Review-Enhanced Hierarchical Contrastive Learning for Recommendation

Ke Wang[1,2], Yanmin Zhu[1*], Tianzi Zang[3], Chunyang Wang[1], Mengyuan Jing[1]

[1]Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China
[2]Hangzhou Innovation Institute, Beihang University, Hangzhou, China
[3]Nanjing University of Aeronautics and Astronautics, Nanjing, China
{onecall,yzhu,wangchy,jingmy}@sjtu.edu.cn, zangtianzi@nuaa.edu.cn

Abstract

Designed to establish potential relations and distill high-order representations, graph-based recommendation systems continue to reveal promising results by jointly modeling ratings and reviews. However, existing studies capture simple review relations, failing to (1) completely explore hidden connections between users (or items), (2) filter out redundant information derived from reviews, and (3) model the behavioral association between rating and review interactions. To address these challenges, we propose a review-enhanced hierarchical contrastive learning, namely ReHCL. First, ReHCL constructs topic and semantic graphs to fully mine review relations from different views. Moreover, a cross-view graph contrastive learning is used to achieve enhancement of node representations and extract useful review knowledge. Meanwhile, we design a neighbor-based positive sampling to capture the graph-structured similarity between topic and semantic views, further performing efficient contrast and reducing redundant noise. Next, we propose a cross-modal contrastive learning to match the rating and review representations, by exploring the association between ratings and reviews. Lastly, these two contrastive learning modes form a hierarchical contrastive learning task, which is applied to enhance the final recommendation task. Extensive experiments verify the superiority of ReHCL compared with state-of-the-arts.

Introduction

Recommender systems have become an indispensable part of e-commerce services (e.g., Amazon and Yelp). They often model rating interactions to capture user preferences and item characteristics (Do et al. 2022; Wang, Cai, and Wang 2022). However, numerical ratings are often sparse, and only modeling single interactions fails to learn enriched representations of understanding users' intents. Therefore, recent efforts (Liu et al. 2019, 2020c) adopt reviews to alleviate the sparsity. Review-based methods absorb additional knowledge to enhance generated representations by modeling free-form textual reviews (Wang et al. 2022).

Recently, Graph-based Recommender Systems (GRS) (Gao et al. 2020; Shuai et al. 2022; Ren et al. 2022) capture potential relations existing in reviews, and leverage graph neural networks to learn node representations. Several works focus on studying topical links (Wang et al. 2023) or semantic similarity (Liu et al. 2020c) between users (or items) derived from reviews, alleviating the sparsity of user-item interactions. Despite exhibiting satisfactory performance, these technics suffer from the following limitations.

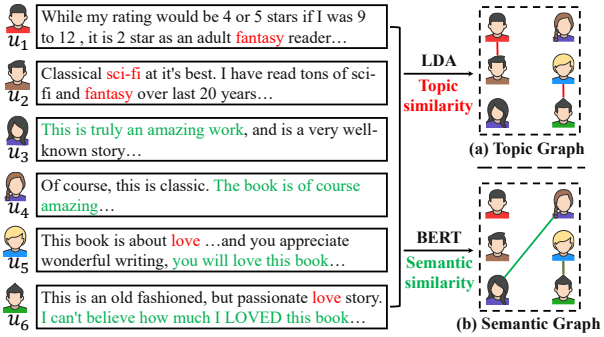


Figure 1: Six review comments written by different users in the *Book* domain. Two types of graphs are constructed from topic and semantic views via LDA and BERT, respectively.

- **Incomplete Relation Extraction**. Existing GRS either extract the key topic factors at word level to discover highly interpretable textual cues, or capture implicit semantics at review level to search for contextual clues between users (or items), while ignoring that user-generated reviews carry both topic and semantic properties (Chin et al. 2018). In fact, the critical topics and whole contexts jointly affect latent relations from different views, especially when users expound local topics and holistic contexts simultaneously.
- **Redundant Review Information**. Although review texts can provide detailed descriptions of users' interests, sometimes they also contain too much redundant or repeated information, thus introducing unnecessary noise while extracting review knowledge (Wang, Cai, and Wang 2022). However, existing GRS are often not specifically designed to consider how to extract useful review knowledge and make the obtained review representations more effective (Liu et al. 2020a).
- **Consistent Interaction Behavior**. Different behavioral

*Corresponding author.

interactions of a user in the same time period are also often consistent. For example, a user gave one item a low star (e.g., 2 stars), and generally expressed his dissatisfaction in his review. In turn, one other item drew lavish praise from this user attached with a 5 star rating. Intuitively, while ratings and reviews offer complementary signals (Liu et al. 2020b), they also exist behavioral associations that can be utilized to generate more robust and interactive embeddings. However, existing GRS lacks the exploration of this association.

To tackle the above challenges, we propose a graph-based **Re**view-enhanced **H**ierarchical **C**ontrastive **L**earning approach, called **ReHCL**.

Firstly, we construct topic and semantic graphs to preserve the key topical relations and contextual semantic relations between users (or items) and then employ graph encoders to produce the high-order node representations. Separate graph encoding processes will encourage ReHCL to generate node representations at different levels simultaneously, fully capturing individual topic and holistic semantic preferences of users. From Figure 1, we apply LDA (Blei, Ng, and Jordan 2003) and BERT (Reimers and Gurevych 2019) to extract topic factors and semantic embeddings, respectively. Then similarity calculation strategies are used to judge whether there are connection edges between users, which help construct topic and semantic graphs.

Secondly, we design a cross-view Graph Contrastive Learning (GCL) between topic and semantic views to distill more robust representations from reviews. GCL pulls together the same node representations and pushes the different node representations away in the two views (Peng et al. 2020). It automatically identifies the effective information of data itself by generating self-supervised signals (Wu et al. 2021). Therefore, by promoting the alignment of the two associated views, each view can extract supplemental knowledge from the other view, and perform cross-view contrast to achieve mutual enhancement of node representations at topic and semantic levels. Since the two views are derived from the same source of review data, we design a neighbor-based positive sampling to mine the intrinsic graph-structured similarity by searching for the same neighbors of the node in different views (e.g., users u_5 and u_6 in Figure 1). This strategy generates positive samples from the same view to relieve inefficient contrast that only utilizes positive pairs from different views, further refining effective review knowledge and reducing redundant noise.

Thirdly, we build a cross-modal contrastive learning to explore the association between ratings and reviews, which regards rating and review interactions as two modalities, learning informative representations by maximizing the similarity between the two modalities. Since ratings and reviews are heterogeneous, we design a projection function to match the generated rating and review representations. This requires similarity in representations while preserving the distinctive information that each modality brings (Salvador et al. 2021).

Specifically, the cross-view GCL and cross-modal contrastive learning form a hierarchical contrastive learning task to capture different self-supervised signals by jointly modeling ratings and reviews. Then, we further combine the hierarchical contrastive learning task with the primary recommendation task, and conduct joint training to output the final representations of users and items for item recommendation.

Our contributions are summarized as follows:

- We establish the topic and semantic graphs to fully mine review relations from different views. We present a neighbor-based positive sampling to identify the effective information of graph-structure data itself, helping extract useful review information and reduce redundant noise.
- We propose a graph-based model ReHCL to combine cross-view GCL and cross-modal contrastive learning by jointly modeling ratings and reviews.
- Extensive experiments are conducted on three datasets to verify the superiority of ReHCL over strong baselines.

Preliminaries and Related Work

Let $\mathcal{U} = \{u_1, u_2, ..., u_M\}$ and $\mathcal{I} = \{i_1, i_2, ..., i_N\}$ be the set of users and items, respectively. Each interaction between users and items can be defined as a tuple $(u, i, y_{u,i}, d_{u,i})$, where $y_{u,i}$ denotes the interaction that user u has rated item i and $d_{u,i}$ is the review comment that u described i. Let $\mathcal{P}^+ = \{y_{u,i} | u \in \mathcal{U}, i \in \mathcal{I}\}$ denote the observed rating interactions.

Definition 1: User-item Rating Graph. The user-item graph $\mathcal{G}^R = (\mathcal{V}^R, \mathcal{E}^R)$ indicates rating interactions between users and items. $\mathcal{V}^R = \mathcal{U} \cup \mathcal{I}$ are the initial nodes involving all the users and items. $\mathcal{E}^R$ is the set of edges ($\mathcal{E}^R = \mathcal{P}^+$).

Definition 2: Topic Graph and Semantic Graph. Topic graph $\mathcal{G}^1$ is composed of the user-user topic graph $\mathcal{G}^1_u = (\mathcal{V}^1_u, \mathcal{E}^1_u)$ and item-item topic graph $\mathcal{G}^1_i = (\mathcal{V}^1_i, \mathcal{E}^1_i)$. $\mathcal{G}^1_u$ (or $\mathcal{G}^1_i$) records topic relations between users (or items), where the node set $\mathcal{V}^1_u$ (or $\mathcal{V}^1_i$) indicates all the users (or items) and the edge set $\mathcal{E}^1_u$ (or $\mathcal{E}^1_i$) denotes topic similarities between nodes. Analogously, we obtain semantic graph $\mathcal{G}^2_u$, consisting of user-user graph $\mathcal{G}^2_u = (\mathcal{V}^2_u, \mathcal{E}^2_u)$ and item-item graph $\mathcal{G}^2_i = (\mathcal{V}^2_i, \mathcal{E}^2_i)$, retaining semantic relations.

Graph-based Recommender Systems

Graph Neural Network. Graph Neural Networks (GNNs) introduce propagation or diffusion mechanism (Wang et al. 2019) to capture graph-structured knowledge (Berg, Kipf, and Welling 2017; Velickovic et al. 2017). Recent studies concentrate on converting reviews into graph structures, such as word-level graphs (Liu et al. 2021b), review-level graphs (Gao et al. 2020), and document-level graphs (Liu et al. 2020c; Zhu et al. 2020). They then employ GNNs to distill node-based or graph-based features from these graph-structured review patterns.

Graph Contrastive Learning. Graph Contrastive Learning (GCL) encourages the same node in different views to stay close to each other in the embedding space (Zhu et al. 2021; Zang et al. 2023). This enables self-discrimination of node representations in an unsupervised way and supplements the supervised task with the unlabeled data (Yu et al. 2021; Shuai et al. 2022). For example, SGL (Wu et al. 2021) designs three graph-based data augmentation operators, i.e., node dropout, edge dropout, and random walk, to reinforce node embeddings along the user-item graph.

Review-based Recommender Systems

Topic-level Methods. Topic-level methods use topic model (Bao, Fang, and Zhang 2014; Cheng et al. 2018) to extract topic factors from reviews. Typical works, such as HFT (McAuley and Leskovec 2013) and RBLT (Tan et al. 2016), introduce LDA (Blei, Ng, and Jordan 2003) to infer topic distributions of reviews. Despite their success in tapping into topical cues, they generally portray reviews as shallow features at the phrase level and thereby remain oblivious of plentiful semantic contents (Dong et al. 2020).

Semantic-level Methods. Semantic-level methods evaluate contextual information from reviews to capture semantic features (Choi et al. 2022). Achievable technical routes focus on deep network paradigm, containing CNNs (Zheng, Noroozi, and Yu 2017; Liu et al. 2019) and RNNs (Li et al. 2019). Moreover, recent BERT-based works (Devlin et al. 2018; Reimers and Gurevych 2019; Su et al. 2021) fine-tune Transformer (Vaswani et al. 2017) to encode abundant reviews and leverage multi-head self-attention to capture the contextual features. In this paper, we select the BERT-based model as the basic unit to exploit review texts in the semantic view and obtain semantic representations.

ReHCL

Figure 2 shows an overview of ReHCL, which jointly combines Cross-View Graph Contrastive Learning (CVGCL) and Cross-Modal Contrastive Learning (CMCL).

Graph Construction and Encoder

Topic Graph Construction. Given the set of reviews written by user u, we concatenate them as the user document $\mathcal{D}_u$. Here, we adopt LDA (Blei, Ng, and Jordan 2003) to extract the topic distribution of dimension K, where each dimension reveals a probability that user u enjoys a certain topic. This process is formulated as:

$$\boldsymbol{\theta}_u = \{\theta_u^1, \theta_u^2, ..., \theta_u^K\} = LDA(\mathcal{D}_u). \tag{1}$$

Then his topic preference is judged with the largest relevance probability (e.g., θ_u^k) to finalize the topic factor. This helps recognize the nub of review contents. We connect any two similar users with the same topic factor to construct the user-user topic graph $\mathcal{G}_u^1$, as shown in Figure 1(a). However, users' preferences are multi-faceted when considering fine-grained descriptions in reviews (Zhao et al. 2020), e.g., theme, author, scene in $Book$ domain. Hence, we pick out diversified factors to depict fine-grained procedures with the largest relevance probabilities (the number of topic factors larger than 2) and extend the topic graph to multi-aspect patterns. This also makes us easier to attach possible edges between similar users to explore diverse potential connections. Analogously, we obtain the item-item topic graph $\mathcal{G}_i^1$.

Semantic Graph Construction. To portray the whole context in reviews, we introduce siamese BERT networks (Reimers and Gurevych 2019) to encode review text. Each user-generated item review contributes to the user's interests, but only the significant ones play a more important role. Hence we utilize mean operation and max-pooling (Wang et al. 2022) to manufacture semantic embeddings:

$$\boldsymbol{p}_u = \frac{1}{|\mathcal{D}_u|} \sum_{i \in \mathcal{D}_u} \boldsymbol{p}_{u,i} + MaxPooling(\boldsymbol{p}_{u,1}, ..., \boldsymbol{p}_{u,|\mathcal{D}_u|}), \tag{2}$$

where $\mathcal{D}_u$ is a set of items that user u has reviewed. To match the semantic relation between two users, we measure cosine similarity between embeddings of users u and u':

$$Sim(u, u') = ReLU(\frac{\boldsymbol{p}_u \boldsymbol{p}_{u'}}{\|\boldsymbol{p}_u\| \, \|\boldsymbol{p}_{u'}\|}), \tag{3}$$

where $Sim(.)$ is similarity function. We then compute the similarities between all users and produce semantic edges with the top-Q cosine values (Liu et al. 2020c), building the user-user semantic graph $\mathcal{G}_u^2$ as shown in Figure 1(b). Analogously, we obtain the item-item semantic graph $\mathcal{G}_i^2$.

Graph Encoder We encoder initial nodes $[\boldsymbol{e}_v^{(0)}]_{v \in \mathcal{U} \cup \mathcal{I}}$ and aggregate their neighbors to promote message propagation. For target node v at l_{th} propagation layer, we aggregate the embeddings of neighbor nodes to update the embeddings of the target node iteratively. Here, we select the simple but effective LightGCN (He et al. 2020) as the encoder:

$$\boldsymbol{e}_v^{(l+1)} = \sum_{j \in \mathcal{N}_v} \frac{1}{\sqrt{|\mathcal{N}_v \| \mathcal{N}_j|}} \boldsymbol{e}_j^{(l)}, \tag{4}$$

where $\frac{1}{\sqrt{|\mathcal{N}_v \| \mathcal{N}_j|}}$ is a symmetric normalization constant. $\mathcal{N}_v$ and $\mathcal{N}_j$ denote the neighbors of v and j. The matrix form propagation rule can be described as follows:

$$\boldsymbol{E}^{(l+1)} = \mathcal{L} \boldsymbol{E}^{(l)}, \tag{5}$$

where $\mathcal{L}$ is the Laplacian matrix of the target graph. After obtaining L layer embeddings, the graph encoding adopts average function to produce the high-order representations:

$$\boldsymbol{E} = \frac{1}{L+1} \sum_{l=0}^{L} \boldsymbol{E}^{(l)}, \tag{6}$$

In addition, we also choose other graph encoders to compare with LightGCN in the experimental part.

Cross-View Graph Contrastive Learning

Embedding-based Data Augmentation. Following the previous step, we generate two types of review representations ($\boldsymbol{E}^1$ and $\boldsymbol{E}^2$) by encoding topic and semantic graphs, respectively. Since our design generates two different views, we develop an embedding-based augmentation to improve the robustness of the model. Compared with graph-based augmentation which revises the topological structure on the original graph, embedding-based augmentation revamps the learned propagated embeddings during the graph encoding process. Specifically, we adopt the random dropout strategy (Srivastava et al. 2014) to independently handle these two views, and the process is formulated as follows:

$$\widetilde{\boldsymbol{E}}^1 = Dropout(BatchNorm(\boldsymbol{E}^1), \gamma), \tag{7}$$

$$\widetilde{\boldsymbol{E}}^2 = Dropout(BatchNorm(\boldsymbol{E}^2), \gamma), \tag{8}$$

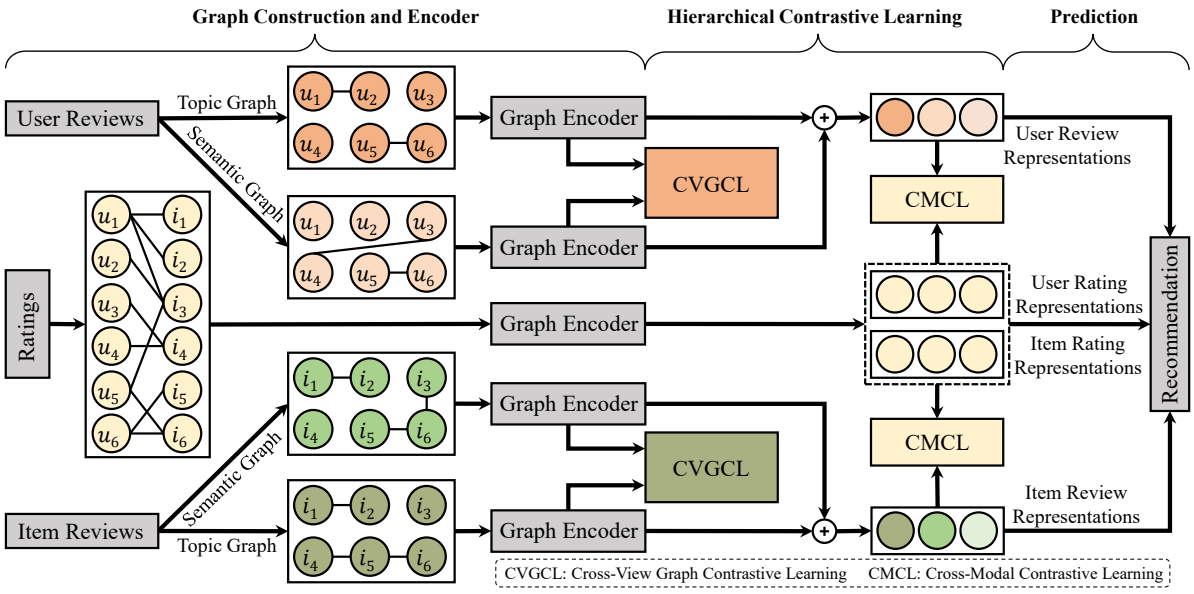


Figure 2: Overview of the proposed review-enhanced hierarchical contrastive learning.

where $\widetilde{\boldsymbol{E}}^1$ and $\widetilde{\boldsymbol{E}}^2$ indicate augmented data from topic and semantic views, respectively. $BatchNorm$ accelerates neural network training. $Dropout$ randomly discards some object of embeddings with the probability γ.

Graph Contrastive Learning. GCL enforces the agreement between the same node representations from different views while promoting the divergence between different nodes. Formally, we follow InfoNCE (Oord, Li, and Vinyals 2018) to define contrastive loss and maximize the similarity of positive pair (i,e,. $\{(\tilde{\boldsymbol{e}}_v^1, \tilde{\boldsymbol{e}}_v^2)|v \in \{\mathcal{U} \cup \mathcal{I}\}\}$) and minimize that of the negative pairs (i,e,. $\{(\tilde{\boldsymbol{e}}_v^1, \tilde{\boldsymbol{e}}_{v'}^2)|v, v' \in \{\mathcal{U} \cup \mathcal{I}\}, v \neq v'\}$):

$$\mathcal{L}_{gcl} = \sum_{v \in \{\mathcal{U} \cup \mathcal{I}\}} -\log \frac{exp(sim(\tilde{\boldsymbol{e}}_v^1, \tilde{\boldsymbol{e}}_v^2)/\tau)}{\sum\limits_{v' \in \{\mathcal{U} \cup \mathcal{I}\}} exp(sim(\tilde{\boldsymbol{e}}_v^1, \tilde{\boldsymbol{e}}_{v'}^2)/\tau)}, \tag{9}$$

where $sim(\cdot)$ is the discriminator which computes the similarity between two vectors. τ is a temperature parameter. By extracting self-supervised signals from unlabeled data over two different views, GCL accomplishes self-discrimination of review information and enhances review representations.

Neighbor-based Positive Sampling. The aforementioned loss only notes positive pairs for different views and neglects the intra-view positive samples, which leads to inefficient contrast, as the two graphs constructed from the same sources of review data share similar graph-structured signals. This enables us to find a flexible principle: if an anchor node has the same neighbor node in different views, then we take the same neighbor node as its positive samples.

For example, the sample pair (u_6,u_5) from Figure 3 can be regarded as an intra-view positive pair since the node u_6 in different views has the same 1-hop neighbor u_5. Marking strongly related intra-view nodes as positive samples will pull together their embeddings in the latent space. Hence, this sampling strategy connects the anchor node and its similar neighbors to strengthen positive samples and expand the applicability of graph contrastive learning. Finally, we combine intra-view positive pairs (i,e,. $\{(\tilde{\boldsymbol{e}}_v^1, \tilde{\boldsymbol{e}}_{v^+}^1)|v^+ \in \mathcal{N}_v^+\}$) and inter-view positive pair (i,e,. $\{(\tilde{\boldsymbol{e}}_v^1, \tilde{\boldsymbol{e}}_v^2)|v \in \{\mathcal{U} \cup \mathcal{I}\}\}$) to update the graph contrastive loss:

$$\mathcal{L}_{gcl} = \sum_{v \in \{\mathcal{U} \cup \mathcal{I}\}} -\log \left(\frac{\overbrace{exp(sim(\tilde{\boldsymbol{e}}_v^1, \tilde{\boldsymbol{e}}_v^2)/\tau)}^{\text{inter-view positive pair}}}{\sum\limits_{v' \in \{\mathcal{U} \cup \mathcal{I}\}} exp(sim(\tilde{\boldsymbol{e}}_v^1, \tilde{\boldsymbol{e}}_{v'}^2)/\tau)} + \frac{\overbrace{\sum_{v^+ \in \mathcal{N}_v^+} exp(sim(\tilde{\boldsymbol{e}}_v^1, \tilde{\boldsymbol{e}}_{v^+}^1/\tau)}^{\text{intra-view positive pairs}}}{\sum\limits_{v' \in \{\mathcal{U} \cup \mathcal{I}\}} exp(sim(\tilde{\boldsymbol{e}}_v^1, \tilde{\boldsymbol{e}}_{v'}^2)/\tau)} \right), \tag{10}$$

where $\mathcal{N}_v^+$ denotes the same neighbors of node v in two different views. We combine user side $\mathcal{L}_{gcl}^{user}$ and item side $\mathcal{L}_{gcl}^{item}$ to generate GCL loss as $\mathcal{L}_{gcl} = \mathcal{L}_{gcl}^{user} + \mathcal{L}_{gcl}^{item}$.

Cross-Modal Contrastive Learning

Having producing $\boldsymbol{e}_v^1$ and $\boldsymbol{e}_v^2$, we concatenate them as the final review representation $\boldsymbol{e}_v^{re} = [\boldsymbol{e}_v^1||\boldsymbol{e}_v^2]$. Meanwhile, we apply LightGCN to encode the user-item graph $\mathcal{G}^R$ and obtain the rating representations $\boldsymbol{e}_v^{ra}$. Motivated by the superiority of CLIP (Radford et al. 2021) and CrossCLR (Zolfaghari et al. 2021) to explore cross-modal paired data, we design a cross-modal contrastive learning network to explore the association between ratings and reviews.

Since ratings and reviews are two different structures of data, the information contained in a review cannot be fully

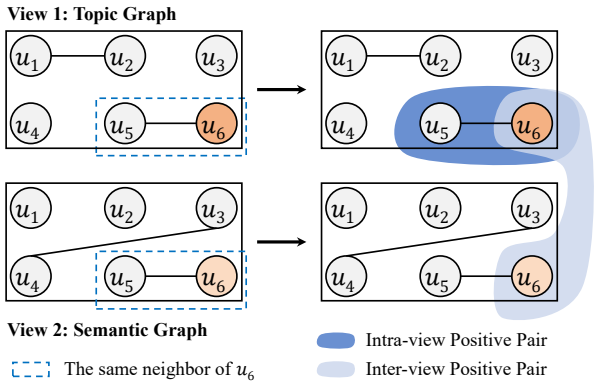


Figure 3: A toy example of neighbor-based positive samples.

presented in a rating. Thus, we introduce a projection function to map $\boldsymbol{e}_v^{re}$ to another latent space as $\tilde{\boldsymbol{e}}_v^{re}$ via a MLP:

$$\tilde{\boldsymbol{e}}_v^{re} = MLP(\boldsymbol{e}_v^{re}). \tag{11}$$

Aiming to mine similar signals between different modalities, we weaken the impact of negative samples on loss function distinguishing from the original InfoNCE:

$$\mathcal{L}_{cro} = \sum_{v \in \mathcal{U}/\mathcal{I}} -\log \frac{exp(sim(\boldsymbol{e}_v^{ra}, \tilde{\boldsymbol{e}}_v^{re})/\tau)}{exp(sim(\boldsymbol{e}_v^{ra}, \tilde{\boldsymbol{e}}_v^{re})/\tau) + \rho Neg}, \tag{12}$$

where $Neg = \sum_{v' \in \{\mathcal{U} \cup \mathcal{I}\}, v' \neq v} exp(sim(\boldsymbol{e}_v^{ra}, \tilde{\boldsymbol{e}}_{v'}^{re})/\tau)$ indicates negative pairs. ρ is a hyper-parameter that controls the strength of negative samples. Through our design, the representations of these two modalities are similar but still retain their individual information. Specifically, we combine user side $\mathcal{L}_{cro}^{user}$ and item side $\mathcal{L}_{cro}^{item}$ to obtain cross-modal contrastive objective as $\mathcal{L}_{cro} = \mathcal{L}_{cro}^{user} + \mathcal{L}_{cro}^{item}$.

Prediction

We sum two generated modalities ($\boldsymbol{e}_v^{ra}$ and $\boldsymbol{e}_v^{re}$) to manufacture the final representations of users and items, namely $\boldsymbol{e}_u$ and $\boldsymbol{e}_i$. Then we employ the inner product to infer the predicted rating that user u would give target item i:

$$\hat{y}_{u,i} = \boldsymbol{e}_u \boldsymbol{e}_i^\top. \tag{13}$$

Here, the item recommendation is typically presented as a supervised learning task with the supervision signals coming from the observed interactions $\mathcal{P}^+$. The Bayesian Personalized Ranking (BPR) loss (Rendle et al. 2012) can be adopted as our optimization benchmark:

$$\mathcal{L}_{rec} = \sum_{(u,i,j) \in \mathcal{P}} -\ln \sigma(\hat{y}_{u,i} - \hat{y}_{u,j}), \tag{14}$$

where $\mathcal{P} = \{(u,i,j)|(u.i) \in \mathcal{P}^+, (u.j) \in \mathcal{P}^-\}$ is training set and $\mathcal{P}^- = \mathcal{U} \times \mathcal{I} / \mathcal{P}^+$ is a set of unobserved interactions.

Finally, we combine the recommendation task and two contrastive learning tasks to form a joint learning objective:

$$\mathcal{L} = \mathcal{L}_{rec} + \beta_1 \mathcal{L}_{gcl} + \beta_2 \mathcal{L}_{cro} + \beta_3 ||\Theta||_2^2, \tag{15}$$

where β_1 and β_2 are two hyperparameters that control the strength of CVGCL and CMCL. β_3 is the weight of regularization term $||\Theta||_2^2$ and Θ denotes model parameters.

Dataset	#Users	#Items	#Interactions	Density
Instrument	1,429	900	10,261	0.798%
Music	5,541	3,568	64,706	0.327%
Toy	19,412	11,924	167,597	0.072%

Table 1: Statistics of datasets.

Experiments

Experimental Settings

Dataset. We evaluate our model on Amazon dataset (McAuley and Leskovec 2013)[1], which contains ratings and user-generated reviews. Following previous studies (Chen et al. 2018; Shuai et al. 2022), we randomly split the user–item pairs of each dataset into 80% training set, 10% validation set, and 10% testing set. The detailed statistics of the datasets are summarized in Table 1.

Evaluation Metric. To evaluate the top-N recommendation performance, we employ three widely used metrics: Hit Ratio (HR), Mean Reciprocal Rank (MRR), and Normalized Discounted Cumulative Gain (NDCG). We run each experiment five times and report the average results.

Comparison Baselines. We compare our ReHCL with different lines of item recommendation methods.

- **Rating-based GNN** uses GNNs to model user-item rating interactions, such as GC-MC (Berg, Kipf, and Welling 2017), GCN (Kipf and Welling 2016), NGCF (Wang et al. 2019), and LightGCN (He et al. 2020).
- **Rating-based GCL** uses GCL to exploit the user-item rating graph, such as SGL (Wu et al. 2021).
- **Review-based topic** adopts statistical models to infer review-based topic factors, such as TopicMF (Bao, Fang, and Zhang 2014) and ALFM (Cheng et al. 2018).
- **Review-based CNN** mainly designs CNNs to encode user-generated reviews, such as DeepCoNN (Zheng, Noroozi, and Yu 2017) and NARRE (Chen et al. 2018).
- **Review-based GNN** transforms reviews into semantic connectivity to construct graphs, such as HGNR (Liu et al. 2020c) and SSG (Gao et al. 2020).
- **Review-based GCL** proposes RGCL (Shuai et al. 2022) that combines review-enhanced edges with rating-based edges to produce self-supervised signals.

Implementation Details. ReHCL is implemented with Tensorflow. We adopt Adam optimizer with an initial learning rate of 10^{-3}. The layer number is 3 and the embedding size is 64. We used the $L2$ regularization and its weight β_3 is set to 10^{-4}. Each observed user-item interaction in the training stage is defined as a positive sample, and then a negative item that the user has never interacted with is sampled.

[1] http://jmcauley.ucsd.edu/data/amazon/

Model		Instrument			Music			Toy		
		H@10	M@10	N@10	H@10	M@10	N@10	H@10	M@10	N@10
(1) Rating-based GNN	GC-MC	0.2451	0.0682	0.1080	0.4614	0.1609	0.2308	0.3304	0.1237	0.1717
	NGCF	0.2500	0.0957	0.1317	0.4729	0.1854	0.2523	0.3379	0.1428	0.1841
	LightGCN	0.2661	0.0933	0.1336	0.4839	0.2156	0.2783	0.3459	0.1478	0.1906
(2) Rating-based GCL	SGL	0.2773	0.0941	0.1367	0.4889	0.2170	0.2794	0.3549	0.1507	0.2044
(3) Review-based topic	TopicMF	0.2969	0.1118	0.1575	0.4408	0.1765	0.2381	0.2796	0.1072	0.1472
	ALFM	0.3249	0.1419	0.1849	0.4848	0.1946	0.2618	0.3090	0.1218	0.1653
(4) Review-based CNN	DeepCoNN	0.3213	0.1427	0.1847	0.5376	0.2505	0.3180	0.3478	0.1427	0.1905
	NARRE	0.3432	0.1492	0.2033	0.5647	0.2637	0.3346	0.3617	0.1493	0.1987
(5) Review-based GNN	SSG	0.2605	0.0853	0.1259	0.4733	0.1944	0.2596	0.3469	0.1476	0.1919
	HGNR	0.2780	0.0999	0.1414	0.4958	0.2232	0.2873	0.3692	0.1556	0.2091
(6) Review-based GCL	RGCL	0.2731	0.0914	0.1335	0.4935	0.2042	0.2717	0.3636	0.1531	0.2049
(7) OURS	**ReHCL**	**0.3936***	**0.1612***	**0.2155***	**0.6116***	**0.2797***	**0.3570***	**0.3885***	**0.1833***	**0.2313***
Improv.		14.68%	8.03%	6.00%	8.30%	6.07%	6.71%	5.23%	17.80%	10.62%

Table 2: Performance results. The row of "Improv." indicates the improvement of the best result of ReHCL (boldface) compared with the best baseline (underscore). * indicates the statistical significance for $p < 0.05$ compared with the best baseline.

Performance Comparison

We compare the performance of ReHCL with six types of typical methods for item recommendation. Table 2 summarizes the detailed results on three datasets in terms of HR@10 (H@10), MRR@10 (M@10), and NDCG@10 (N@10). First, review-based baselines (Table 2 (3)-(6)) excel rating-based baselines (Table 2 (1)-(2)) baselines with the average improvements of 5.50% H@10, 11.46% M@10, and 9.49% N@10, illustrating the effectiveness of review knowledge. Besides, exploiting high-order signals along the review-based graph structure can generally exert a favorable impact on the performance. Especially, these review-enhanced graph-based baselines (Table 2 (5)-(6)) jointly capture multi-hop collaborative signals between ratings and reviews, surpassing other graph-based baselines (Table 2 (1)-(2)) that only model user-item rating interactions. Lastly, we observe relative improvements on three datasets for ReHCL of 24.83% H@10, 37.27% M@10, and 31.74% N@10 on average, compared to all the baselines. It also reveals the efficacy of distilling useful rating and review knowledge by fusing CVGCL and CMCL jointly.

Ablation Study

Effect of Graph Structures We evaluate the influence of different graphs of ReHCL (i.e., $\mathcal{G}^R$, $\mathcal{G}_u^1$, $\mathcal{G}_i^1$, $\mathcal{G}_u^2$, and $\mathcal{G}_i^2$), as shown in Table 3. First, Table 3 (3) simultaneously captures topic and semantic relations and establishes GCL with the improvement of at least 8.91%, compared to Table 3 (2) only recording a single view. Next, Table 3 (4) has better performance than a single modality (Table 3 (1) and (2)) by aligning a single view of review representations with the rating part. In addition, we discover that both user and item sides (Table 3 (5)) contribute to the performance by assembling user-user and item-item graphs derived from reviews to fill in sparse rating interactions. At last, the proposed ReHCL (Table 3 (6)) achieves the best gains by skillfully fusing these different types of graphs above.

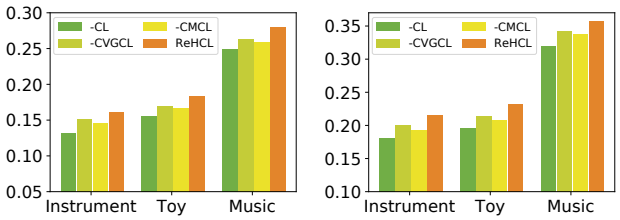
	Graph	Instrument	Music	Toy
(1)	$\mathcal{G}^R$	0.2941	0.4897	0.3483
(2)	$\mathcal{G}_u^1 + \mathcal{G}_i^1$	0.2906	0.4847	0.3371
	$\mathcal{G}_u^2 + \mathcal{G}_i^2$	0.2808	0.4523	0.3147
(3)	$\mathcal{G}_u^1 + \mathcal{G}_i^1 + \mathcal{G}_u^2 + \mathcal{G}_i^2$	0.3165	0.5121	0.3469
(4)	$\mathcal{G}^R + \mathcal{G}_u^1 + \mathcal{G}_i^1$	0.3242	0.5531	0.3541
	$\mathcal{G}^R + \mathcal{G}_u^2 + \mathcal{G}_i^2$	0.3522	0.5782	0.3657
(5)	$\mathcal{G}^R + \mathcal{G}_u^1 + \mathcal{G}_u^2$	0.3789	0.5848	0.3799
	$\mathcal{G}^R + \mathcal{G}_i^1 + \mathcal{G}_i^2$	0.3817	0.5830	0.3733
(6)	**ReHCL**	**0.3936**	**0.6116**	**0.3885**

Table 3: Comparison of different graphs in terms of HR@10. The results with the best performance are marked in bold.

Effect of Contrastive Learning Figure 4 shows the results of encoding graphs without contrastive learning (*w/o CL*), and discarding CVGCL and CMCL (*w/o CVGCL*, *w/o CMCL*). The remarkable gains of *w/o CVGCL* and *w/o CMCL* compared to *w/o CL* reveal that both contrastive parts produce a positive effect of recommendation. Moreover, the hierarchical model ReHCL is conducive to performance gain by mixing two types of contrastive learning and generating different levels of self-supervised signals.

Model Study

Effect of Graph Encoder. We replace the graph encoder of ReHCL with GCN (Kipf and Welling 2016) and NGCF (Wang et al. 2019) to obtain the variants ReHCL_GCN and ReHCL_NGCF. The results are illustrated in Table 4 (1). ReHCL outperforms ReHCL_GCN and ReHCL_NGCF. This is



(a) M@10 (b) N@10

Figure 4: Performance comparison over different contrastive learning tasks on M@10 and N@10.

mainly because LightGCN only preserves the most essential component in GCN , namely neighborhood aggregation, to simplify graph structure.

Effect of Embedding-based Data Augmentation. We try different ways of data augmentations to demonstrate their impacts. ReHCL_NA is defined as no augmentation. ReHCL_ED and ReHCL_ND refer to edge and node dropout of the graph-based augmentation, randomly discarding some edges and nodes with a ratio γ by revising the original graph.

$$EdgeDropout : \widetilde{\mathcal{G}} = ED(\mathcal{G}) = (\mathcal{V}, \boldsymbol{D}_1 \odot \mathcal{E}), \quad (16)$$

$$NodeDropout : \widetilde{\mathcal{G}} = ND(\mathcal{G}) = (\boldsymbol{D}_2 \odot \mathcal{V}, \mathcal{E}), \quad (17)$$

where $\boldsymbol{D}_1 \in \{0,1\}^{|\mathcal{E}|}$ and $\boldsymbol{D}_2 \in \{0,1\}^{|\mathcal{V}|}$ denote masking vectors by operating edge set $\mathcal{E}$ and node set $\mathcal{V}$ to produce augmented graphs $\widetilde{\mathcal{G}}$. The results in Table 4 (2) reveal the embedding-based augmentation achieves relative gain over the three ways above. We put the performance gains down to two main reasons. First, the message dropout strategy itself can prevent over-fitting of graph embedding. Further, by varying the learned embeddings to enrich the representations of nodes, embedding-based augmentation withstands more disturbances during the message-passing process and improves the robustness of ReHCL.

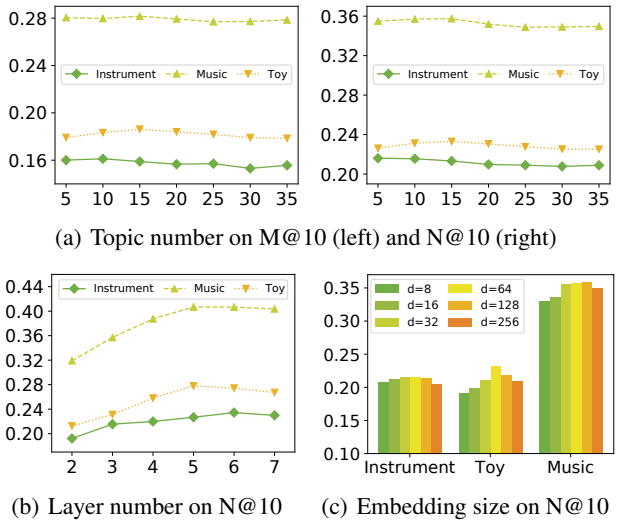
Effect of Neighbor-based Positive Sampling. To assess the effect of the neighbor-based positive sampling strategy, we compare it to typical InfoNCE (ReHCL_Info). The results in Table 4 (3) show that in contrast to ReHCL_Info, enforcing intra-view positive sampling upgrades contrastive capability by incorporating similar neighbor nodes into positive pairs. This is largely due to two factors. First, exploring positive nodes from similar neighbor nodes preserves original graph-structured information. Marking these related intra-view nodes as positive samples relieves inefficient contrast that only considers positive pairs from different views. In addition, strengthening positive samples conversely reduces false negative samples existing in the InfoNCE estimator and helps weaken the noise information.

Effect of Hyper-parameters

Topic Factor. We vary the number of latent factors (K) to evaluate the effect of topics in Figure 5 (a). Overall, the performance remains relatively stable within a certain margin, demonstrating ReHCL is insensitive to topic numbers.

	Model	Instrument	Music	Toy
(1)	ReHCL_GCN	0.3578	0.5601	0.3484
	ReHCL_NGCF	0.3690	0.5831	0.3606
(2)	ReHCL_NA	0.3873	0.6031	0.3788
	ReHCL_ED	0.3898	0.6022	0.3821
	ReHCL_ND	0.3824	0.5951	0.3785
(3)	ReHCL_Info	0.3908	0.6042	0.3806
(4)	ReHCL	0.3936	0.6116	0.3885

Table 4: Comparison of variants in terms of H@10.



(a) Topic number on M@10 (left) and N@10 (right)

(b) Layer number on N@10 (c) Embedding size on N@10

Figure 5: Performance results of hyper-parameters.

But too many topics easily confuse the user's intention when there are few subjects of interest to this user.

Layer Number. To investigate the impacts of multiple embedding propagation layers, we experiment with different model depths. Figure 5 (b) summarizes the results that the performance increases as layer number grows by capturing high-order signals. However, performance deteriorates when the layer number is larger than 6. This is mainly due to the over-smoothing issue (Liu et al. 2021a) that the embeddings of nodes get closer together until they become indistinguishable as stacking more layers.

Embedding Size. From Figure 5 (c), the suitable size of embedding parameters boosts the recommendation performance. However, sparse features assigned by too large embedding sizes (e.g., $d > 64$) are likely to lead to over-fitting problems and the performance starts to decline.

Conclusion

We proposed a graph-based learning paradigm ReHCL to effectively capture review knowledge and reduce redundant noise by combining cross-view and cross-modal contrastive learning efficiently. This process allowed us to generate high-quality representations to enhance the performance of item recommendation.

Acknowledgments

This research is supported in part by National Science Foundation of China (No. 62072304), Shanghai Municipal Science and Technology Commission (No. 21511104700), the Shanghai East Talents Program, the Oceanic Interdisciplinary Program of Shanghai Jiao Tong University (No. SL2020MS032),and Zhejiang Aoxin Co. Ltd.

References

Bao, Y.; Fang, H.; and Zhang, J. 2014. Topicmf: Simultaneously exploiting ratings and reviews for recommendation. In *Twenty-Eighth AAAI conference on artificial intelligence*.

Berg, R. v. d.; Kipf, T. N.; and Welling, M. 2017. Graph convolutional matrix completion. *arXiv preprint arXiv:1706.02263*.

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan): 993–1022.

Chen, C.; Zhang, M.; Liu, Y.; and Ma, S. 2018. Neural attentional rating regression with review-level explanations. In *WWW*, 1583–1592.

Cheng, Z.; Ding, Y.; Zhu, L.; and Kankanhalli, M. 2018. Aspect-aware latent factor model: Rating prediction with ratings and reviews. In *WWW*, 639–648.

Chin, J. Y.; Zhao, K.; Joty, S.; and Cong, G. 2018. ANR: Aspect-based neural recommender. In *CIKM*, 147–156.

Choi, Y.; Choi, J.; Ko, T.; Byun, H.; and Kim, C.-K. 2022. Based Domain Disentanglement without Duplicate Users or Contexts for Cross-Domain Recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 293–303.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Do, V.; Corbett-Davies, S.; Atif, J.; and Usunier, N. 2022. Online certification of preference-based fairness for personalized recommender systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 6532–6540.

Dong, X.; Ni, J.; Cheng, W.; Chen, Z.; Zong, B.; Song, D.; Liu, Y.; Chen, H.; and de Melo, G. 2020. Asymmetrical hierarchical networks with attentive interactions for interpretable review-based recommendation. In *AAAI*, volume 34, 7667–7674.

Gao, J.; Lin, Y.; Wang, Y.; Wang, X.; Yang, Z.; He, Y.; and Chu, X. 2020. Set-Sequence-Graph: A multi-view approach towards exploiting reviews for recommendation. In *CIKM*, 395–404.

He, X.; Deng, K.; Wang, X.; Li, Y.; Zhang, Y.; and Wang, M. 2020. LightGCN: Simplifying and powering graph convolution network for recommendation. *arXiv preprint arXiv:2002.02126*.

Kipf, T. N.; and Welling, M. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Li, C.; Niu, X.; Luo, X.; Chen, Z.; and Quan, C. 2019. A review-driven neural model for sequential recommendation. *arXiv preprint arXiv:1907.00590*.

Liu, D.; Li, J.; Du, B.; Chang, J.; and Gao, R. 2019. Daml: Dual attention mutual learning between ratings and reviews for item recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 344–352.

Liu, D.; Wu, J.; Li, J.; Du, B.; Chang, J.; and Li, X. 2020a. Adaptive Hierarchical Attention-Enhanced Gated Network Integrating Reviews for Item Recommendation. *IEEE Transactions on Knowledge and Data Engineering*.

Liu, F.; Cheng, Z.; Zhu, L.; Gao, Z.; and Nie, L. 2021a. Interest-aware Message-Passing GCN for Recommendation. In *Proceedings of the Web Conference 2021*, 1296–1305.

Liu, H.; Wang, Y.; Peng, Q.; Wu, F.; Gan, L.; Pan, L.; and Jiao, P. 2020b. Hybrid neural recommendation with joint deep representation learning of ratings and reviews. *Neurocomputing*, 374: 77–85.

Liu, S.; Ounis, I.; Macdonald, C.; and Meng, Z. 2020c. A heterogeneous graph neural model for cold-start recommendation. In *SIGIR*, 2029–2032.

Liu, Y.; Yang, S.; Zhang, Y.; Miao, C.; Nie, Z.; and Zhang, J. 2021b. Learning hierarchical review graph representations for recommendation. *IEEE Transactions on Knowledge and Data Engineering*.

McAuley, J.; and Leskovec, J. 2013. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, 165–172.

Oord, A. v. d.; Li, Y.; and Vinyals, O. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Peng, Z.; Huang, W.; Luo, M.; Zheng, Q.; Rong, Y.; Xu, T.; and Huang, J. 2020. Graph representation learning via graphical mutual information maximization. In *WWW*, 259–270.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 8748–8763. PMLR.

Reimers, N.; and Gurevych, I. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Ren, Y.; Zhang, H.; Li, Q.; Fu, L.; Ding, J.; Cao, X.; Wang, X.; and Zhou, C. 2022. Disentangled Graph Contrastive Learning for Review-based Recommendation. *arXiv preprint arXiv:2209.01524*.

Rendle, S.; Freudenthaler, C.; Gantner, Z.; and Schmidt-Thieme, L. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618*.

Salvador, A.; Gundogdu, E.; Bazzani, L.; and Donoser, M. 2021. Revamping cross-modal recipe retrieval with hierarchical transformers and self-supervised learning. In *CVPR*, 15475–15484.

Shuai, J.; Zhang, K.; Wu, L.; Sun, P.; Hong, R.; Wang, M.; and Li, Y. 2022. A Review-aware Graph Contrastive Learning Framework for Recommendation. *arXiv preprint arXiv:2204.12063*.

Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. 2014. Dropout: A simple way to prevent neural networks from overfitting. *JMLR*, 15(1): 1929–1958.

Su, J.; Cao, J.; Liu, W.; and Ou, Y. 2021. Whitening sentence representations for better semantics and faster retrieval. *arXiv preprint arXiv:2103.15316*.

Tan, Y.; Zhang, M.; Liu, Y.; and Ma, S. 2016. Rating-boosted latent topics: Understanding users and items with ratings and reviews. In *IJCAI*, volume 16, 2640–2646.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2017. Graph attention networks. *stat*, 1050: 20.

Wang, H.; Wang, T.; Li, S.; Guan, S.; Zheng, J.; and Chen, W. 2022. Heterogeneous Interactive Snapshot Network for Review-Enhanced Stock Profiling and Recommendation. In *IJCAI*, 3962–3969.

Wang, K.; Zhu, Y.; Liu, H.; Zang, T.; and Wang, C. 2023. Learning Aspect-Aware High-Order Representations from Ratings and Reviews for Recommendation. *ACM Transactions on Knowledge Discovery from Data*, 17(1): 1–22.

Wang, P.; Cai, R.; and Wang, H. 2022. Graph-based Extractive Explainer for Recommendations. In *Proceedings of the ACM Web Conference 2022*, 2163–2171.

Wang, X.; He, X.; Wang, M.; Feng, F.; and Chua, T.-S. 2019. Neural graph collaborative filtering. In *SIGIR*, 165–174.

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

Yu, J.; Yin, H.; Gao, M.; Xia, X.; Zhang, X.; and Viet Hung, N. Q. 2021. Socially-aware self-supervised tri-training for recommendation. In *SIGKDD*, 2084–2092.

Zang, T.; Zhu, Y.; Zhang, R.; Wang, C.; Wang, K.; and Yu, J. 2023. Contrastive Multi-View Interest Learning for Cross-Domain Sequential Recommendation. *ACM Transactions on Information Systems*.

Zhao, C.; Li, C.; Xiao, R.; Deng, H.; and Sun, A. 2020. CATN: Cross-domain recommendation for cold-start users via aspect transfer network. *arXiv preprint arXiv:2005.10549*.

Zheng, L.; Noroozi, V.; and Yu, P. S. 2017. Joint deep modeling of users and items using reviews for recommendation. In *WSDM*, 425–434.

Zhu, F.; Wang, Y.; Chen, C.; Liu, G.; and Zheng, X. 2020. A Graphical and Attentional Framework for Dual-Target Cross-Domain Recommendation. In *IJCAI*, 3001–3008.

Zhu, Y.; Xu, Y.; Yu, F.; Liu, Q.; Wu, S.; and Wang, L. 2021. Graph contrastive learning with adaptive augmentation. In *Proceedings of the Web Conference 2021*, 2069–2080.

Zolfaghari, M.; Zhu, Y.; Gehler, P.; and Brox, T. 2021. Crossclr: Cross-modal contrastive learning for multi-modal video representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1450–1459.