Graph Transformer for Recommendation

Chaoliu Li
chaoliuli66@gmail.com
South China University of Technology
Guangzhou, China

Lianghao Xia
aka_xia@foxmail.com
University of Hong Kong
Hong Kong SAR, China

Xubin Ren
xubinrencs@gmail.com
University of Hong Kong
Hong Kong SAR, China

Yaowen Ye
elwin@connect.hku.hk
University of Hong Kong
Hong Kong SAR, China

Yong Xu
yxu@scut.edu.cn
South China University of Technology
Guangzhou, China

Chao Huang[*]
chaohuang75@gmail.com
University of Hong Kong
Hong Kong SAR, China

ABSTRACT

This paper presents a novel approach to representation learning in recommender systems by integrating generative self-supervised learning with graph transformer architecture. We highlight the importance of high-quality data augmentation with relevant self-supervised pretext tasks for improving performance. Towards this end, we propose a new approach that automates the self-supervision augmentation process through a rationale-aware generative SSL that distills informative user-item interaction patterns. The proposed recommender with Graph TransFormer (GFormer) that offers parameterized collaborative rationale discovery for selective augmentation while preserving global-aware user-item relationships. In GFormer, we allow the rationale-aware SSL to inspire graph collaborative filtering with task-adaptive invariant rationalization in graph transformer. The experimental results reveal that our GFormer has the capability to consistently improve the performance over baselines on different datasets. Several in-depth experiments further investigate the invariant rationale-aware augmentation from various aspects. The source code for this work is publicly available at: https://github.com/HKUDS/GFormer.

CCS CONCEPTS

• **Information systems → Recommender systems**.

KEYWORDS

Recommendation, Graph Transformer, Masked Autoencoder

ACM Reference Format:
Chaoliu Li, Lianghao Xia, Xubin Ren, Yaowen Ye, Yong Xu, and Chao Huang. 2023. Graph Transformer for Recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23), July 23–27, 2023, Taipei, Taiwan.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3539618.3591723

1 INTRODUCTION

Self-supervised learning (SSL) has become a popular solution for addressing the label scarcity issue in recommender systems by generating auxiliary supervision signals from unlabeled data [23, 41]. By integrating with graph neural network (GNN) architecture for collaborative filtering, SSL-enhanced graph augmentation has proven effective in modeling user-item interactions with limited training labels. Among contemporary methods, graph contrastive learning (GCL) [35, 42] is one of the most widely used augmentation paradigms for recommendation [1, 16, 25]. The key insight behind GCL-based recommendation models is to obtain supervision signals from auxiliary learning tasks, which aim to supplement the main recommendation objective via SSL-enhanced co-training.

Existing graph contrastive methods aim to maximize mutual information by achieving representation consistency between generated positive samples (*e.g.*, user self-discrimination), and minimizing similarity between negative pairs (*e.g.*, different users). Recent efforts have attempted to contrast different structural views of the user-item interaction graph with heuristic-based data augmentors, following the principle of mutual information maximization. For instance, SGL [35] proposes to corrupt graph structures by randomly removing user and item nodes as well as their connections to construct topological contrastive views. However, blindly corrupting graph topological structures can lead to the loss of crucial relations between users and items, such as unique user interaction patterns or limited labels of long-tail items (as illustrated in Figure 1.(a)). Therefore, it is crucial to explicitly provide essential self-supervision signals for learning informative representations, which requires invariant rationales in the designed augmentors.

From the perspective of aligning local-level and global-level embeddings for augmentation, some research studies obtain semantic-related subgraph representations through various information aggregation techniques, such as hypergraph-based message passing in HCCF [28] and EM algorithm-based node clustering in NCL [13]. However, due to their hand-crafted nature, the quality of augmentation is likely to be influenced by manually constructed hypergraph structures and user cluster settings. As a result, these augmentation schemes are insufficient to regularize the training process with useful self-supervised signals (*e.g.*, truly negative pairs and hard augmented instances). Moreover, these manually designed contrastive methods can be easily misled by commonly existing noise (*e.g.*, misclick behaviors [18], popularity bias [39]) (as shown in Figure 1.(b)). Introducing augmented SSL information from biased

[*] Chao Huang is the corresponding author. This work was completed when Chaoliu Li was a research intern under the supervision of Chao Huang.

SIGIR '23, July 23–27, 2023, Taipei, Taiwan

ACM ISBN 978-1-4503-9408-6/23/07…$15.00
https://doi.org/10.1145/3539618.3591723

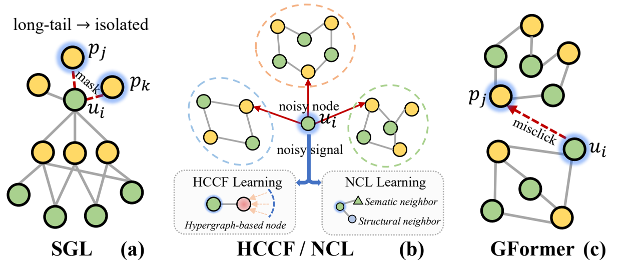


Figure 1: Illustration for the graph augmentations generated by different self-supervised recommendation models.

data can amplify the noisy effects, which dilutes the learning of true user-item interaction patterns. Therefore, existing solutions may fall short in adapting the self-supervision process to changing practical recommendation environments.

Despite the advancements in SSL-enhanced recommender systems, a fundamental question remains poorly understood: *What information is crucial and should be preserved for self-supervised augmentation in recommendation?* Motivated by the recent success of masked autoencoding (MAE) techniques in advancing self-supervised learning [3, 4, 7], this work explores the above question from the perspective of generative self-supervised augmentation with rationale-aware invariant representation learning. Unlike contrastive learning, the masked autoencoder paradigm directly adopts the reconstruction objective as the principled pretext task for data augmentation. It naturally avoids the limitations of manually-generated contrastive views for data augmentation discussed above.

Present Work. In this work, we propose a new recommender system with Graph TransFormer to automatically distill masked self-supervised signals with invariant collaborative rationales. We take inspiration from rationale discovery [26, 40] to bridge the gap between the graph masked autoencoder with adaptive augmentation. Our GFormer makes full use of the power of Transformer in explicitly encoding pairwise relations to discover useful self-supervised signals benefiting the downstream recommendation task, with their own rationales explained. Specifically, we develop a topology-aware graph transformer to integrate into the user-item interaction modeling, enabling automated collaborative rationale discovery. In GFormer, the topological information of the user-item relation graph is treated as the global context in the form of graph positional encoding. To adapt GFormer to diverse recommendation environments, it learns to form appropriate interaction patterns as self-supervision signals, guided by task-adaptive collaborative rationale discovery. Our contributions can be summarized as follows:

- This work revisits the self-supervised recommendation by exploring augmentation schemes from SSL-enhanced collaborative rationalization. We not only realize the automated data augmentors in SSL, but also provide rationale-aware understanding behind the self-supervised augmentation to improve model robustness.
- We propose a principled approach for discovering invariant rationales with collaborative relations over the graph transformer. Task-aware adaptation is introduced to alleviate the issue of data-level variance. Then, the graph autoencoder is required to reconstruct the masked user-item interactions for augmentation.
- We validate the effectiveness of our GFormer on several datasets. Compared with a variety of strong compared methods, our method consistently gains improvements across different settings.

2 PRELIMINARIES AND RELATED WORK

Graph-based Collaborative Filtering. Many recent studies have explored the use of graph representation learning in building graph-enhanced collaborative filtering (CF) models to capture high-order collaborative relations [2, 5, 27]. In this scenario, we assume that there are I users $\mathcal{U} = \{u_1, u_2, ..., u_I\}$ and J items $\mathcal{P} = \{p_1, p_2, ..., p_J\}$ in our recommendation system. The observed user behaviors are represented by an interaction matrix $\mathbf{A} \in \mathbb{R}^{I \times J}$, where $a_{i,j} = 1$ if an interaction between user u_i and item p_j is observed, and $a_{i,j} = 0$ otherwise. To transform the interaction matrix into an interaction graph for graph-based CF, we define a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V} = \mathcal{U} \cup \mathcal{P}$ forms the node set, and $\mathcal{E} = \{e_{i,j} | a_{i,j} = 1\}$ denotes the edge set corresponding to user-item interactions. Using these definitions, the graph-based CF can be abstracted as a prediction function over user-item interactions: $\hat{y}_{i,j} = f(\mathcal{G}; \mathbf{\Theta})$, where $\hat{y}_{i,j}$ is the predicted score for the unknown interaction between user u_i and item p_j, and $\mathbf{\Theta}$ represents the model parameters.

GNN-enhanced CF Models. Graph neural networks (GNNs) [9, 24] have become effective components for modeling user-item relationships in recommender systems. Typically, a GNN encoder is applied to generate user/item embeddings based on recursive message passing operations on the generated graph structures from user-item interactions [22, 31]. Earlier efforts adopt graph convolutional networks to map the interaction graph into latent embeddings, such as NGCF [19] and Star-GCN [38]. To simplify the graph message passing algorithm, recent approaches such as LightGCN [5] and GCCF [2] propose removing the non-linear transformation and activation for embedding propagation. Additionally, inspired by disentangled representation learning, latent intent disentanglement has been used to enhance graph neural networks for fine-grained user preference modeling, as demonstrated in DGCF [20] and DisenHAN [21]. Hyperbolic representation space has been introduced to improve graph collaborative filtering for user embedding, as shown in HGCF [17]. Recent studies have also aimed to capture interaction heterogeneity for graph collaborative filtering, with approaches such as MBGCN [8] and MBGMN [29] designed to enable multiplex GNNs for learning multi-behavior interaction patterns.

Self-Supervised Learning for Recommendation. Recently, data augmentation with self-supervised learning (SSL) has emerged as a promising approach for mitigating the label scarcity and noise issue in recommender systems [30, 32]. One important SSL paradigm is contrastive learning-based augmentation, where semantic-relevant instances are aligned with sampled positive pairs, while unrelated samples as negative pairs are pushed away. For example, random corruptions are performed on graph structures in SGL [25] and node embeddings in SLRec [32]. In addition, pre-defined embedding alignment methods are used to create views for embedding contrasting with heuristics, such as hypergraph construction in HCCF [28] and user clustering in NCL [13].

3 METHODOLOGY

3.1 Graph Invariant Rationale Learning

To eliminate the impact of noisy features and enhance model interpretability, representation learning with rationalization has been explored to identify a subset of important features (*e.g.*, language

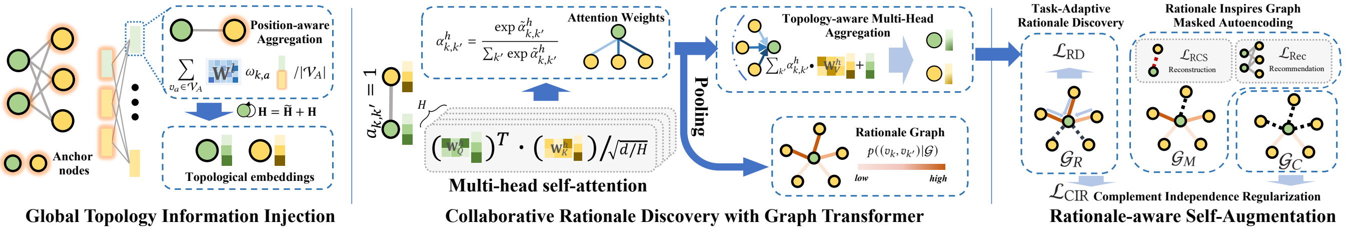

Position-aware Aggregation
$\sum_{v_a \in \mathcal{V}_A}$ $\mathbf{W}^l$ $\omega_{k,a}$ $/|\mathcal{V}_A|$
$\mathbf{H} = \tilde{\mathbf{H}} + \mathbf{H}$
Anchor nodes
Topological embeddings
Global Topology Information Injection
$\alpha^h_{k,k'} = \frac{\exp \tilde{\alpha}^h_{k,k'}}{\sum_{k'} \exp \tilde{\alpha}^h_{k,k'}}$
Attention Weights
$a_{k,k'} = 1$
H
$(\mathbf{W}^h_Q)^T \cdot (\mathbf{W}^h_K)/\sqrt{d/H}$
Multi-head self-attention
Pooling
Topology-aware Multi-Head Aggregation
$\sum_{k'} \alpha^h_{k,k'} \cdot \mathbf{W}^h_V +$
Rationale Graph
$p((v_k, v_{k'})|\mathcal{G})$
low
high
Collaborative Rationale Discovery with Graph Transformer
Task-Adaptive Rationale Discovery
Rationale Inspires Graph Masked Autoencoding
$\mathcal{L}_{\text{RD}}$
$\mathcal{L}_{\text{RCS}}$
Reconstruction
$\mathcal{L}_{\text{Rec}}$
Recommendation
$\mathcal{G}_R$
$\mathcal{G}_M$
$\mathcal{G}_C$
$\mathcal{L}_{\text{CIR}}$ Complement Independence Regularization
Rationale-aware Self-Augmentation

Figure 2: Overall framework illustration of the proposed GFormer model. i) The collaborative rationale discovery is built upon the topology-aware graph transformer for interaction rationalization. ii) Position-aware message passing is enabled to encode pairwise user-item dependency with the global graph context enhancement. iii) Graph autoencoder aims at reconstructing the discovered collaborative rationales, with informative user-item interaction patterns for augmentation. iv) Task-adaptive self-supervision is realized with the awareness of main optimized objective derived from the target recommendation task.

words [36], image pixels [37]) that guide the model prediction results. Recently, rationalization learning techniques have been introduced into graph representation learning by discovering invariant rationales for important graph structural information to benefit downstream graph mining tasks [11, 12, 26]. In our graph-based CF scenario, our invariant rationale discovery scheme is designed to find a subset of graph structures that best guide the self-supervised augmentation for the downstream recommendation task with rationalization. Our invariant rationale discovery with graph collaborative relationships aims to optimize the following objective from two viewpoints: *performance sufficiency* and *complement independence*. This objective is formally given as:

$$\min D\left(f(R(\mathcal{G})), f(\mathcal{G})\right) + I\left(R(\mathcal{G}), C(R(\mathcal{G}))\right) \tag{1}$$

$f(\cdot)$ denotes the predictive function, while $R(\cdot)$ and $C(\cdot)$ denote the rationale and complement of the rationale for the input graph $\mathcal{G}$, respectively. Specifically, for achieving performance sufficiency, the first term aims to minimize the performance difference between using the rationale $R(\mathcal{G})$ and the entire graph $\mathcal{G}$. Here, $D(\cdot)$ represents the difference measurement function. By doing so, important structural information of graph collaborative relations is well-preserved in our learned rationale $R(\mathcal{G})$. Additionally, in pursuit of complement independence by mitigating noisy signals, the second term seeks to minimize the dependency of the complement graph structures $C(R(\mathcal{G}))$ and rationale $R(\mathcal{G})$. With this objective, the complement of our discovered rationales has little influence on the label prediction. Hence, our graph rationale discovery can exploit the invariant relationships between users and items while alleviating the noisy effects of spurious interactions.

3.1.1 **Graph Collaborative Rationale Discovery.** To enable noise-resistant self-supervision for augmentations, our GFormer aims at automatically distilling important graph structures over the interaction graph $\mathcal{G}$, *i.e.*, the collaborative rationales. To generate the informative interaction subgraph structure, our collaborative rationale discovery is designed to estimate the following probability of subgraph $\mathcal{G}_R$ being the rationale of interaction graph $\mathcal{G}$:

$$\begin{aligned} &p(R(\mathcal{G}) = \mathcal{G}_R) = \prod_{e \in \mathcal{E}_R} p(e|\mathcal{G}) \prod_{e\prime \in \mathcal{E}_C} (1 - p(e_{k'}|\mathcal{G})) \\ &\mathcal{G}_R = \{\mathcal{V}, \mathcal{E}_R\}, \quad \mathcal{G}_R \sim p(R(\mathcal{G}) = \mathcal{G}_R), \quad |\mathcal{E}_R| = \rho_R \cdot |\mathcal{E}| \\ &\mathcal{G}_C = \{\mathcal{V}, \mathcal{E}_C\}, \quad \mathcal{E}_C = \{e\prime | e\prime \in \mathcal{E}, e\prime \notin \mathcal{E}_R\} \end{aligned} \tag{2}$$

where ρ_R denotes the proportion of interaction edges selected as the collaborative rationales $\mathcal{G}_R$, and $\mathcal{G}_C$ is defined as the subgraph containing the edges that are not part of $\mathcal{G}_R$. Here, we define e and $e\prime$ to denote user-item interactions in the rationale and complement subgraphs, respectively. To estimate the distribution probability described above, our GFormer proposes to infer the probability of individual edge $p(e|\mathcal{G})$ and $p(e\prime|\mathcal{G})$ being identified as part of the rationale. With a graph encoder for node embeddings, the parameterized rationale generator is formally generalized as follows:

$$p(e|\mathcal{G}) \leftarrow \mathbf{GT}\left(\mathcal{G}, \mathbf{TE}(\mathbf{H}; \mathbf{\Theta}_{\mathbf{TE}}); \mathbf{\Theta}_{\mathbf{GT}}\right); \quad \arg\max_{\mathbf{\Theta}_{\mathbf{TE}}, \mathbf{\Theta}_{\mathbf{GT}}} \mathcal{L}_{\text{RD}} \tag{3}$$

Inspired by the design of dependency rationalization of self-attention in Transformer, our graph encoder $\mathbf{GT}(\cdot)$ is built upon a Graph Transformer architecture, which will be elaborated on in Section 3.1.3. To inject the global topological context into the invariant rationale discovery process, we design the graph topology embedding module $\mathbf{TE}(\cdot)$ to capture the collaborative effects across the entire graph, as detailed in Section 3.1.2. Specifically, $\mathbf{H} \in \mathbb{R}^{(I+J)\times d}$ represents the embedding table containing the representations of all I user nodes and J item nodes. The learnable parameters of the embedding functions $\mathbf{GT}(\cdot)$ and $\mathbf{TE}(\cdot)$ are denoted by $\mathbf{\Theta}_{\mathbf{GT}}$ and $\mathbf{\Theta}_{\mathbf{TE}}$, respectively. In our learning process, $\mathbf{\Theta}_{\mathbf{GT}}$ and $\mathbf{\Theta}_{\mathbf{TE}}$ are inferred by optimizing the BPR-based objective function $\mathcal{L}_{RD}$ for the collaborative filtering task. This enables task-adaptive rationale discovery for SSL augmentation in our GFormer model.

3.1.2 **Global Topology Information Injection.** Motivated by the power of position-aware graph neural networks [34] in capturing global relational information, our GFormer proposes to enhance collaborative rationale discovery by preserving high-order user/item dependencies. We begin by sampling a set of anchor nodes $\mathcal{V}_A \subset \mathcal{V}$ from the user-item interaction graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$. To represent the global topological embeddings of users and items based on their connectivities with anchor nodes, we calculate the distance $d_{k,a}$ between the target node v_k and each anchor node v_a, where distance is defined as the minimum number of edges that must be traversed to go from v_k to v_a in $\mathcal{G}$. Given the calculated distances, we derive the correlation weight $\omega_{k,a}$ for each pair of target-anchor nodes (k, a) as follows:

$$\omega_{k,a} = \begin{cases} \frac{1}{d_{k,a}+1} & if \quad d_{k,a} \leqslant q \\ 0 & otherwise \end{cases} \tag{4}$$

q represents the maximum value allowed for the correlation weight between any target and anchor node, which is used for normalization purposes. The node correlation weights are then normalized to the range of $[0, 1]$. With the weight $\omega_{k,a}$, we refine the target node embedding by considering the correlation weights between the target node k and each anchor node $v_a \in \mathcal{V}_A$:

$$\tilde{\mathbf{h}}_k^l = \sum_{v_a \in \mathcal{V}_A} \mathbf{W}^l \cdot \omega_{k,a} \cdot [\tilde{\mathbf{h}}_k^{l-1} || \tilde{\mathbf{h}}_a^{l-1}] / |\mathcal{V}_A| \tag{5}$$

Here, $\tilde{\mathbf{h}}_k^l, \tilde{\mathbf{h}}_k^{l-1} \in \mathbb{R}^d$ denote the embeddings of node v_k in the l-th and $(l-1)$-th graph propagation layer, respectively. To represent the global topological context based on anchor nodes, we define $\mathbf{H}^{l_T}$ to denote the global topological embedding matrix in the l-th layer. $\mathbf{W}^l \in \mathbb{R}^{d \times 2d}$ is a learnable transformation matrix. $[\cdot||\cdot]$ denotes vector concatenation. After L graph information propagation steps, the embeddings in $\tilde{\mathbf{H}}^L$ preserve the high-order topological information. We then inject this information into the id-corresponding embeddings to obtain the topological embeddings, as follows:

$$\bar{\mathbf{H}} = \mathbf{TE}(\mathbf{H}; \{\mathbf{W}_T^{l_T}\}) \tag{6}$$

In this way, our parameterized rationale generator can capture global collaborative relationships and identify informative patterns of interactions between users and items for SSL augmentation.

3.1.3 **Rationale Discovery with Graph Transformer.** Our rationale discovery aims to extract informative patterns of user-item interactions that can be used for self-supervised augmentation in changing recommendation environments with limited supervision labels. To address this challenge, we draw inspiration from the Transformer architecture and its core self-attention mechanisms. Specifically, we propose a novel approach to learning environment-invariant user preference information as generative self-supervision signals with selective augmentation. This design allows our GFormer to mitigate the noise induced by observational behavior data, which is prone to contain biases and confounding factors that can negatively affect recommendation performance.

Our parameterized rationale discovery module is built over graph transformer framework to encode implicit label-invariant node-wise relations as selected rationales. To incorporate the positional information of user and item nodes into the topology learning process, we feed the global topology-aware node embeddings $\bar{\mathbf{H}}$ into the multi-head self-attention mechanism for rationalization. Specifically, we learn the correlation between node v_k and $v_{k'}$ with respect to the h-th attention head as follows:

$$\alpha_{k,k'}^h = \frac{\exp \tilde{\alpha}_{k,k'}^h}{\sum_{k'} \exp \tilde{\alpha}_{k,k'}^h}; \quad \tilde{\alpha}_{k,k'}^h = \frac{(\mathbf{W}_Q^h \cdot \bar{\mathbf{h}}_k)^\top \cdot (\mathbf{W}_K^h \cdot \bar{\mathbf{h}}_{k'})}{\sqrt{d/H}} \tag{7}$$

Here, $\mathbf{W}_Q^h$ and $\mathbf{W}_K^h \in \mathbb{R}^{\frac{d}{H} \times d}$ represent the transformations used to obtain the query and key embeddings for calculating the attention scores. Since the attention scores encoded by our graph transformer capture the strength of node-wise dependencies, we aggregate the multi-head scores to obtain the probability scores, $p((v_k, v_{k'})|\mathcal{G})$, of graph edges, such as v_k–$v_{k'}$, being selected as rationales. These rationales correspond to the subset of important user-item interaction patterns that best illuminate the user preference learning process with invariant representations, which is presented as:

$$p((v_k, v_{k'})|\mathcal{G}) = \frac{\bar{\alpha}_{k,k'}}{\sum_{(v_k,v_{k'}) \in \mathcal{E}} \bar{\alpha}_{k,k'}}; \quad \bar{\alpha}_{k,k'} = \sum_{h=1}^{H} \alpha_{k,k'}^h / H \tag{8}$$

To sample a rationale estimated by our topology-aware graph transformer, we individually sample $\rho_R \cdot |\mathcal{E}|$ edges from the edge set $\mathcal{E}$ according to their probability scores, $p((v_k, v_{k'})|\mathcal{G})$. Here, the hyperparameter $\rho_R \in \mathbb{R}$ controls the size of the subset of important edges that are selected for rationalization.

3.1.4 **Task-Adaptive Rationale Discovery.** To perform task-level adaptation in our rationale discovery, our GFormer is a task-adaptive rationale discovery paradigm that can perform task-specific rationalization to provide customized recommendations. Specifically, our model leverages the embeddings and the distilled rationales from the graph transformer to generate predictions for users' preferences over items. This process is formally given as follows:

$$\bar{y}_{i,j} = \mathbf{z}_i^{L\top} \cdot \mathbf{z}_j^L; \quad \mathbf{z}_k^L = \sum_{(v_k,v_{k'}) \in \mathcal{E}_R} \beta_{k,k'} \cdot \mathbf{z}_{k'}^{l-1};$$

$$\mathbf{Z}^0 = \mathbf{GT}(\mathcal{G}, \mathbf{TE}(\mathbf{H})) = \Big\Vert_{h=1}^{H} \sum_{k'} \alpha_{k,k'}^h \mathbf{W}_V^h \bar{\mathbf{h}}_{k'} + \bar{\mathbf{h}}_k \tag{9}$$

$\bar{y}_{i,j} \in \mathbb{R}$ denotes the predicted probability of user u_i adopting item p_j. The embeddings $\mathbf{z}_i^L$ and $\mathbf{z}_j \in \mathbb{R}^d$ are used to make predictions on user-item interactions, while $\mathbf{z}_k^L$ for a vertex v_k is obtained through an L-layer LightGCN [5]. Here $\mathcal{E}_R$ denotes the edge set of the sampled rationale graph $\mathcal{G}_R$. $\beta_{k,k'} = 1/\sqrt{d_k d_{k'}}$ denotes the Lapalacian normalization with degrees d_k and $d_{k'}$ of node v_k and $v_{k'}$. The 0-order embeddings $\mathbf{Z}^0$ are obtained through multi-head embedding aggregation in the topology-aware graph transformer, where H is the number of attention heads, and $\mathbf{W}_V^h \in \mathbb{R}^{d/H \times d}$ denotes the value transformation in the self-attention. We employ a residual connection to also use the topology-aware embeddings $\bar{\mathbf{h}}_{k'}$ as input. With the predicted probability score for each (u_i, y_j) interaction, we apply the following BPR loss to guide the rationale discovery process and optimize the objective of the downstream task:

$$\mathcal{L}_{\text{RD}} = \sum_{(u_i, p_j^+, p_j^-)} -\log \text{sigm}(\bar{y}_{i,j^+} - \bar{y}_{i,j^-}) \tag{10}$$

A pair-wise training triplet is formed by sampling a user and items such that $u_i \in \mathcal{U}$ and $v_j^+, v_j^- \in \mathcal{P}$, and the triplet satisfies $(u_i, p_j^+) \in \mathcal{E}$ and $(u_i, p_j^-) \notin \mathcal{E}$. The sigmoid function is represented by $\text{sigm}(\cdot)$. To incorporate task-specific knowledge, a topology-aware graph transformer is employed. This transformer provides task-aware parameter learning to customize the collaborative rationale discovery for different recommendation scenarios. As a result, the collaborative rationale discovery satisfies the sufficiency principle [36] with the objective of $f(R(\mathcal{G})) = f(\mathcal{G})$.

3.2 Rationale-Aware Self-Augmentation

3.2.1 **Rationales Inspire Graph Masked Autoencoding.** Our proposed self-distillation paradigm for discovering collaborative rationales involves performing self-augmentation over the distilled informative user-item interaction patterns through graph masked autoencoding. To achieve this, we configure our GFormer with the rationale-aware mask autoencoder, which masks identified rationables from the interaction graph for autoencoding-based reconstruction. To sample the masked graph structure $\mathcal{G}_M = \{\mathcal{V}, \mathcal{E}_M\}$, we use the reciprocal of the rationale scores. This allows us to mask the most important rationale structures, as shown below:

$$\mathcal{E}_M \sim p_M(\mathcal{E}_M|\mathcal{G}) = \prod_{(v_k,v_{k'})\in\mathcal{E}_M} \alpha^M_{k,k'} \prod_{(v_k,v_{k'})\in\mathcal{E}\setminus\mathcal{E}_M} \alpha^M_{k,k'}$$

$$|\mathcal{E}_M| = \rho_M|\mathcal{E}|;\ \ \alpha^M_{k,k'} = \frac{\bar{\alpha}^M_{k,k'}}{\sum\limits_{(v_k,v_{k'})\in\mathcal{E}} \bar{\alpha}^M_{k,k'}};\ \ \bar{\alpha}^M_{k,k'} = \frac{1}{\bar{\alpha}_{k,k'} + \epsilon} \tag{11}$$

$p_M(\cdot)$ is the probability of sampling edges in the masked graph. $\alpha^M_{k,k'}$ is the probability of selecting an edge between nodes v_k and $v_{k'}$ in the mask generator. $\bar{\alpha}^M_{k,k'}$ is the un-normalized attention score calculated using the reciprocal of the weights $\alpha_{k,k'}$. A small value ϵ is added to avoid a zero denominator. The masked graph has a higher edge density than the rationale graph to only remove the most important rationale edges for noise-resistant autoencoding. The masked graph $\mathcal{G}_M$ with edge set $\mathcal{E}_M$ is then used as input for the autoencoder network, which is presented as follows:

$$\mathbf{S} = \mathbf{GT}(\mathcal{G}_M, \mathbf{TE}(\bar{\mathbf{S}}^L));\quad \bar{\mathbf{s}}^l = \sum_{(v_k,v_{k'})\in\mathcal{E}_M} \beta_{k,k'} \cdot \bar{\mathbf{s}}^{l_1}_{k'} \tag{12}$$

$\mathbf{S} \in \mathbb{R}^{(I+J)\times d}$ represents the final embeddings in the autoencoder. $\mathbf{GT}(\cdot)$ and $\mathbf{TE}(\cdot)$ denote our graph transformer network and the topological information encoder, respectively. We enhance the initial embeddings with L-order local node embeddings $\bar{\mathbf{S}}^L$, encoded from LightGCN. $\bar{\mathbf{S}}^0$ is initialized with the id-corresponding embeddings $\mathbf{H}$. The embeddings $\mathbf{S}$ are used for training the reconstruction of the masked user-item interactions. This can be expressed as:

$$\mathcal{L}_{\text{MAE}} = \sum_{(v_k,v_{k'})\in\mathcal{E}\setminus\mathcal{E}_M} -\tilde{y}_{k,k'};\quad \tilde{y}_{k,k'} = \mathbf{s}_k^\top \mathbf{s}_{k'} \tag{13}$$

$\mathcal{L}_{\text{MAE}}$ is the training objective for reconstructing the masked interaction patterns. $\tilde{y}_{k,k'}$ represents the predicted scores for edge $(v_k, v_{k'})$ on graph $\mathcal{G}$. Inspired by our collaborative rationale discovery, our graph masked autoencoder is trained to reconstruct important interaction patterns that are adaptable to downstream recommendation tasks. Our rationale-aware augmentation approach prevents our generative SSL from being influenced by noisy edges.

3.2.2 **Complement Independence Modeling.** To achieve complement independence in rationale discovery (as discussed in Section 3.1), we introduce a learning component to encourage independence between the distilled collaborative rationales and their corresponding complements, thereby reducing information redundancy. This is done through contrastive regularization, where we minimize the mutual information between the rationale graph $\mathcal{G}_R$ and a sampled complement graph $\mathcal{G}_C$. The complement graph is sampled in a manner similar to graph masking, but with a different sampling rate $\rho_C << \rho_M$ to identify noisy edges. The complement graph $\mathcal{G}_C = \{\mathcal{V}, \mathcal{E}_C\}$ is generated as follows:

$$\mathcal{E}_C \sim p_C(\mathcal{E}_C|\mathcal{G}) = p_M(\mathcal{E}_C|\mathcal{G});\quad |\mathcal{E}_C| = \rho_C \cdot |\mathcal{E}| \tag{14}$$

To ensure that the complement graph $\mathcal{G}_C$ does not contain non-noise edges that could affect the independence regularization, we use a low sampling rate ρ_C. We then apply the following loss to minimize the similarities between the rationale graph $\mathcal{G}_R$ and the complement graph $\mathcal{G}_C$ in high-order representations:

$$\mathcal{L}_{\text{CIR}} = \log \sum_{v_k\in\mathcal{V}} \exp\cos(\mathbf{e}_k^R, \mathbf{e}_k^C)/\tau$$

$$\mathbf{E}^R = \mathbf{L\text{-}GCN}^L(\mathbf{H}, \mathcal{G}_R);\quad \mathbf{E}^C = \mathbf{L\text{-}GCN}^L(\mathbf{H}, \mathcal{G}_C) \tag{15}$$

$\mathbf{L\text{-}GCN}^L(\cdot)$ represents the stacking of L graph layers in LightGCN [5] for recursively passing messages over the input graph ($\mathcal{G}_R$ and $\mathcal{G}_C$). τ is the hyperparameter for the temperature coefficient. The contrastive independence regularization $\mathcal{L}_{\text{CIR}}$ pushes the embeddings of the rationale embeddings $\mathbf{e}_k^R$ and the complement embeddings $\mathbf{e}_k^C$ away from each other for all nodes. This enhances the model's ability to encourage independence between the discovered rationales and complements, thereby improving the noise mitigation ability in our GFormer for SSL-based augmentation.

Table 1: Statistics of the experimental datasets.

Dataset	#Users	#Items	#Interactions	Density
Yelp	42,712	26,822	182,357	$1.6e^{-4}$
Ifashion	31,668	38,048	618,629	$5.1e^{-4}$
LastFM	1,889	15,376	51,987	$1.8e^{-3}$

3.2.3 **SSL-Augmented Model Optimization.** During the training phase, we use the embeddings $\mathbf{S} \in \mathbb{R}^{(I+J)\times d}$ to make predictions for training the recommender. The following point-wise loss function is minimized for model training:

$$\mathcal{L}_{\text{Rec}} = \sum_{a_{i,j}=1} -\log \frac{\exp \mathbf{s}_i^\top \mathbf{s}_j}{\sum_{p_{j'}\in\mathcal{P}} \exp \mathbf{s}_i^\top \mathbf{s}_{j'}} \tag{16}$$

GFormer maximizes the predictions for all positive user-item interactions and minimizes the predictions for all negative user-item interactions as a contrast. During the testing phase, we replace the masked graph $\mathcal{G}_M$ in GFormer with the observed interaction graph $\mathcal{G}$ and predict the relations between user u_i and item p_j by $\hat{y}_{i,j} = \mathbf{s}_i^\top \mathbf{s}_j$. By combining the multiple training objectives, our GFormer is optimized to minimize the following overall objective:

$$\mathcal{L} = \mathcal{L}_{\text{Rec}} + \mathcal{L}_{\text{RCS}} + \lambda_1 \cdot \mathcal{L}_{\text{RD}} + \lambda_2 \cdot \mathcal{L}_{\text{CIR}} + \lambda_3 \cdot \|\Theta\|_{\text{F}}^2 \tag{17}$$

λ_1, λ_2, and λ_3 are hyperparameters used for loss balancing. The last term is the Frobenius-norm regularization for the parameters.

3.3 Discussion of Time and Space Complexity

Time Complexity. Our GFormer employs graph transformer for collaborative rationale discovery and LightGCN as our graph encoder for rationale subgraph structures. The former takes $O(L_T \times (I+J) \times d^2)$ complexity for embedding transformation and $O(L_T \times (\rho_R + \rho_M) \times |\mathcal{E}| \times d)$ for information propagation and aggregation. LightGCN requires $O((L_C\rho_R + L_A\rho_M + L_I\rho_R) \times |\mathcal{E}| \times d)$ cost.

Space Complexity. Our collaborative rationale discovery module is built directly upon the graph encoder–Graph Transformer, which means that no additional rationale learning parameters are needed compared to other rationale graph structure learning methods (*e.g.* [12, 14]). As a result, our GFormer model requires a smaller space cost (i.e., $O((I+J) \times d + d^2)$) than these methods.

Table 2: Performance comparison between our proposed GFormer and all baselines on Ifashion, Yelp, LastFM datasets.

Datasets	Metric	BiasMF	NCF	AutoRec	PinSage	NGCF	GCCF	LightGCN	EGLN	SLRec	NCL	HCCF	SGL	GFormer	p-val.
Yelp	Recall@10	0.0122	0.0166	0.0230	0.0278	0.0438	0.0484	0.0422	0.0458	0.0418	0.0493	0.0518	0.0522	**0.0562**	2.8e-6
	NDCG@10	0.0070	0.0101	0.0133	0.0171	0.0269	0.0296	0.0254	0.0278	0.0258	0.0301	0.0318	0.0319	**0.0350**	9.8e-9
	Recall@20	0.0198	0.0292	0.0410	0.0454	0.0678	0.0754	0.0761	0.0726	0.0650	0.0806	0.0789	0.0815	**0.0878**	5.2e-8
	NDCG@20	0.0090	0.0138	0.0186	0.0224	0.0340	0.0378	0.0373	0.0360	0.0327	0.0402	0.0391	0.0410	**0.0442**	1.6e-6
	Recall@40	0.0303	0.0442	0.0678	0.0712	0.1047	0.1163	0.1031	0.1121	0.1026	0.1192	0.1244	0.1249	**0.1328**	8.3e-11
	NDCG@40	0.0117	0.0167	0.0253	0.0287	0.0430	0.0475	0.0413	0.0456	0.0418	0.0485	0.0510	0.0517	**0.0551**	7.8e-9
Ifashion	Recall@10	0.0302	0.0268	0.0309	0.0291	0.0375	0.0373	0.0437	0.0473	0.0373	0.0474	0.0489	0.0512	**0.0542**	6.3e-7
	NDCG@10	0.0281	0.0253	0.0264	0.0276	0.0350	0.0352	0.0416	0.0438	0.0353	0.0446	0.0464	0.0487	**0.0520**	2.0e-6
	Recall@20	0.0523	0.0451	0.0537	0.0505	0.0636	0.0639	0.0751	0.0787	0.0633	0.0797	0.0815	0.0845	**0.0894**	1.5e-7
	NDCG@20	0.0360	0.0306	0.0351	0.0352	0.0442	0.0445	0.0528	0.0549	0.0444	0.0558	0.0578	0.0603	**0.0635**	6.3e-5
	Recall@40	0.0858	0.0785	0.0921	0.0851	0.1062	0.1047	0.1207	0.1277	0.1043	0.1283	0.1306	0.1354	**0.1424**	9.0e-9
	NDCG@40	0.0474	0.0423	0.0483	0.0470	0.0585	0.0584	0.0677	0.0715	0.0582	0.0723	0.0744	0.0773	**0.0818**	9.9e-8
LastFM	Recall@10	0.0609	0.0574	0.0543	0.0899	0.1257	0.1230	0.1490	0.1133	0.1175	0.1491	0.1502	0.1496	**0.1573**	5.8e-7
	NDCG@10	0.0696	0.0645	0.0599	0.1046	0.1489	0.1452	0.1739	0.1263	0.1384	0.1745	0.1773	0.1775	**0.1831**	1.8e-6
	Recall@20	0.0980	0.0956	0.0887	0.1343	0.1918	0.1816	0.2188	0.1823	0.1747	0.2196	0.2210	0.2236	**0.2352**	5.0e-8
	NDCG@20	0.0860	0.0800	0.0769	0.1229	0.1759	0.1681	0.2018	0.1557	0.1613	0.2021	0.2047	0.2070	**0.2145**	1.7e-8
	Recall@40	0.1450	0.1439	0.1550	0.1990	0.2794	0.2649	0.3156	0.2747	0.2533	0.3130	0.3184	0.3194	**0.3300**	4.3e-7
	NDCG@40	0.1067	0.1055	0.1031	0.1515	0.2146	0.2049	0.2444	0.1966	0.1960	0.2437	0.2458	0.2498	**0.2567**	4.2e-7

4 EVALUATION

In this section, we conduct extensive experiments for model evaluation to answer the following key research questions:

- **RQ1**: How effective is our GFormer compared to various state-of-the-art (SOTA) recommendation models?
- **RQ2**: How does the model performance change if we substitute key modules of GFormer with different naive implementations?
- **RQ3**: How does our rationale-aware graph transformer perform against data noise and data scarcity issues?
- **RQ4**: What is the training efficiency of the proposed GFormer?
- **RQ5**: How do key parameters affect the model performance?
- **RQ6**: How does our collaborative rationale discovery paradigm realize the interpretability of user-item interaction patterns?

4.1 Experimental Setup

4.1.1 **Datasets.** We conduct experiments on three widely-used real-world datasets for evaluating recommender systems: Yelp, Ifashion, and LastFM. The Yelp dataset is used for recommending businesses venues to users, and it is collected from the well-known Yelp platform. Ifashion is a fashion outfit dataset collected by Alibaba, while LastFM is a dataset that tracks user interaction activities in music applications and internet radio sites. Table 1 summarizes the statistics of the three experimental datasets.

4.1.2 **Evaluation Protocols.** We split the observed interactions of each dataset into training set, validation set, and test set using a ratio of $0.70 : 0.05 : 0.25$. To measure the recommendation accuracy for each user over the whole item set, we adopted the all-rank protocol, following [25, 28]. This protocol helps alleviate the evaluation bias introduced by negative sampling. We evaluate all models using two representative metrics: Recall Ratio (*Recall@K*) and Normalized Discounted Cumulative Gain (*NDCG@K*), with $K = 10, 20, 40$.

4.1.3 **Baseline Methods.** To comprehensively study the performance of GFormer, we compare it with many baseline methods covering various techniques for collaborative filtering.

Non-GNN Collaborative Filtering Approaches. We first include several conventional CF methods as benchmarks for comparison.

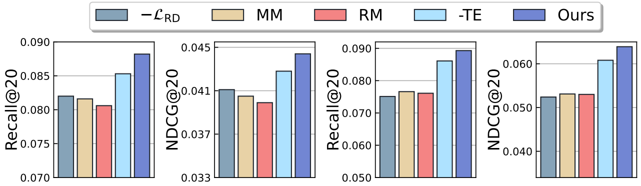
- **BiasMF** [10]: This is a method based on matrix factorization which maps users and items to vector representations in the latent space and takes their bias score into consideration.
- **NCF** [6]: This method uses neural networks with multiple layers to encode non-linear features of user-item interactions.
- **AutoRec** [15]: This model adopts the autoencoder structure and learns embeddings through reconstructing observed interactions.

GNN-based Recommendation Methods without SSL. Graph neural networks (GNNs) have shown their effectiveness in injecting high-order collaborative signals into user and item embeddings. For this research line, we compared our GFormer with representative GNN-enhanced recommendation models that use various message passing schemes for performance evaluation.

- **PinSage** [33]: This model leverages graph convolutional networks with random-walk-based message passing to encode the user-item interaction graph with high efficiency.
- **NGCF** [19]: This model captures high-order collaborative information through multiple layers of graph neural networks.
- **LightGCN** [5]: This approach simplifies the architecture of NGCF and employs a light-weighted convolutional graph encoder for better representation learning and model training.
- **GCCF** [2]: This model also introduces several improvements to GCN-based CF methods, including omitting the non-linear transformation and applying residual connections.

SSL-enhanced Recommendation Models. For comprehensive model evaluation, we included many recent SSL-enhanced recommender systems as baselines. In these models, different augmentation strategies are designed to provide self-supervision signals.

- **EGLN** [31]: This model incorporates a node embedding learning module and a graph structure learning module, and encourages them to learn from each other for better representations.



(a) Yelp Data (b) Ifashion Data

Figure 3: Performance of ablated models on Yelp and Ifashion datasets in terms of *Recall@20* and *NDCG@20*.

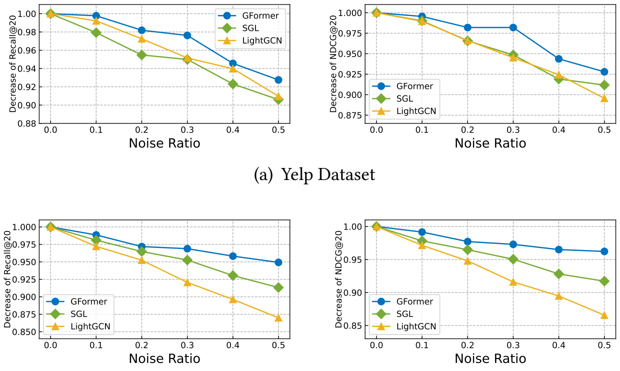
- **SLRec** [32]: This approach conducts contrastive learning between node features to regularize the recommendation learning.
- **NCL** [13]: This model first uses an EM algorithm to perform clustering over users, and then conducts neighborhood-enriched contrastive learning within each cluster.
- **SGL** [25]: This model uses random data augmentation operators (*e.g.*, edge dropping, node dropping, and random walks) to construct views over interaction structures for contrastive learning.
- **HCCF** [28]: This is a state-of-the-art model that conducts contrastive learning through constructing hypergraph-based global and local views for modeling global relations.

4.1.4 **Hyperparameter Settings.** We implement our GFormer using PyTorch. We use the Adam optimizer for parameter learning with a learning rate of $1e^{-3}$ and no learning rate decay. For model hyperparameters of GFormer, we set the embedding size to $d = 32$ by default, the size of the anchor node set to $|\mathcal{V}_A| = 32$, and tuned the graph rationale keep rate ρ_R in $\{0.5, 0.6, 0.7, 0.8, 0.9\}$. For coefficients of different loss terms, we search for λ_1 in $\{0.5, 1, 2, 4, 8\}$, λ_2 in the range $\{1, 1e^{-1}, 1e^{-2}, 1e^{-3}, 1e^{-4}\}$, and λ_3 in the range $\{1e^{-3}, 1e^{-4}, 1e^{-5}, 1e^{-6}, 1e^{-7}, 1e^{-8}\}$ respectively. We chose the number of graph transformer layers in the range of $\{1, 2, 3, 4, 5, 6\}$, and the number of graph convolutional layers in $\{1, 2, 3, 4, 5\}$.

4.2 Overall Performance Comparison (RQ1)

We report the overall performance of GFormer and all compared baselines in terms of *Recall@K* and *NDCG@K* under top-10, top-20, and top-40 settings in Table 2. To validate the superiority of our model compared to the strongest baselines, we conduct the test of significance where p-val< 0.05 suggests a statistically significant improvement achieved by GFormer. From the experimental results in Table 2, we mainly have the following observations:

- GFormer consistently outperformed all baselines, including the strong SSL-enhanced methods (*e.g.*, SGL, HCCF) by a large margin. We ascribe this superiority to our rationale-aware SSL augmentation, which automatically derived informative self-supervision signals from the learned collaborative rationale. In contrast, models with stochastic SSL augmentations (*e.g.*, Dropout in SGL) performed much worse due to the possible loss of important graph structural information of sparse users and long-tail items in their randomized contrastive views. Moreover, although HCCF and NCL both adopted carefully-designed hand-crafted CL tasks, they may not be able to provide accurate self-supervision signals (*e.g.*, hard augmented instance) compared to GFormer. In the face of interaction noise, their pretext tasks are easily misguided by the noisy information contained in the augmented data. Our GFormer tackled these limitations of existing CL models by introducing rationale-aware self-augmentation via masked autoencoding and thus achieved better performance.
- Despite the disadvantages of existing CL methods mentioned above, we observed that baseline models with SSL augmentation generally performed better than those without (*e.g.*, NGCF, LightGCN). This could be due to the labeled data scarcity problem of the recommendation task, while SSL can mitigate this problem by introducing additional self-supervision signals from limited observed interactions. Also, incorporating SSL can also alleviate the over-fitting issue of GNNs for user representations, which is used in most strong baselines, on such sparse data and help learn better embeddings for the recommendation.



(a) Yelp Dataset

(b) LastFM Dataset

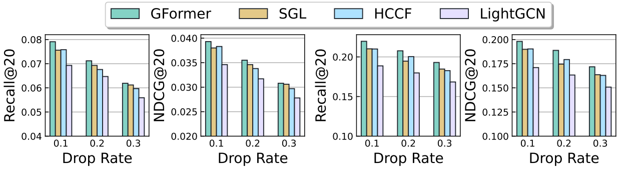
Figure 4: Performance on Yelp and LastFM datasets with noise perturbation in terms of *Recall@20* and *NDCG@20*.

4.3 Ablation Study (RQ2)

To study the effectiveness of the key components of GFormer, we performed an ablation study over several variants, *i.e.*, **-TE**, **RM**, **MM**, and $-\mathcal{L}_{\mathbf{RD}}$. Results in terms of *Recall@20* and *NDCG@20* are plotted in Figure 3, from which we had the following discussions.

4.3.1 **Effect of Global Topology Information Injection.** In the variant **-TE**, we replace the topological embedding with pure id-corresponding embedding to disable topology information injection. The results show that the ablated model has a worse performance on both datasets. This is because the injection of global topology information enables our GFormer to capture high-order collaborative relationships for our graph Transformer to encode informative interaction patterns, so that better graph rationale can be provided for both the recommendation task and the reconstruction task to ultimately boost the overall performance.

4.3.2 **Effect of Rationale-Aware Self-Augmentation.** To study the influence of the key module of GFormer, *i.e.*, the rationale-aware self-augmentation module, we replace it with random masking in variant **RM** and an MLP-based masking strategy in variant **MM**. Specifically, for an edge (u, v), the **MM** variant first feeds the embeddings $\bar{\mathbf{h}}_u, \bar{\mathbf{h}}_v$ into a multi-layer perceptron (MLP) to compute



(a) Yelp Data (b) LastFM Data

Figure 5: Performance on Yelp and LastFM datasets under different sparsity level in terms of *Recall@20* and *NDCG@20*.

the importance score of nodes u, v, and then obtains the mask probability of the edge by dot product of node importance score, *i.e.*, $\text{MLP}(\bar{\mathbf{h}}_u) \cdot \text{MLP}(\bar{\mathbf{h}}_u)$. The results show that both variants have a significant performance drop, suggesting that random masking and trivial adaptive masking through MLPs are both unable to discover important graph structures. Instead, they may corrupt informative structures of the interaction graph and introduce additional noise to the SSL task. In contrast, our GFormer avoids these disadvantages by incorporating meaningful self-supervision signals from the learned graph rationale $R(\mathcal{G})$ and utilizing the complement $C(R(\mathcal{G}))$ for model denoising, resulting in better performance.

4.3.3 **Effect of Adaptive Rationale Learning.** The downstream recommendation loss $\mathcal{L}_{\text{RD}}$ (Eq. 10) plays a crucial role in GFormer by guiding graph invariant rationale learning with adaptive supervision signals. To verify its contribution, we remove $\mathcal{L}_{\text{RD}}$ in the variant $-\mathcal{L}_{\text{RD}}$, which leads to severe performance degradation. This is because $\mathcal{L}_{\text{RD}}$ allows our GFormer to discover relevant graph rationale that captures graph structures specifically useful for the recommendation task. Different SSL modules in GFormer can then be optimized in a better-aligned way. Furthermore, we observe that the degradation caused by removing $\mathcal{L}_{\text{RD}}$ is larger on the Ifashion dataset. This may be due to the larger amount and denser interaction data of Ifashion compared to Yelp, which provides the loss $\mathcal{L}_{\text{RD}}$ with more supervision signals and higher importance.

4.4 Model Robustness Study (RQ3)

In this section, we study the robustness of our model against data noise and data scarcity by testing the model performance on manually damaged training data from the corresponding two dimensions, in comparison to representative baseline methods.

4.4.1 **Robustness against Data Noise.** To study the robustness of GFormer against noise perturbation, we randomly inject different proportions (10%, 20%, 30%, 40%, 50%) of edges with artificial noise on the original interaction graph and evaluate the performance of GFormer and representative strong baseline methods on these noisy datasets. As shown by the results illustrated in Figure 4, our GFormer consistently achieves the lowest performance degradation under all noise levels. Under lower levels of noise (*e.g.*, 10% to 40%), we observe that SSL-enhanced methods (SGL) have a degradation ratio similar to LightGCN, suggesting that stochastic SSL augmentation does not significantly improve robustness against data noise. In contrast, our GFormer adopts an automated SSL paradigm that is rationale-aware, making it possible to discover latent informative structures in a noisy dataset for representation.

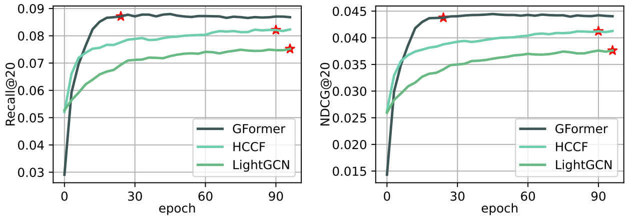


Figure 6: Test results in terms of *Recall@20* and *NDCG@20* *w.r.t* training epochs on Yelp dataset. The stars represent points of convergence. Compared with baselines, the faster convergence rate of our GFormer can be observed.

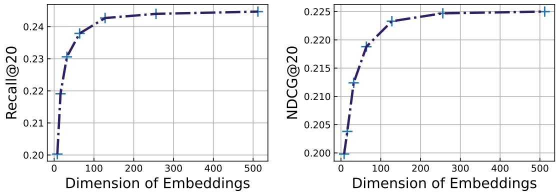


Figure 7: Performance in terms of *Recall@20* and *NDCG@20* under different embedding dimensionality.

4.4.2 **Robustness against Data Sparsity.** We also conduct experiments to evaluate the performance of GFormer on various sparsity levels. Specifically, we drop a certain proportion (10%, 20%, 30%) of interactions in the dataset and run GFormer as well as representative baseline methods on the sparsified datasets. As shown by the results in Figure 5, our GFormer outperforms all other models under all sparsity levels, suggesting that our rationale-enhanced SSL framework enables GFormer to generate more meaningful self-supervision signals than common SSL models on sparse data, thereby increasing the model's robustness against data sparsity. Additionally, we observe that the performance degradation with respect to the drop rate is more significant on the Yelp dataset compared to the LastFM dataset. This may be caused by the relatively higher interaction sparsity of the Yelp dataset, such that dropping more interactions severely hinders effective CF modeling.

4.5 Model Convergence Study (RQ4)

To analyze the training efficiency of our proposed GFormer, we plot the convergence curve (*i.e.*, test metric values with respect to training epochs) of GFormer and three strong baseline methods, including SSL-enhanced methods (*e.g.*, HCCF), in Figure 6. It can be observed from the stars indicating convergence that GFormer only takes fewer than 30 epochs to converge, which is much faster compared to other models. We attribute this superiority to the helpful task-adaptive self-supervision signals derived from our learned graph rationale. Stochastic data augmentation strategies cannot actively discover important graph structures for the recommendation task and may provide misleading self-supervision signals during the training stage, thus the baseline models are naturally optimized in a slower manner. Our proposed SSL augmentation focuses on the recommendation task and adapts fast, enabling GFormer to achieve the best performance in an early stage. The faster performance improvement of SSL-based methods compared to LightGCN is likely caused by the low-temperature contrastive learning.

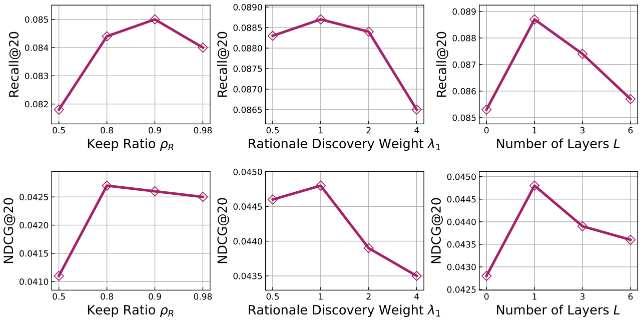
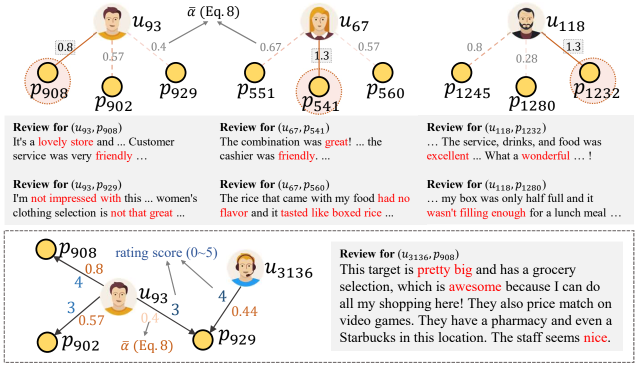


Figure 8: Performance under different hyperparameter settings in terms of *Recall@20* and *NDCG@20* on Yelp dataset.

4.6 Hyperparameter Study (RQ5)

To study the effect of various hyperparameters on the model performance, we present experiment results in terms of *Recall@20* and *NDCG@20* on the LastFM dataset in Figure 7 and 8. The following observations corresponding to different hyperparameters are made:

- **Dimension of latent space** d: We tune the size of user and item embeddings from 8 to 512. As shown in Figure 7, an increase in the embedding dimension leads to significant performance improvement at the beginning. This is because a larger dimensionality allows our topological embedding to capture richer information about the global user-item topology and provide useful representations for other modules in GFormer. However, too large sizes (*e.g.*, $d > 256$) only bring subtle improvement, since this may cause the over-fitting problem of GNNs.
- **Keep ratio** ρ_R: This hyperparameter controls the proportion of interaction edges to be selected and form the graph rationale $\mathcal{G}_R$. Results show that a low keep ratio harms model performance since insufficient collaborative relations are obtained in the graph rationale for representation learning. However, setting the keep ratio close to 1 also leads to a performance drop, because noisy interactions in the original graph cannot be adequately dropped.
- **Collaborative rationale discovery weight** λ_1: This hyperparameter controls the regularization strength of the objective from the downstream recommendation task. $\mathcal{L}_{\text{RD}}$ is used to guide the graph rationale discovery in our masked graph transformer paradigm. From the results, we observe that applying large enough weights greatly improves the model performance due to the benefits brought by the discovery of task-relevant graph rationale for augmentation. However, setting the weight for $\mathcal{L}_{\text{RD}}$ too large may be counterproductive due to the overfitting effect.
- **Number of layers** L **in Global Information Injection**: In our GFormer, the designed global information injection module can achieve competitive results with one iteration. We found in our experiments that global information encoding with too many layers results in serious damage to performance. Such over-smoothed representations may adversely affect the discovery of rationale-aware augmentation with indistinguishable embeddings. Additionally, setting $L = 0$, *i.e.* removing global topology information injection (Section 3.1.2), leads to even worse performance due to insufficient modeling of high-order relations.



Review for (u_{93}, p_{908})
It's a lovely store and ... Customer service was very friendly …

Review for (u_{93}, p_{929})
I'm not impressed with this ... women's clothing selection is not that great ...

Review for (u_{67}, p_{541})
The combination was great! ... the cashier was friendly. ...

Review for (u_{67}, p_{560})
The rice that came with my food had no flavor and it tasted like boxed rice ...

Review for (u_{118}, p_{1232})
… The service, drinks, and food was excellent ... What a wonderful … !

Review for (u_{118}, p_{1280})
… my box was only half full and it wasn't filling enough for a lunch meal …

Review for (u_{3136}, p_{908})
This target is pretty big and has a grocery selection, which is awesome because I can do all my shopping here! They also price match on video games. They have a pharmacy and even a Starbucks in this location. The staff seems nice.

Figure 9: Case study of collaborative rationale discovery for distilling informative knowledge from noisy interactions.

4.7 Case Study (RQ6)

To study the interpretation ability of GFormer, a case study is performed over several representative users sampled from the Yelp dataset. Specifically, we inspect the corresponding reviews and ratings given by a user to their interacted items, which are not used for model training, and see whether they match our correlation score $\bar{\alpha}$ for collaborative rationale discovery. As illustrated in Figure 9, the items with the highest correlation scores for users u_{93}, u_{67}, u_{118} all match the items they gave positive feedback, while edges to items with unsatisfying reviews are generally given lower scores. These results suggest that GFormer can emphasize learning on informative user-item correlations and utilize these interaction data to enhance the model learning through the reconstruction SSL. Furthermore, when we investigate the specific ratings given by u_{93}, we observe that the edge with the highest correlation score (*i.e.*, (u_{93}, p_{908})) corresponds to the user's highest rating. Meanwhile, although u_{3136} gives a higher rating (4) than u_{93} (3) to item p_{929}, the edge (u_{3136}, p_{929}) is given a low correlation score similar to (u_{93}, p_{929}). This could be due to the fact that user u_{3136} is very likely to provide many high ratings (36 out of 47 of all ratings given by u_{3136} are 4 or 5). Our rationale discovery module successfully identifies such bias and adaptively adjusts the correlation score with respect to different user behaviors.

5 CONCLUSION

This paper aims to uncover useful user-item interaction patterns as rationales for augmenting collaborative filtering with the learning of invariant rationales for SSL. Our proposed GFormer model provides guidance to distill semantically informative graph connections with the integration of global topology embedding and task-adaptation. Our work opens avenues for constructing rationale-aware general augmentation through masked graph autoencoding. Our empirical results suggest that SSL-based augmentation with effective rationalization can facilitate user preference learning, and thus significantly boost recommendation performance. While our new model already endows adaptive augmentation with task-aware rationale discovery, it is an interesting open question on how to adapt it to other recommendation scenarios, such as social-aware recommendation and knowledge graph-enhanced recommenders.

REFERENCES

[1] Xuheng Cai, Chao Huang, Lianghao Xia, and Xubin Ren. 2023. LightGCL: Simple Yet Effective Graph Contrastive Learning for Recommendation. In *The Eleventh International Conference on Learning Representations (ICLR).*

[2] Lei Chen, Le Wu, Richang Hong, Kun Zhang, and Meng Wang. 2020. Revisiting graph based collaborative filtering: A linear residual graph convolutional network approach. In *AAAI Conference on Artificial Intelligence (AAAI)*, Vol. 34. 27–34.

[3] Yabo Chen, Yuchen Liu, Dongsheng Jiang, Xiaopeng Zhang, Wenrui Dai, Hongkai Xiong, and Qi Tian. 2022. Sdae: Self-distillated masked autoencoder. In *European Conference on Computer Vision (ECCV).* Springer, 108–124.

[4] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2022. Masked autoencoders are scalable vision learners. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* 16000–16009.

[5] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *International Conference on Research and Development in Information Retrieval (SIGIR).* 639–648.

[6] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *The Web Conference (WWW).* 173–182.

[7] Zhenyu Hou, Xiao Liu, Yukuo Cen, Yuxiao Dong, Hongxia Yang, Chunjie Wang, and Jie Tang. 2022. Graphmae: Self-supervised masked graph autoencoders. In *International Conference on Knowledge Discovery and Data Mining (KDD).*

[8] Bowen Jin, Chen Gao, Xiangnan He, Depeng Jin, and Yong Li. 2020. Multi-behavior recommendation with graph convolutional networks. In *International Conference on Research and Development in Information Retrieval (SIGIR).* 659–668.

[9] Thomas N Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations (ICLR).*

[10] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37.

[11] Haoyang Li, Ziwei Zhang, Xin Wang, and Wenwu Zhu. 2022. Learning invariant graph representations for out-of-distribution generalization. In *Advances in Neural Information Processing Systems (NeurIPS).*

[12] Sihang Li, Xiang Wang, An Zhang, Yingxin Wu, Xiangnan He, and Tat-Seng Chua. 2022. Let invariant rationale discovery inspire graph contrastive learning. In *International Conference on Machine Learning (ICML).* PMLR, 13052–13065.

[13] Zihan Lin, Changxin Tian, Yupeng Hou, and Wayne Xin Zhao. 2022. Improving graph collaborative filtering with neighborhood-enriched contrastive learning. In *The Web Conference (WWW).* 2320–2329.

[14] Dongsheng Luo, Wei Cheng, Wenchao Yu, Bo Zong, Jingchao Ni, Haifeng Chen, and Xiang Zhang. 2021. Learning to drop: Robust graph neural network via topological denoising. In *International Conference on Web Search and Data Mining (WSDM).* 779–787.

[15] Suvash Sedhain, Aditya Krishna Menon, Scott Sanner, and Lexing Xie. 2015. Autorec: Autoencoders meet collaborative filtering. In *Proceedings of the 24th international conference on World Wide Web.* 111–112.

[16] Jie Shuai, Kun Zhang, Le Wu, Peijie Sun, Richang Hong, Meng Wang, and Yong Li. 2022. A review-aware graph contrastive learning framework for recommendation. In *International Conference on Research and Development in Information Retrieval (SIGIR).* 1283–1293.

[17] Jianing Sun, Zhaoyue Cheng, Saba Zuberi, Felipe Pérez, and Maksims Volkovs. 2021. Hgcf: Hyperbolic graph convolution networks for collaborative filtering. In *The Web Conference (WWW).* 593–601.

[18] Changxin Tian, Yuexiang Xie, Yaliang Li, Nan Yang, and Wayne Xin Zhao. 2022. Learning to Denoise Unreliable Interactions for Graph Collaborative Filtering. In *International Conference on Research and Development in Information Retrieval (SIGIR).* 122–132.

[19] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural graph collaborative filtering. In *International Conference on Research and Development in Information Retrieval (SIGIR).* 165–174.

[20] Xiang Wang, Hongye Jin, An Zhang, Xiangnan He, Tong Xu, and Tat-Seng Chua. 2020. Disentangled graph collaborative filtering. In *International Conference on Research and Development in Information Retrieval (SIGIR).* 1001–1010.

[21] Yifan Wang, Suyao Tang, Yuntong Lei, Weiping Song, Sheng Wang, and Ming Zhang. 2020. Disenhan: Disentangled heterogeneous graph attention network for recommendation. In *International Conference on Information & Knowledge Management (CIKM).* 1605–1614.

[22] Zhenyi Wang, Huan Zhao, and Chuan Shi. 2022. Profiling the Design Space for Graph Neural Networks based Collaborative Filtering. In *International Conference on Web Search and Data Mining (WSDM).* 1109–1119.

[23] Wei Wei, Chao Huang, Lianghao Xia, and Chuxu Zhang. 2023. Multi-Modal Self-Supervised Learning for Recommendation. In *The Web Conference (WWW).* 790–800.

[24] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. 2019. Simplifying graph convolutional networks. In *International Conference on Machine Learning (ICML).* PMLR, 6861–6871.

[25] Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. 2021. Self-supervised graph learning for recommendation. In *International Conference on Research and Development in Information Retrieval (SIGIR).* 726–735.

[26] Yingxin Wu, Xiang Wang, An Zhang, Xiangnan He, and Tat-Seng Chua. 2022. Discovering Invariant Rationales for Graph Neural Networks. In *International Conference on Learning Representations (ICLR).*

[27] Lianghao Xia, Chao Huang, Jiao Shi, and Yong Xu. 2023. Graph-less Collaborative Filtering. In *ACM Web Conference (WWW).* 17–27.

[28] Lianghao Xia, Chao Huang, Yong Xu, Jiashu Zhao, Dawei Yin, and Jimmy Huang. 2022. Hypergraph contrastive collaborative filtering. In *International Conference on Research and Development in Information Retrieval (SIGIR).* 70–79.

[29] Lianghao Xia, Yong Xu, Chao Huang, Peng Dai, and Liefeng Bo. 2021. Graph meta network for multi-behavior recommendation. In *International Conference on Research and Development in Information Retrieval.* 757–766.

[30] Yuhao Yang, Chao Huang, Lianghao Xia, Chunzhen Huang, Da Luo, and Kangyi Lin. 2023. Debiased Contrastive Learning for Sequential Recommendation. In *The Web Conference (WWW).* 1063–1073.

[31] Yonghui Yang, Le Wu, Richang Hong, Kun Zhang, and Meng Wang. 2021. Enhanced graph learning for collaborative filtering via mutual information maximization. In *International Conference on Research and Development in Information Retrieval (SIGIR).* 71–80.

[32] Tiansheng Yao, Xinyang Yi, Derek Zhiyuan Cheng, Felix Yu, Ting Chen, Aditya Menon, Lichan Hong, Ed H Chi, Steve Tjoa, Jieqi Kang, et al. 2021. Self-supervised learning for large-scale item recommendations. In *International Conference on Information & Knowledge Management (CIKM).* 4321–4330.

[33] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining.* 974–983.

[34] Jiaxuan You, Rex Ying, and Jure Leskovec. 2019. Position-aware graph neural networks. In *International Conference on Machine Learning (ICML).* PMLR, 7134–7143.

[35] Yuning You, Tianlong Chen, Yang Shen, and Zhangyang Wang. 2021. Graph contrastive learning automated. In *International Conference on Machine Learning (ICML).* PMLR, 12121–12132.

[36] Mo Yu, Shiyu Chang, Yang Zhang, and Tommi S Jaakkola. 2019. Rethinking cooperative rationalization: Introspective extraction and complement control. *International Conference on Empirical Methods in Natural Language Processing (EMNLP).*

[37] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. From recognition to cognition: Visual commonsense reasoning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* 6720–6731.

[38] Jiani Zhang, Xingjian Shi, Shenglin Zhao, and Irwin King. 2019. Star-gcn: Stacked and reconstructed graph convolutional networks for recommender systems. In *AAAI Conference on Artificial Intelligence (AAAI).*

[39] Yang Zhang, Fuli Feng, Xiangnan He, Tianxin Wei, Chonggang Song, Guohui Ling, and Yongdong Zhang. 2021. Causal intervention for leveraging popularity bias in recommendation. In *International Conference on Research and Development in Information Retrieval (SIGIR).* 11–20.

[40] Zhenyu Zhang, Bowen Yu, Xiaobo Shu, Xue Mengge, Tingwen Liu, and Li Guo. 2021. From What to Why: Improving Relation Extraction with Rationale Graph. In *Association for Computational Linguistics (ACL).* 86–95.

[41] Kun Zhou, Hui Wang, Wayne Xin Zhao, Yutao Zhu, Sirui Wang, Fuzheng Zhang, Zhongyuan Wang, and Ji-Rong Wen. 2020. S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization. In *International Conference on Information & Knowledge Management (CIKM).* 1893–1902.

[42] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. 2021. Graph contrastive learning with adaptive augmentation. In *The Web Conference (WWW).* 2069–2080.