Chen[2*], Zhiyuan Chen[2], Dario Landa-Silva[3]
[1]Universiti Putra Malaysia, Serdang, Malaysia
[2]University of Nottingham Malaysia, Semenyih, Malaysia
[3]University of Nottingham, Nottingham, United Kingdom
yunqihan1014@gmail.com; 17863816026@163.com; gs71599@student.upm.edu.my;
hcxyc1@nottingham.edu.my; zhiyuan.chen@nottingham.edu.my; dario.landasilva@nottingham.ac.uk
[*]Corresponding author: Yifan Chen

***Abstract*—Social recommendation plays a critical role in modern recommender systems. Graph-based methods effectively model relational signals to improve performance. However, real-world interaction data contains abundant noise, while social links remain typically sparse and unreliable. These limitations significantly hinder the robustness of current models. Existing research rarely explores implicit social relations induced by user preferences. To address these challenges, Prior and Implicit Reliability Graph Learning is proposed. This unified framework explicitly models reliability across interaction, relation, and fusion levels. Specifically, the model integrates a self-paced interaction denoising strategy with a hybrid social graph containing preference-based neighbors. A reliability-aware fusion gate further balances these heterogeneous user representations. Experiments on three real-world benchmarks demonstrate consistent improvements in Recall and NDCG. These findings suggest that reliability-oriented graph learning is a promising direction for robust social recommendation.**

***Index Terms*—Social recommendation, graph neural networks, reliability modeling, hybrid social graph, interaction denoising, representation fusion**

I. Introduction

Personalized recommendation serves as a critical mechanism for information filtering in modern online platforms [1]. Collaborative filtering has established itself as a dominant paradigm for implementing such services [2]. This approach frequently suffers from severe data sparsity issues due to insufficient interactions. Incorporating social relations into preference learning provides valuable auxiliary signals to alleviate this limitation [3]. Such heterogeneous relational data naturally forms complex graph structures requiring powerful modeling techniques. Graph neural networks (GNNs) have consequently become the mainstream framework by capturing high-order connectivity patterns [4], [5]. Representative models like LightGCN efficiently capture collaborative signals through message passing on bipartite graphs [6]. Social recommendation approaches further enhance user representations by aggregating information from trust connections [7]. However, most existing studies implicitly assume that the observed interaction and social structures are fully reliable.

In practical scenarios, observed data inevitably suffers from noise and unreliability issues [8]. Implicit feedback contains numerous false positive interactions that mislead model optimization. Recent works have introduced various denoising techniques to improve recommendation robustness [9]. However, most approaches lack training-driven mechanisms to progressively reweight interactions based on model confidence [10]. Consequently, noisy signals continue to adversely affect the optimization process. Social networks also present significant challenges regarding sparsity and noisy connections [11]. Existing models typically fail to enhance such structures using interaction-induced implicit neighbors [7]. Furthermore, frameworks often adopt static fusion strategies to combine heterogeneous user representations [12]. This approach overlooks the varying reliability of information sources for different users.

To address these challenges, Prior and Implicit Reliability Graph Learning (PIRGL) is proposed as a unified framework. This method explicitly models reliability across interaction, relation, and fusion levels. A progressive interaction denoising mechanism is introduced based on the ranking margin of the optimization objective. The contribution of each interaction is adjusted using a self-paced exponential moving average scheme. Furthermore, a hybrid social graph is constructed by integrating explicit trust links with implicit preference-based neighbors. A reliability-weighted propagation strategy is applied to emphasize trustworthy social connections. At the representation level, a reliability-guided fusion module dynamically balances interaction and social signals. Additionally, a contrastive learning objective is incorporated to further enhance representation robustness against perturbations.

Extensive experiments are conducted on multiple real-world benchmarks to evaluate the proposed framework. The results demonstrate consistent improvements over state-of-the-art methods in terms of Recall and NDCG. These findings indicate that reliability-oriented graph learning is a promising direction for robust social recommendation.

II. Related Work

A. Graph-based Collaborative Filtering and Social Recommendation

GNNs have become a dominant paradigm for collaborative filtering by modeling user–item interactions as bipartite graphs and performing high-order message passing. Representative models such as NGCF [4] and LightGCN [6] demonstrate that even simple aggregation schemes can substantially improve recommendation quality, while efficiency- or regularization-oriented variants, including UltraGCN [13] and DirectAU [14], further refine this paradigm.

To alleviate interaction sparsity, social recommendation methods incorporate user–user relations into graph-based models. SEPT [15] and SocialLGN [12] exploit dual interaction–social graph propagation, while GraphRec [7] and DiffNet++ [16] model social influence and diffusion effects. Hypergraph-based approaches such as MHCN [17] further capture higher-order social relations. Despite their effectiveness, most existing methods implicitly assume that interaction and social graphs are reliable, making them sensitive to noisy interactions and sparse or unreliable trust links in real-world settings.

B. Robust and Reliability-aware Learning for Social Recommendation

The presence of noisy interactions and unreliable social links has motivated robustness-oriented learning for social recommendation. DSL [8], Sun et al. [9], and Li et al. [18] explicitly denoise interaction and social information through self-augmented or unified learning frameworks. Other robustness-enhancing strategies include robust graph collaborative filtering [19], stochastic structure regularization such as DropEdge [20], and topology refinement or graph structure learning [21], which primarily focus on improving graph quality.

Self-supervised and contrastive learning have also been adopted to enhance representation robustness. SGL [22], SimGCL [23], and NCL [24] construct contrastive objectives from augmented graphs or embedding perturbations, while bootstrapping-based methods [25] and masked graph modeling [26] provide alternative forms of auxiliary supervision. Although these approaches improve representation stability, they do not explicitly model reliability across interactions, social relations, and representation fusion.

III. METHODOLOGY

This section presents PIRGL, a reliability-aware framework for social recommendation. It learns user representations from interaction and social graphs, and is trained with weighted BPR and auxiliary contrastive regularisation.

A. Preliminaries

Let $\mathcal{U}$ and $\mathcal{V}$ denote the user and item sets. Let $R \in \{0,1\}^{I\times J}$ be the interaction matrix, where $r_{ui} = 1$ indicates an observed interaction, and let $\mathcal{D} = \{(u,i) \mid r_{ui} = 1\}$ be the interaction set.

The user–item bipartite graph is defined as $G_r = \{\mathcal{U} \cup \mathcal{V}, \mathcal{E}_r\}$, where $\mathcal{E}_r = \mathcal{D}$. Following LightGCN, message passing is performed using the normalised adjacency matrix:

$$\widehat{\mathbf{A}}_r = \mathbf{D}_r^{-1/2}\mathbf{A}_r\mathbf{D}_r^{-1/2}. \tag{1}$$

Let S denote the social relation matrix and $G_s = \{\mathcal{U}, \mathcal{E}_{trust}\}$ the observed social graph. To alleviate sparsity, preference-based edges are added to form the enhanced social graph. Specifically, the enhanced social edge set is defined as:

$$\widetilde{\mathcal{E}}_s = \mathcal{E}_{trust} \cup \mathcal{E}_{pref}. \tag{2}$$

Accordingly, the enhanced social graph is given by $\widetilde{G}_s = \{\mathcal{U}, \widetilde{\mathcal{E}}_s\}$. Each social edge $(u,u') \in \widetilde{\mathcal{E}}_s$ is associated with a relation reliability weight $w_{uu'} \in (0,1)$.

Let $\mathbf{P} \in \mathbb{R}^{I\times d}$ and $\mathbf{Q} \in \mathbb{R}^{J\times d}$ be trainable user and item embeddings. The predicted preference score is computed as:

$$\hat{y}_{ui} = \mathbf{e}_u^\top \mathbf{e}_i \tag{3}$$

PIRGL models three types of reliability: interaction reliability w_{ui}, relation reliability $w_{uu'}$, and fusion reliability λ_u^{eff}.

B. Model Framework

Figure 1 illustrates the overall architecture of PIRGL, which adopts a dual-level reliability modelling framework for social recommendation, covering structure-level and embedding-level reliability.

At the structure level, PIRGL performs dual-graph representation learning on the interaction graph G_r and the enhanced social graph $\widetilde{G}_s$. For the interaction graph, stochastic interaction views generated via user-adaptive edge dropout are encoded by shared LightGCN layers to support representation learning and contrastive regularisation. For the social graph, preference-based neighbours constructed by KNN are integrated with observed social relations and encoded by LightGCN to obtain social-view user representations.

At the embedding level, interaction-view and social-view representations are fused through a reliability-aware gate λ_u^{eff} to form the final user embedding. Training is guided by a weighted BPR objective with self-paced interaction reweighting, together with an auxiliary InfoNCE-based contrastive loss for representation stabilisation.

1) Structure-level Reliability Modeling: **Dual-Graph Representation Learning.** PIRGL learns representations from two complementary graphs, namely the interaction graph G_r and the enhanced social graph $\widetilde{G}_s$. For the interaction view, LightGCN-style propagation is adopted. Let $\mathbf{E}^{(0)} = [\mathbf{P};\mathbf{Q}]$ denote the initial embedding matrix formed by concatenating user and item embeddings. The layer-wise propagation is defined as:

$$\mathbf{E}^{(l+1)} = \widehat{\mathbf{A}}_r\mathbf{E}^{(l)} \tag{4}$$

In Eq.(4), l denotes the layer index, $\mathbf{E}^{(l)}$ is the embedding matrix at layer l, and $\widehat{\mathbf{A}}_r$ is the normalised adjacency matrix defined in Eq.(1), which propagates information along interaction edges.

After L_r layers, the interaction-view embedding is obtained by layer-wise averaging:

$$\mathbf{E}^{UI} = \frac{1}{L_r+1}\sum_{l=0}^{L_r}\mathbf{E}^{(l)} \tag{5}$$

In Eq.(5), L_r denotes the number of propagation layers and $\mathbf{E}^{UI}$ is the aggregated interaction-view embedding.

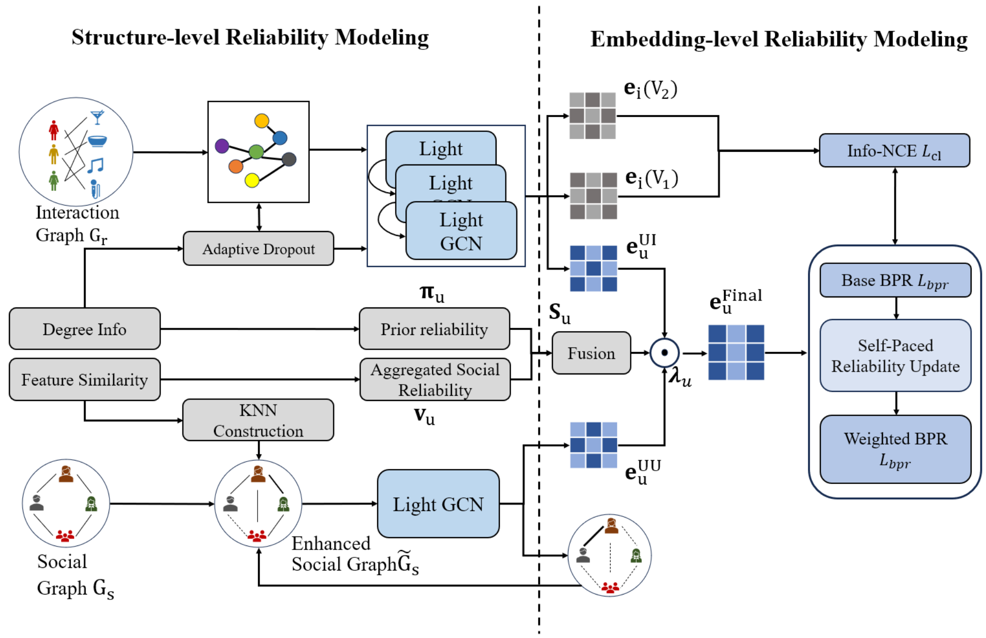

Fig. 1: Overall Framework of PIRGL.

For the social view, LightGCN-style propagation is also applied on the enhanced social graph $\widetilde{G}_s$. Degree-based normalisation is computed on the unweighted graph for stability, while relation reliability weights $w_{uu'}$ are injected multiplicatively to modulate neighbour contributions.

Stochastic Interaction Views. To improve robustness and construct contrastive views, stochastic interaction graphs are generated via user-adaptive edge dropout. Let d_u denote the interaction degree of user u. We define the degree-informed prior signal as

$$\pi_u = \frac{d_u}{\max_v d_v + \epsilon}, \tag{6}$$

In Eq. (6), ϵ is a small constant. Thus, a larger π_u indicates higher interaction confidence. The dropout probability is defined as:

$$p_u = \mathrm{clip}(\alpha_0 + \alpha_1(1-\pi_u), 0, p_{\max}) \tag{7}$$

In Eq. (7), α_0 controls the base dropout intensity, α_1 determines how strongly the dropout probability is adjusted by the degree-informed prior signal π_u, and $p_{\max}$ bounds the maximum dropout probability.

Edges are sampled with probability $1-p_u$ to generate two stochastic interaction views, denoted as V_1 and V_2. The same encoder is applied to both views to obtain view-specific embeddings, which are used as positive pairs for contrastive learning.

Prior-Enhanced Social Graph Construction. Since observed trust relations $\mathcal{E}_{trust}$ are often sparse, preference-based neighbours are constructed from interaction behaviours to complement the social graph. The construction uses training interactions only to avoid test leakage. Let $\mathcal{I}(u)$ denote the set of items interacted with by user u in the training set. Preference similarity between users u and u' is defined as:

$$\mathrm{sim}(u,u') = \frac{|\mathcal{I}(u) \cap \mathcal{I}(u')|}{\sqrt{|\mathcal{I}(u)|}\sqrt{|\mathcal{I}(u')|}} \tag{8}$$

In Eq.(7), the numerator counts co-interacted items, while the denominator normalises interaction frequencies to reduce bias toward highly active users.

Based on $\mathrm{sim}(u,u')$, the top-K most similar users are selected to form the preference-based edge set $\mathcal{E}_{pref}$. The enhanced social edge set is given by $\widetilde{\mathcal{E}}_s = \mathcal{E}_{trust} \cup \mathcal{E}_{pref}$. Each edge $(u,u') \in \widetilde{\mathcal{E}}_s$ is associated with a relation reliability weight $w_{uu'}$, where trust edges are initialised with higher weights than preference edges. Relation reliabilities can be further refined during end-to-end training.

2) Embedding-level Reliability Modeling: **Reliability-aware Fusion Gate.** Not all users benefit equally from social information. Therefore, social signals are injected into the interaction-view representation in a residual and reliability-aware manner. Let $\mathbf{e}_u^{UI}$ denote the interaction-view user representation and let $\mathbf{e}_u^{UU}$ denote the social-view user representation. The social residual is defined as:

$$\Delta_u = \gamma(\mathbf{e}_u^{UU} - \mathbf{e}_u^{UI}) \tag{9}$$

In Eq.(8), Δ_u represents the residual social correction for user u, and γ is a scaling factor that controls the magnitude of the residual signal.

The final user representation is obtained by:

$$\mathbf{e}_u = \mathbf{e}_u^{UI} + \lambda_u^{\mathrm{eff}} \Delta_u \tag{10}$$

In Eq.(9), λ_u^{eff} is the user-specific fusion gate that controls the contribution of social information, and $\mathbf{e}_u$ denotes the final user embedding used for preference prediction in Eq.(3).

To compute λ_u^{eff}, two reliability cues are considered. The first is an implicit social reliability signal ν_u, which is computed as the mean of incident relation reliabilities $w_{uu'}$ on the enhanced social graph. The second is the interaction prior π_u, reflecting the interaction degree information of user u. They are combined as:

$$s_u = \alpha\nu_u + (1-\alpha)\pi_u \tag{11}$$

In Eq.(10), s_u denotes the combined reliability score that integrates social and interaction-level cues. The coefficient α controls the relative importance of ν_u and π_u. To ensure a valid convex combination, α is parameterised as $\alpha = \sigma(\ell_\alpha)$, where ℓ_α is a learnable scalar parameter.

A base fusion gate is defined as $\lambda_u = \sigma(\ell_u)$, where ℓ_u is a learnable parameter for user u. The effective fusion gate is computed as:

$$\lambda_u^{\text{eff}} = \text{clip}(\lambda_u + \beta\tilde{s}_u, \epsilon, \lambda_{\max}) \tag{12}$$

In Eq.(11), $\tilde{s}_u$ denotes the centred version of s_u, β controls the adjustment strength, ϵ prevents degenerate gates, and $\lambda_{\max}$ bounds the maximum injection strength.

Self-paced Interaction Denoising. Implicit feedback often contains noisy interactions. Each observed interaction (u, i) is therefore assigned an interaction reliability weight $w_{ui} \in [w_{\min}, 1]$. For each training triple (u, i, j), the BPR margin is defined as:

$$m_{uij} = \mathbf{e}_u^\top \mathbf{e}_i - \mathbf{e}_u^\top \mathbf{e}_j \tag{13}$$

In Eq.(12), j denotes a sampled negative item for user u, and m_{uij} measures how confidently the model ranks the positive item i over the negative item j.

Based on this margin, the weighted BPR loss is defined as:

$$\mathcal{L}_{BPR} = \mathbb{E}_{(u,i,j)}\Big[w_{ui} \cdot \big(-\log\sigma(m_{uij})\big)\Big] \tag{14}$$

In Eq.(13), $\mathbb{E}_{(u,i,j)}[\cdot]$ denotes the expectation over sampled training triples, and w_{ui} downweights potentially unreliable interactions during optimisation.

The target interaction reliability is computed as:

$$\hat{w}_{ui} = \text{clip}\Big(\sigma\big(\frac{m_{uij} - \mathbb{E}_{\mathcal{B}}[m]}{T_0}\big), w_{\min}, 1\Big) \tag{15}$$

In Eq. (14), $\mathbb{E}_{\mathcal{B}}[m]$ denotes the mean margin within the current mini-batch $\mathcal{B}$, T_0 is a temperature constant that controls the smoothness of the mapping, and $w_{\min}$ is the minimum allowed reliability weight. Although $\hat{w}_{ui}$ is estimated from the sampled BPR margin in the current mini-batch, the exponential moving average in Eq. (15) helps smooth short-term variance across iterations.

The interaction reliability weight is updated using an exponential moving average:

$$w_{ui} = \beta_w w_{ui} + (1-\beta_w)\hat{w}_{ui} \tag{16}$$

In Eq.(15), $\beta_w \in (0, 1)$ is the moving average coefficient that controls the update inertia. The update starts after a short warm-up period to improve training stability.

TABLE I: Statistics of datasets.

Dataset	Douban-Book	Douban-Movie	Ciao
Users	13,024	2,848	7,355
Items	22,347	39,586	17,867
Interactions	792,026	894,887	112,452
Social Relations	169,150	35,770	223,356
Interaction Density	0.048%	0.794%	0.085%
Relation Density	0.099%	0.441%	0.412%

C. Joint Optimization Target

1) Weighted BPR Loss: The weighted BPR loss is defined in Eq.(13). It incorporates interaction reliability w_{ui} to reduce the influence of noisy interactions and supports progressive denoising during training.

2) Contrastive Regularization: To improve robustness against graph perturbations, two stochastic interaction views V_1 and V_2 are constructed. Let $\mathbf{e}_u^{(V_1)}$ and $\mathbf{e}_u^{(V_2)}$ denote the user embeddings obtained from V_1 and V_2, respectively. The contrastive loss is defined as:

$$\mathcal{L}_{cl} = -\sum_{u\in\mathcal{B}} \log \frac{\exp(\text{sim}(\mathbf{e}_u^{(V_1)}, \mathbf{e}_u^{(V_2)})/\tau)}{\sum_{u'\in\mathcal{B}} \exp(\text{sim}(\mathbf{e}_u^{(V_1)}, \mathbf{e}_{u'}^{(V_2)})/\tau)} \tag{17}$$

In Eq.(16), $\mathcal{B}$ denotes the set of users in the current mini-batch, $\text{sim}(\cdot,\cdot)$ is a similarity function, and τ is the temperature parameter. The numerator corresponds to a positive pair across two views, while the denominator treats other users in the mini-batch as negatives.

3) Overall Training Objective: All components in PIRGL are trained end-to-end under the following joint objective:

$$\mathcal{L} = \mathcal{L}_{BPR} + \alpha_{cl}\mathcal{L}_{cl} + \lambda\|\Theta\|_2^2 \tag{18}$$

In Eq.(17), $\mathcal{L}_{BPR}$ and $\mathcal{L}_{cl}$ are defined in Eq.(13) and Eq.(16), respectively. α_{cl} controls the contribution of contrastive regularisation, λ is the coefficient of ℓ_2 regularisation, and Θ denotes the set of trainable parameters.

IV. EXPERIMENT

A. Experimental Setup

Experiments are conducted on three widely used real-world social recommendation datasets: Douban-Book, Douban-Movie, and Ciao. The dataset statistics are summarised in Table I.

The proposed method is compared with several representative baselines, including MF, NGCF [4], LightGCN [6], DiffNet++ [16], MHCN [17], SEPT [15], and GBSR [27].

All methods are implemented in PyTorch and optimised using Adam, with the embedding dimension fixed to 64. For PIRGL, the batch size is set to 2048, and the learning rates are 1×10^{-3} for main parameters and 1×10^{-2} for learnable social edge reliability parameters. The hyperparameters of baseline methods follow the original papers or released implementations.

Evaluation follows the standard top-K recommendation protocol. For each user, all unobserved items are ranked with training interactions masked, corresponding to full-ranking evaluation. Recall@K and NDCG@K are reported with $K =$ 10 and 20. For pairwise learning methods, BPR-style training with negative sampling is adopted.

TABLE II: Performance comparison on three datasets.

Models	Douban-Book				Douban-Movie				Ciao			
	R@10	N@10	R@20	N@20	R@10	N@10	R@20	N@20	R@10	N@10	R@20	N@20
MF	0.0836	0.0948	0.1275	0.1040	0.0471	0.1965	0.0776	0.1857	0.0405	0.0345	0.0629	0.0412
NGCF	0.0813	0.1015	0.1261	0.1112	0.0507	0.2315	0.0830	0.2161	0.0361	0.0263	0.0562	0.0326
LightGCN	0.1013	0.1166	0.1489	0.1251	0.0526	0.2507	0.0880	0.2306	0.0453	0.0330	0.0668	0.0453
DiffNet++	0.0845	0.1068	0.1272	0.1144	0.0473	0.2049	0.0816	0.1944	0.0317	0.0230	0.0502	0.0290
MHCN	0.1029	0.1427	0.1504	0.1473	0.0541	0.2409	0.0909	0.2251	0.0421	0.0349	0.0724	0.0444
SEPT	0.1017	0.1385	0.1510	0.1444	0.0574	0.2672	0.0932	0.2466	0.0463	0.0393	0.0723	0.0468
GBSR	0.1201	0.1459	0.1702	0.1530	0.0555	0.2495	0.0918	0.2306	0.0477	0.0404	0.0737	0.0481
PIRGL (Ours)	**0.1210**	**0.1615**	**0.1746**	**0.1660**	**0.0650**	**0.2816**	**0.1028**	**0.2598**	**0.0611**	**0.0514**	**0.0919**	**0.0608**

TABLE III: Ablation study on Douban-Book and Douban-Movie datasets (seed = 2025).

Variants	Douban-Book				Douban-Movie			
	R@10	N@10	R@20	N@20	R@10	N@10	R@20	N@20
w/o Pref-KNN	0.1101	0.1498	0.1648	0.1551	0.0587	0.2713	0.0965	0.2525
w/o Gate	0.1180	0.1575	0.1736	0.1629	0.0611	0.2752	0.0989	0.2553
w/o Self-Paced	0.1123	0.1529	0.1641	0.1567	0.0574	0.2652	0.0958	0.2462
PIRGL (Full)	**0.1210**	**0.1615**	**0.1746**	**0.1660**	**0.0650**	**0.2816**	**0.1028**	**0.2598**

B. Overall Comparison

Experimental results are reported in Table II, where the best performance is highlighted in bold. PIRGL achieves consistent improvements over the baselines across most settings, with especially notable gains on Douban-Movie and Ciao. This demonstrates the effectiveness of multi-level reliability modelling for social recommendation under noisy interactions and unreliable social relations.

Compared with graph-based collaborative filtering methods such as NGCF and LightGCN, PIRGL generally achieves stronger performance. These methods treat all interactions as equally reliable, which can amplify noise under implicit feedback, whereas PIRGL progressively suppresses unreliable interactions through self-paced reweighting, leading to more stable optimisation and stronger top-K performance.

Compared with social recommendation baselines, PIRGL better handles sparse and heterogeneous social relations by incorporating preference-based neighbour augmentation and explicit relation reliability modelling, reducing the impact of low-quality social links.

Performance gains vary with data characteristics. On Douban-Book, preference-based augmentation and self-paced denoising play a dominant role due to extremely sparse interactions. On Douban-Movie, the reliability-aware fusion gate becomes more important as interactions are denser but social relations remain sparse. On Ciao, relation-level reliability modelling is critical for handling heterogeneous social links.

Overall, these results indicate that PIRGL achieves consistent improvements across datasets with different sparsity and social reliability conditions.

C. Ablation and Sensitivity Analyses

This subsection evaluates the contribution of key components in PIRGL and analyses its sensitivity to important hyperparameters.

Ablation study. Several representative variants are constructed, including w/o Pref-KNN, w/o Gate, and w/o Self-Paced. The ablation results on Douban-Book and Douban-Movie are summarised in Table III. Removing Pref-KNN results in the largest performance drop, highlighting the importance of preference-based neighbours in compensating sparse social relations. Replacing the adaptive fusion gate with a fixed weight also causes consistent degradation, indicating the necessity of user-specific reliability control. Disabling the self-paced strategy further degrades performance, demonstrating the role of progressive interaction denoising. Overall, all components contribute complementary and non-trivial improvements.

Sensitivity analysis. Sensitivity analysis is conducted on the Douban-Book dataset. As shown in Fig. 2(a), performance improves with increasing Pref-KNN size K and peaks around $K = 60$, after which it gradually declines, indicating a balance between information enrichment and noise. Fig. 2(b) shows that performance is stable over a wide range of the self-paced inertia parameter β_{ui}, with the best result around 0.45, reflecting a trade-off between adaptivity and stability. Similarly, Fig. 2(c) shows that overly large fusion gate driver β_λ degrades performance, suggesting that aggressive gate adjustment may amplify unreliable social signals.

V. CONCLUSION

This paper proposes PIRGL as a unified social recommendation framework. The model explicitly captures reliability across interaction, relation, and fusion levels. Specifically, it integrates progressive interaction denoising with hybrid social graph enhancement to handle noise. A reliability-aware fusion gate further adapts user representations under heterogeneous conditions. Additionally, a contrastive objective stabilizes learning against graph perturbations. Extensive experiments on real-world benchmarks demonstrate consistent improvements over state-of-the-art methods. Ablation studies validate the contribution of each specific design. Future research will address computational scalability challenges and explore temporal reliability evolution in dynamic networks. Overall, this work suggests that reliability-oriented modelling is a promising direction for robust graph-based recommendation.

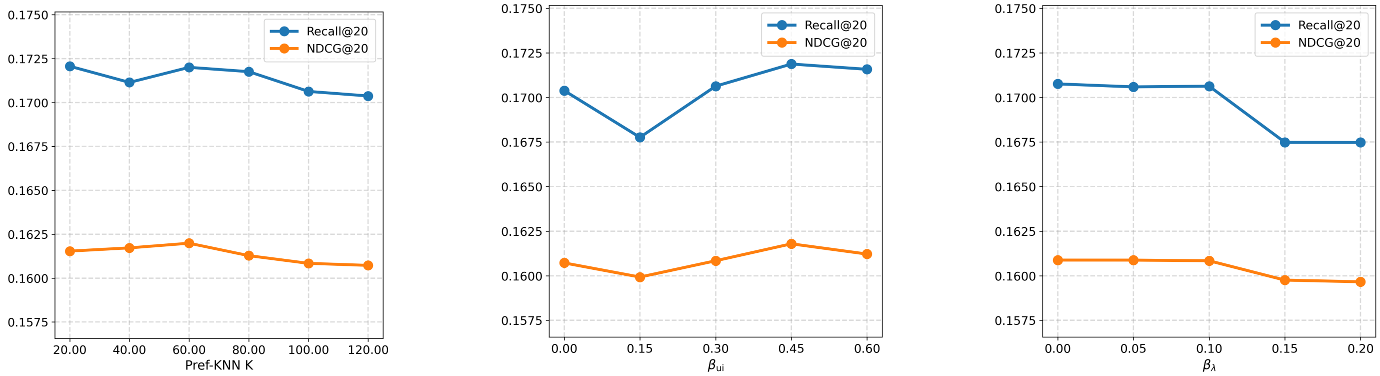

Fig. 2: Sensitivity analysis on the Douban-Book dataset (seed = 2025). (a) Effect of the Pref-KNN size K. (b) Effect of the self-paced inertia parameter β_{ui}. (c) Effect of the fusion gate driver β_{λ}.

REFERENCES

[1] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[2] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Advances in artificial intelligence*, vol. 2009, no. 1, p. 421425, 2009.

[3] J. Tang, X. Hu, and H. Liu, "Social recommendation: a review," *Social network analysis and mining*, vol. 3, no. 4, pp. 1113–1133, 2013.

[4] X. Wang, X. He, M. Wang, F. Feng, and T.-S. Chua, "Neural graph collaborative filtering," in *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*, pp. 165–174, 2019.

[5] Y. Han, Y. Chen, H. Ruan, D. Song, H. Xu, and H. Zhu, "Rs-stgcn: Regional-synergy spatio-temporal graph convolutional network for emotion recognition," *Frontiers in Neuroscience*, vol. 19, p. 1704476, 2025.

[6] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang, "Lightgcn: Simplifying and powering graph convolution network for recommendation," in *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pp. 639–648, 2020.

[7] W. Fan, Y. Ma, Q. Li, Y. He, E. Zhao, J. Tang, and D. Yin, "Graph neural networks for social recommendation," in *The world wide web conference*, pp. 417–426, 2019.

[8] T. Wang, L. Xia, and C. Huang, "Denoised self-augmented learning for social recommendation," *arXiv preprint arXiv:2305.12685*, 2023.

[9] Y. Sun, Z. Sun, Y. Du, J. Zhang, and Y. S. Ong, "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*, pp. 2806–2817, 2024.

[10] T. Huang, Y. Dong, M. Ding, Z. Yang, W. Feng, X. Wang, and J. Tang, "Mixgcf: An improved training method for graph neural network-based recommender systems," in *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pp. 665–674, 2021.

[11] M. Jamali and M. Ester, "A matrix factorization technique with trust propagation for recommendation in social networks," in *Proceedings of the fourth ACM conference on Recommender systems*, pp. 135–142, 2010.

[12] J. Liao, W. Zhou, F. Luo, J. Wen, M. Gao, X. Li, and J. Zeng, "Sociallgn: Light graph convolution network for social recommendation," *Information Sciences*, vol. 589, pp. 595–607, 2022.

[13] K. Mao, J. Zhu, X. Xiao, B. Lu, Z. Wang, and X. He, "Ultragcn: ultra simplification of graph convolutional networks for recommendation," in *Proceedings of the 30th ACM international conference on information & knowledge management*, pp. 1253–1262, 2021.

[14] C. Wang, Y. Yu, W. Ma, M. Zhang, C. Chen, Y. Liu, and S. Ma, "Towards representation alignment and uniformity in collaborative filtering," in *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 1816–1825, 2022.

[15] J. Yu, H. Yin, M. Gao, X. Xia, X. Zhang, and N. Q. Viet Hung, "Socially-aware self-supervised tri-training for recommendation," in *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pp. 2084–2092, 2021.

[16] L. Wu, J. Li, P. Sun, R. Hong, Y. Ge, and M. Wang, "Diffnet++: A neural influence and interest diffusion network for social recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 10, pp. 4753–4766, 2020.

[17] J. Yu, H. Yin, J. Li, Q. Wang, N. Q. V. Hung, and X. Zhang, "Self-supervised multi-channel hypergraph convolutional network for social recommendation," in *Proceedings of the web conference 2021*, pp. 413–424, 2021.

[18] A. Li and B. Yang, "Dual graph denoising model for social recommendation," in *Proceedings of the ACM on Web Conference 2025*, pp. 347–356, 2025.

[19] S. Peng and T. Mine, "A robust hierarchical graph convolutional network model for collaborative filtering," *arXiv preprint arXiv:2004.14734*, 2020.

[20] Y. Rong, W. Huang, T. Xu, and J. Huang, "Dropedge: Towards deep graph convolutional networks on node classification," in *International Conference on Learning Representations*, 2020.

[21] Y. Chen, L. Wu, and M. Zaki, "Iterative deep graph learning for graph neural networks: Better and robust node embeddings," *Advances in neural information processing systems*, vol. 33, pp. 19314–19326, 2020.

[22] J. Wu, X. Wang, F. Feng, X. He, L. Chen, J. Lian, and X. Xie, "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, 2021.

[23] J. Yu, H. Yin, X. Xia, T. Chen, L. Cui, and Q. V. H. Nguyen, "Are graph augmentations necessary? simple graph contrastive learning for recommendation," in *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, pp. 1294–1303, 2022.

[24] Z. Lin, C. Tian, Y. Hou, and W. X. Zhao, "Improving graph collaborative filtering with neighborhood-enriched contrastive learning," in *Proceedings of the ACM web conference 2022*, pp. 2320–2329, 2022.

[25] D. Lee, S. Kang, H. Ju, C. Park, and H. Yu, "Bootstrapping user and item representations for one-class collaborative filtering," in *Proceedings of the 44th international ACM SIGIR conference on Research and Development in information retrieval*, pp. 317–326, 2021.

[26] J. Li, R. Wu, W. Sun, L. Chen, S. Tian, L. Zhu, C. Meng, Z. Zheng, and W. Wang, "What's behind the mask: Understanding masked graph modeling for graph autoencoders," in *KDD*, pp. 1268–1279, ACM, 2023.

[27] Y. Yang, L. Wu, Z. Wang, Z. He, R. Hong, and M. Wang, "Graph bottlenecked social recommendation," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 3853–3862, 2024.