Semantic Enhanced Heterogeneous Hypergraph Network for Collaborative Filtering

Mingtao Xu[1,2], **Wei Wei**[1,2*], **Peixuan Yang**[2,3], **Hulong Wu**[4]

[1] Cognitive Computing and Intelligent Information Processing (CCIIP) Laboratory,
School of Computer Science and Technology, Huazhong University of Science and Technology
[2] Joint Laboratory of HUST and Pingan Property & Casualty Research (HPL)
[3]Ping An Property & Casualty Insurance Company of China, Ltd
[4]Shenzhen Yishi Huolala Technology Limited
xumingtao252@gmail.com, weiw@hust.edu.cn, ypx_hust@163.com, senter.wu@huolala.cn

Abstract

Collaborative Filtering (CF) based on graph neural networks (GNNs) has yielded immense success for recommendation systems by capturing high-order dependencies from implicit feedback. Recently, the outstanding text comprehension ability of the Large Language Models (LLMs) has shown promising potential to provide auxiliary semantics for collaborative representation. However, when aligning textual information with collaborative signals, inconsistent semantics between user-item and item-item text pairs may lead to the degradation of the alignment model, thus hindering the recommender system from effectively utilizing heterogeneous information. In this paper, we propose a novel method: **S**emantic **E**nhanced **H**eterogeneous **H**ypergraph **N**etwork (SEHHN), which enhances the representations of CF correlations with semantics, thereby avoiding alignment degradation. To better model the collaborative signals, we design a graph autoencoder that captures the bidirectional relationship between user preferences and item features in review semantics. Furthermore, we develop an LLM-based item classifier to adaptively exploit potential correlations of items via the co-occurrences of item features. Finally, we design a heterogeneous hypergraph network to achieve efficient alignment and propagation of heterogeneous information, thereby alleviating the impact of semantic inconsistency on CFs. Extensive experiments on three real-world datasets demonstrate that our proposed SEHHN outperforms existing SOTA methods and validates the effectiveness of each component.

Introduction

Recommender systems (RSs) have emerged as a core engine for personalized information filtering on the Internet by providing users with potentially interesting items. Among them, collaborative filtering (CF) has become the central theme of RS due to its powerful ability to model complex relationships from historical interactions (He et al. 2017). By leveraging graph neural networks (GNNs) to encode high-order dependencies such as user-item collaboration (Chen et al. 2020) and item-item co-occurrence (Ye, Xia, and Huang 2023) in recommender systems, a series of GNN-based CF methods have greatly improved the baseline performance and attracted widespread attention from researchers.

*Corresponding author.

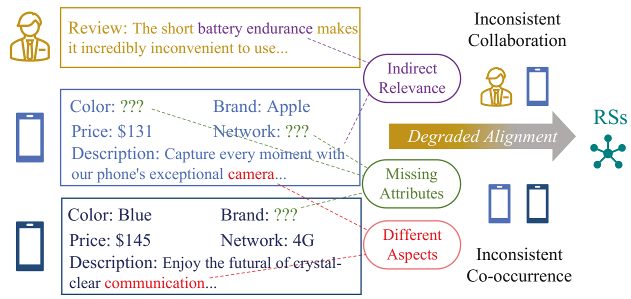

Figure 1: An illustration of semantic inconsistency within user-item collaboration and item-item co-occurrence.

However, the above methods only capture user preferences via ID-based information from implicit feedback, neglecting the benefits of auxiliary semantics conveyed in interaction contexts and textual attributes. Recently, Large Language Models (LLMs) like GPT3 (Brown et al. 2020) and LLaMA (Touvron et al. 2023) have been proposed to empower general models with text comprehension and natural language reasoning capacities (Zhao et al. 2023b), which shows promising potential to enrich collaborative representation with additional semantics. Building on this, researchers have proposed using LLMs to integrate auxiliary textual information into recommendation systems. For instance, KAR (Xi et al. 2023) uses LLM to deduce the knowledge of users and items and extracts relevant semantic representation to enhance collaborative embeddings. RLMRec (Ren et al. 2024) further improves the exploitation of semantic information by optimizing collaborative embeddings with a mutual information maximizing paradigm.

While the introduced semantics improve recommendation performance, they still face the challenge of **semantic inconsistency** during representation alignment. Specifically, CFs mainly utilized user-item collaboration and item-item co-occurrence to learn representations, leading to inherent similarities between related entity embeddings. On the other hand, the similarity in the LLM-extracted representations is solely associated with the semantic overlap in the textual description. Therefore, if the above dependencies cannot be accurately depicted in the text, the semantic embeddings of users or items offered by LLM will not exhibit the corresponding similarity as in CF's representation, which can be elaborated by the analysis of the following main relations:

- User-item collaboration: The description texts of users and items might not convey direct relevance between preferences and characteristics. As shown in the upper part of Figure 1, the user expresses dissatisfaction with the phone's battery life, but the corresponding item description does not involve relevant information. As a result, the representations generated by LLMs for the user and the item might not be semantically consistent and fail to capture the interactive relation between them.
- Item-item co-occurrence (Wang et al. 2020): Some common features between items will make them co-occur in the user's interaction history. Nevertheless, the lower portion of Figure 1 demonstrates that certain essential features may be omitted from the description, and the textual attributes of different items may possess unique emphases. These factors make it challenging to represent the co-occurrence through the consistency of semantic representation, thus hindering the effective utilization of feature information.

In the above cases, the training objectives of recommender systems (such as BPR and contrastive loss) require the model to align a pair of inconsistent semantic representations with similar collaborative representations. This unreasonable task may lead to the degradation of the aligning model, i.e., the aligning model is required to output similar collaborative embeddings even when the input semantics do not justify such similarity.

In light of the issues mentioned above, our objective is to prevent alignment degradation by enriching the relations in collaborative filtering with semantic information. In this paper, we propose a novel approach named **S**emantic **E**nhanced **H**eterogeneous **H**ypergraph **N**etwork (SEHHN). To enhance the representation of collaboration, we leverage the semantics of explicit feedback (i.e., reviews) to exploit the interconnection among users and items. Since review reflects not only users' subjective evaluation of items but also items' features related to user preferences, we design a graph autoencoder to align review semantics through bidirectional relations. Moreover, we develop an LLM-based item classifier to mine potential co-occurrences via their descriptions. The reasoning ability of LLMs enables the classifier to consolidate inconsistent semantics into regularized discrete categories, thus adaptively extracting the commonalities in item features. In addition, we designed a heterogeneous hypergraph network to achieve alignment and propagation of heterogeneous information (Zhao et al. 2023a). The hypergraph comprises four types of nodes (users, items, categories, reviews) and defines beyond-pairwise dependencies through four types of hyperedges (interactions, observations, preferences, features). Finally, we design three aggregators to propagate heterogeneous information on each hyperedge and implement recommendations based on the derived entity representations.

We summarize our main contributions as follows:

- We discuss the semantic inconsistency and alignment degradation issues in LLM-enhanced collaborative filtering, and then address these challenges by introducing semantics to enhance the representation of collaboration and co-occurrence.
- We propose a heterogeneous hypergraph network that aggregates representations from different modalities to achieve efficient alignment and propagation of heterogeneous information.
- Our experiments on three real-world datasets demonstrate the superiority of SEHHN over SOTA methods and the effectiveness of each component.

Related Work

GNN-based Collaborative Filtering

Collaborative filtering (CF) aims to capture user preferences and recommend items that users are interested in based on large-scale interaction history (Juan, Yue-xin, and Chunying 2019; He et al. 2017). Due to the remarkable ability of graph neural networks to encode structural information, graph-based collaborative filtering has been intensively studied (Lin et al. 2022; Zhang et al. 2016; Ren et al. 2023b). For example, NGCF (Wang et al. 2019b), GCCF (Chen et al. 2020), and LightGCN (He et al. 2020) introduced and improved graph-based network structures to model complex dependencies in user-item interactions. However, the heavy reliance on high-quality interactions limits the effectiveness of GNN-based CF models due to data sparsity and noise. Recent research explores utilizing self-supervised learning (SSL) to provide improved interactive information, such as SGL (Wu et al. 2021), SimGCL (Yu et al. 2022), and AutoCF (Xia et al. 2023). Despite the achievements, they overlook the rich semantics of side-information in the interaction context. In this paper, we leverage the textual semantics extracted by LLMs to improve collaboration and co-occurrence dependencies, thereby alleviating the perturbations of data sparsity and noise on entity representation.

LLM-enhanced Recommendation

Given the superior semantic comprehension and reasoning abilities of Large Language Models (LLMs) (Devlin et al. 2018; Brown et al. 2020; Touvron et al. 2023), researchers have explored using them in various tasks of recommendation systems (Bao et al. 2023a; Yue et al. 2023; Wang and Lim 2023). Since LLMs have difficulty in utilizing collaborative information, recent studies have focused on bridging the semantic gap by integrating LLMs and recommender models, which generally follow two technical routes. The first approach uses collaborative information to assist LLMs (Bao et al. 2023b), for example, CoLLM (Zhang et al. 2023) introduced external recommendation models into LLM to utilize collaborative representations. On the other hand, considering that directly using LLM as a recommender will result in significant computational overhead, the second methods utilize LLMs to provide auxiliary semantics for conventional RSs, ensuring scalable recommendations (Hou et al. 2022a; Li et al. 2023). RLMRec (Ren et al. 2024) extracts semantic information from descriptive text through LLMs, and optimizes representations of users and items through alignment, producing state-of-the-art performance. In this paper, we discuss the alignment degradation caused by inconsistent textual semantics, and introduce review and category information to enhance the representations of collaboration and co-occurrence in CFs. By enriching these major relationships in recommendation systems, the proposed SEHHN better exploits the semantics provided by LLM.

Preliminary

Problem Statement

Let $\mathcal{U} = \{u_1, u_2, ..., u_{|\mathcal{U}|}\}$ and $\mathcal{I} = \{i_1, i_2, ..., i_{|\mathcal{I}|}\}$ denote the set of users and items, respectively. We utilize the adjacency matrix M_{ui} to record users' historical interactions, where its element $\mathrm{m}_{nm} = 1$ indicates that user u_n has interacted with item i_m. Following each interaction, users will offer a review $\mathcal{R}$ to describe their satisfaction, which we utilize to enhance the interaction semantics. For convenience, we use adjacent matrixes M_{ur} and M_{ir} to label the user and item corresponding to each review. The CF-based recommendation system aims to learn the representation of user $\mathbf{E}^u \in \mathbb{R}^{|\mathcal{U}| \times d}$ and item $\mathbf{E}^i \in \mathbb{R}^{|\mathcal{I}| \times d}$. Finally, we predict the following top-K items that users will most likely interact with.

Inference of Item Categories

We first utilize an LLM to classify items into pre-defined categories, which aims to consolidate inconsistent semantics in item features and adaptively extract commonalities between items. According to the actual text form in the specific dataset, we utilize items' reviews or descriptions to construct a prompt for LLM as the basis for classification. The prompt template of the *Book* dataset is shown in Figure 2. The classification result can be represented by the adjacency matrix M_{ic}. Among them, the specific semantics in the category set are collected from the dataset, but its corresponding representation is learned through collaborative filtering (Wang et al. 2019a). Let $\mathcal{C}$ denote the set of categories; we learn the corresponding embedding matrix $\mathbf{E}^c \in \mathbb{R}^{|\mathcal{C}| \times d}$.

> Classify the book into one or more categories based on their description with the following instructions:
> 1. The classification must be in the given category list.
> 2. Provide your classification in this structure: "a list of categories"
> 3. Do not provide any other text outside the string.
> Here are all the candidate categories of books: [Fiction, Technology, ..., Arts]
> The description of this book: <*Description*>

Figure 2: Prompt template for *Book* dataset.

Heterogeneous Hypergraph Construction

Since reviews and categories naturally correspond to collaboration and co-occurrence in RSs, the degradation of the model can be avoided as much as possible under their guidance. To empower collaborative filtering with diverse semantics, we construct a heterogeneous hypergraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, which is illustrated on the left side of Figure 3. The heterogeneous nodes $\mathcal{V}$ include users $\mathcal{V}^u$, items $\mathcal{V}^i$, reviews $\mathcal{V}^r$, and categories $\mathcal{V}^c$. Since the relationships between nodes are beyond pairwise, we design four types of hyperedges $\mathcal{E}$ to represent the complex relations.

(1) **Interaction hyperedge** $\mathcal{E}^i$, which connects a user node, an item node, and their corresponding review node within an interaction.

(2) **Observation hyperedge** $\mathcal{E}^o$, which is made up of an item node and every review node associated with it.

(3) **Preference hyperedge** $\mathcal{E}^p$, which includes a user node and the reviews on every item it has interacted with.

(4) **Feature hyperedge** $\mathcal{E}^f$, which contains an item node and the related category nodes.

Among them, $\mathcal{E}^i, \mathcal{E}^o, \mathcal{E}^p$ are derived by adjacent matrix M_{ur} and M_{ir}, and $\mathcal{E}^f$ is defined by M_{ic} given by the LLM-based classifier. The four types of hyperedges $\mathcal{E} = \mathcal{E}^i \cup \mathcal{E}^o \cup \mathcal{E}^p \cup \mathcal{E}^f$ collectively define which nodes in a heterogeneous hypergraph are adjacent and serve as paths for subsequent message passing. In addition, $\mathcal{E}^o$, $\mathcal{E}^p$, and $\mathcal{E}^i$ are utilized to generate subgraphs corresponding to review nodes for the graph autoencoder, thereby achieving alignment of heterogeneous information guided by collaborative signals.

Methodology

In this section, we demonstrate the framework of the semantic enhanced heterogeneous hypergraph network. The workflow of SEHHN is presented on the right side of Figure 3.

Aligning via Graph AutoEncoder

To align the review semantics into the collaborative space and obtain the representation of review nodes, we exploit their dependencies with other heterogeneous nodes through a graph autoencoder (GAE). The encoder compresses the high-dimensional semantics into a low-dimensional collaborative representation using the interconnection of reviews in users and items, while the decoder provides supervised signals for alignment by reconstructing semantic representation with user and item embedding.

Firstly, we utilize an LLM to encode the review texts into semantic representations.

$$S = \mathrm{LLM}(\mathcal{R}) \tag{1}$$

where $\mathcal{R}$ contains review texts, and $S \in \mathbb{R}^{|\mathcal{R}| \times d_s}$ is the corresponding higher-order embeddings.

Then, we calculate the adjacency matrix representing the relationships between review nodes based on observation hyperedges $\mathcal{E}^o$ and preference hyperedges $\mathcal{E}^p$. Since the number of review nodes is proportional to the number of interactions, we construct a lightweight review adjacency by relationship sampling:

$$\mathrm{M}_{rr} = \mathrm{Sample}_{p_g}(\mathrm{M}_{ir}^{\mathrm{T}} \mathrm{M}_{ir} + \mathrm{M}_{ur}^{\mathrm{T}} \mathrm{M}_{ur}) + \mathbf{I}, \tag{2}$$

where p_g is the sampling probability, M_{ir} and M_{ur} are adjacent matrices containing the relation among reviews, users, and items. We utilize the graph attention network (GAT) (Velickovic et al. 2017) as a graph encoder to compress semantic information. Let $s \in \mathbb{R}^{d_s}$ be the embedding

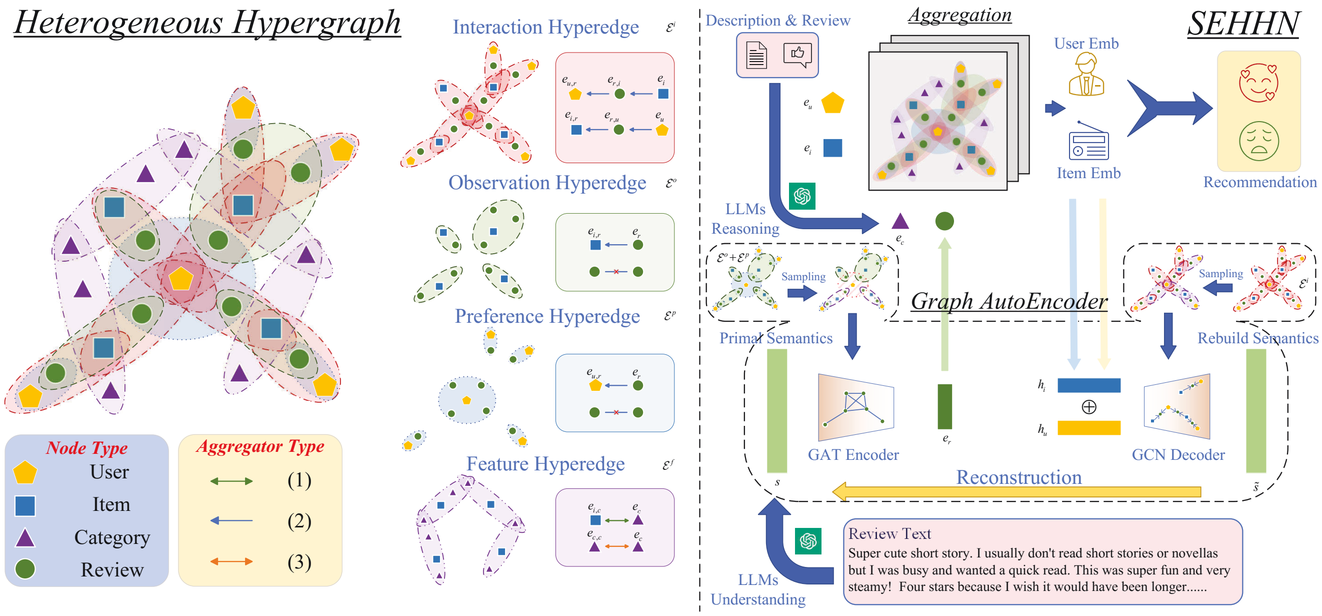

Figure 3: An overview of the proposed SEHHN. The left side shows the construction of the heterogeneous hypergraph and its information propagation patterns with three aggregators, while the right side demonstrates the proposed graph autoencoder and the architecture of the LLM-enhanced framework.

of a target review node from S, N_s represents its neighborhood that defined by $\mathrm{M}_{rr} > 0$. Formally, we have:

$$e_r = \sum_{s_j \in N_s} \alpha_j \cdot \mathrm{MLP}_1(s_j) \tag{3}$$

$$\alpha_j = \frac{\exp(q_s^{\mathrm{T}}[s; s_j])}{\sum_{s_k \in N_s} \exp(q_s^{\mathrm{T}}[s; s_k])} \tag{4}$$

where $q_s \in \mathbb{R}^{2d_s}$ is the attention vector, [;] is a concatenation. The compressed semantic representation $e_r \in \mathbb{R}^d$ will participate in the subsequent heterogeneous hypergraph aggregation as the embedding of review nodes.

After heterogeneous hypergraph aggregation, the information of the review nodes is integrated into user and item representations h_u and h_i (described in detail in the next section). We utilize graph convolution network (GCN) (He et al. 2020) as graph decoder to reconstruct semantic representation, where the relations are defined by the interaction hyperedges $\mathcal{E}^i$.

$$\tilde{s} = \mathrm{MLP}_2(h_u \oplus h_i) \tag{5}$$

where $\oplus$ denotes the convolution sum of user embedding and item embedding corresponding to the review nodes through $\mathcal{E}^i$.

Finally, we guide the learning of the graph autoencoder through a reconstruction task, which can adaptively mine self-supervised signals by exploiting the collaborative relationships on both sides of the review nodes (Hou et al. 2022b):

$$\mathcal{L}_{\text{recon}} = -\sum_{s \in \mathcal{N}_s} \log \frac{\cos(s, \tilde{s})}{\sum_{s_j \in \mathcal{N}_s} \cos(s_j, \tilde{s})} \tag{6}$$

where function $\cos(\cdot)$ calculates the cosine similarity, $\mathcal{N}_s$ denotes the semantic embeddings that were chosen to reconstruct. In implementation, we randomly select certain semantic representations for reconstruction with a ratio of p_s.

Heterogeneous Hypergraph Aggregation

To enrich the relationships in CFs with diverse semantics, we design a heterogeneous hypergraph network to provide propagation paths for heterogeneous nodes. Among these nodes, review nodes contain semantic information retrieved by LLM, whereas the embeddings of the other three types of nodes are randomly initialized. We preserve the review semantics by truncating the update of each semantic representation e_r and preventing information from other heterogeneous nodes from entering the review node.

Then, in order to address the heterogeneity between nodes in each hyperedge, we classify the propagation between nodes into three categories: (1)bidirectional propagation between heterogeneous nodes; (2)unidirectional propagation between heterogeneous nodes; (3)bidirectional propagation between homogeneous nodes. We design different aggregators for each type of propagation to guide the message passing on the heterogeneous hypergraph, which can be described as follows:

$$e_{\tau,\tau'} = \sum_{e_{\tau'} \in N_{\tau'}} 1/\sqrt{|N_\tau||N_{\tau'}|} \cdot e_{\tau'} \tag{7a}$$

$$e_{\tau,\tau'} = \sum_{e_{\tau'} \in N_{\tau'}} 1/|N_{\tau'}| \cdot e_{\tau'} \tag{7b}$$

$$e_{\tau,\tau} = \sum_{e'_\tau \in N_\tau} \frac{\exp(q^{\mathrm{T}}[e_\tau; e'_\tau])}{\sum_{e''_\tau \in N_\tau} \exp(q^{\mathrm{T}}[e_\tau; e''_\tau])} \cdot e'_\tau, \tag{7c}$$

where e_τ is the representation of nodes with type τ, and N_τ is the set of nodes adjacent to the target node with type τ, $q \in \mathbb{R}^{2d}$ is the attention vector. Equations 7a, 7b, and 7c correspond to the above-mentioned types (1), (2), and (3) of information propagation, respectively. Following LightGCN, we define propagation between heterogeneous nodes through lightweight graph convolution. Since the type (2) propagation is unidirectional, we perform unilateral normalization instead of bidirectional normalization in type (1). Similar to the graph autoencoder in the previous section, we use GAT to capture the different importance of homogeneous neighboring nodes for type (3) propagation.

We indicate the specific type of pairwise information propagation split between each hyperedge and the corresponding aggregator on the left side of Figure 3. Next, the node representations are updated by sum pooling:

$$\begin{aligned} e_i^{l+1} &= e_{i,r}^l + e_{i,c}^l + e_i^l \\ e_u^{l+1} &= e_{u,r}^l + e_u^l \\ e_c^{l+1} &= e_{c,i}^l + e_{c,c}^l + e_c^l \end{aligned} \tag{8}$$

where l is the l-th layer of the heterogeneous hypergraph. On the interaction hyperedge $\mathcal{E}^i$, the review node bridges user preferences and item features to enrich the semantics of the collaboration. On the feature hyperedge $\mathcal{E}^f$, we aggregate the embeddings of items that belong to the same category and further enhance the item representation through feature-based co-occurrence.

Finally, as different levels of node embeddings reflect varying fusion degrees of heterogeneous information, we synthesize node representations through average pooling:

$$h_u = \sum_{l=1}^{L} e_u^l \tag{9}$$

$$h_i = \sum_{l=1}^{L} e_i^l \tag{10}$$

where L is the total number of layers of the heterogeneous hypergraph aggregation.

Joint Training

By leveraging the category and review semantics provided by LLMs and the graph autoencoder, we optimize the collaboration and co-occurrence relationships along the heterogeneous hypergraph to obtain semantic enhanced user and item representations h_u and h_i. Our model can be jointly trained by BPR loss and reconstruction loss as follows:

$$\mathcal{L} = \mathcal{L}_{\mathrm{BPR}} + \lambda_1 \cdot \mathcal{L}_{\mathrm{recon}} + \lambda_2 \cdot ||\Theta||_2^2 \tag{11}$$

where λ_1 and λ_2 are the trade-off parameters to adjust the magnitude of reconstruction and regularization.

Experiment

Experimental Settings

Dataset and Preprocessing We evaluate SEHHN on three real-world datasets:

Dataset	#user	#item	#category	#review
Book	11,000	9,332	1,058	120,464
Yelp	11,091	11,010	667	166,620
Steam	23,310	5,237	338	316,190

Table 1: Statistics of the datasets after preprocessing, where the number of reviews is equal to the number of interactions.

- *Book* is a subset of the widely used dataset *Amazon* for assessing recommendation systems, which includes users' interactions with reviews and the metadata of books from Amazon.com (McAuley et al. 2015).
- *Yelp* is a business dataset with user feedback and business attributes.
- *Steam* contains the user's purchasing history and comments on electronic games on the Steam platform (Kang and McAuley 2018).

We perform data preprocessing following the similar settings in (Ren et al. 2024; Wang et al. 2019b). Each interaction is divided into three portions with a ratio of 6:2:2, which are used for training, validation, and testing separately. We encode reviews within training interactions only while inferring categories for every item. The statistics of the datasets are summarized in Table 1.

Evaluation Metrics We adopt two widely used metrics, Recall@k and MRR@k, where the former represents the proportion of ground truth items in the top$k$ recommendation lists and the latter considers the reciprocal ranking of the target items so that the recommendation order can be observed. In this study, we report the results for $k = 10, 20$.

Implementation Details We evaluate all baselines using the SSLRec framework (Ren et al. 2023a). We optimize models with Adam optimizer with the learning rate 1e-3, the dimension of collaborative embeddings d is set to 32, and the mini-batch size is set to 4096. The trade-off parameters are set λ_1, λ_2 to $\{$5e-2, 1e-6$\}$. The sampling probability p_g of graph autoencoder is set to 0.05, and the reconstruction ratio p_s is set to 1e-3. We search for the optimal number of GNN layers for each dataset. With the time complexity of $O(|\mathcal{R}| \times max(|\mathcal{U}|, |\mathcal{V}|))$ and the space complexity of $O(|\mathcal{R}| \times d)$, we evaluate SEHHN on a single RTX3090.

To ensure a fair comparison, We use OpenAI API to implement item classifier by GPT-3.5-turbo and encoding review texts by text-embedding-ada-002 (Neelakantan et al. 2022). Note that the LLM for semantic representation and classification can be easily replaced with other open-source models without fine-tuning, such as the embedding model BGE-large (Xiao et al. 2023) and the chat model LLaMA2 (Touvron et al. 2023). The dimension of semantic embeddings d_s is 1536 according to the LLM.

Baselines To evaluate the performance of the proposed model SEHHN, we choose the following competitive methods as baselines:

Method	Book				Yelp				Steam			
	R@10	M@10	R@20	M@20	R@10	M@10	R@20	M@20	R@10	M@10	R@20	M@20
GCCF	0.0876	0.1128	0.1346	0.1246	0.0640	0.1022	0.1064	0.1162	0.0823	0.1244	0.1314	0.1392
LightGCN	0.0871	0.1158	0.1368	0.1284	0.0691	0.1125	0.1144	0.1272	0.0863	0.1294	0.1351	0.1440
SGL	0.1023	0.1318	0.1480	0.1441	0.0744	0.1197	0.1218	0.1354	0.0911	0.1379	0.1425	0.1534
SimGCL	0.0982	0.1274	0.1505	0.1410	0.0755	0.1237	0.1252	0.1399	<u>0.0930</u>	<u>0.1396</u>	<u>0.1450</u>	<u>0.1552</u>
DCCF	0.1013	0.1315	0.1489	0.1443	0.0784	0.1270	0.1249	0.1428	0.0928	0.1388	0.1444	0.1542
AutoCF	0.1053	0.1451	0.1561	0.1585	0.0801	0.1268	0.1295	0.1432	0.0853	0.1298	0.1357	0.1450
RLMRec	<u>0.1091</u>	<u>0.1473</u>	<u>0.1608</u>	<u>0.1609</u>	<u>0.0812</u>	<u>0.1317</u>	<u>0.1326</u>	<u>0.1468</u>	0.0891	0.1349	0.1390	0.1499
SEHHN	**0.1132**	**0.1486**	**0.1677**	**0.1630**	**0.0836**	**0.1345**	**0.1360**	**0.1518**	**0.0965**	**0.1448**	**0.1512**	**0.1613**

Table 2: The performance of SEHHN with other baselines, where R and M refer to Recall and MRR, respectively. Methods with the best and second-best performance are denoted in bold and underlined fonts respectively. Each result is averaged of five experiments, and all improvements are significant with p-value < 0.05 based on paired t-tests.

- **GCCF** (Chen et al. 2020): It is a graph convolution network for collaborative filtering that removes non-linear activation functions.
- **LightGCN** (He et al. 2020): A widely used baseline that implements recommendations using lightweight convolutional frameworks without redundant parameters.
- **SGL** (Wu et al. 2021): It introduces self-supervised learning into CF frameworks, which augments subgraph structure to enhance the model's noise tolerance.
- **SimGCL** (Yu et al. 2022): It applies uniform noise to the embedding space to generate contrastive views, thus enhancing the robustness of representations.
- **DCCF** (Ren et al. 2023b): It utilizes self-supervised augmentation by intent disentanglement to distill finer-grained user preferences.
- **AutoCF** (Xia et al. 2023): It adaptively generates self-supervised signals by reconstructing masked graphs for data augmentation.
- **RLMRec** (Ren et al. 2024): It is a semantic enhancement module based on LLMs, which generates profiles to enhance the representation of users and items. We choose AutoCF_plus as the baseline since it performs the best on average across all datasets.

Overall Performance

Table 2 reports the performance comparison between baselines and our proposed SEHHN on three real-world datasets. We can observe the following results:

For previous GNN-based methods, LightGCN outperformed GCCF in all datasets, indicating the effectiveness of the lightweight graph convolutional network. By introducing self-supervised learning, SGL and SimGCL achieved superior performance, which can be attributed to alleviating sparsity and noise in interactions through a contrastive learning paradigm. In addition, DCCF and AutoCF further enhance the self-supervised signal through disentangled user intent and graph mask autoencoder, respectively. Subsequently, RLMRec proposed to enhance collaborative representation using the semantic information provided by LLMs. In particular, RLMRec introduced additional semantics into user and item embeddings by maximizing mutual information, resulting in greater performance.

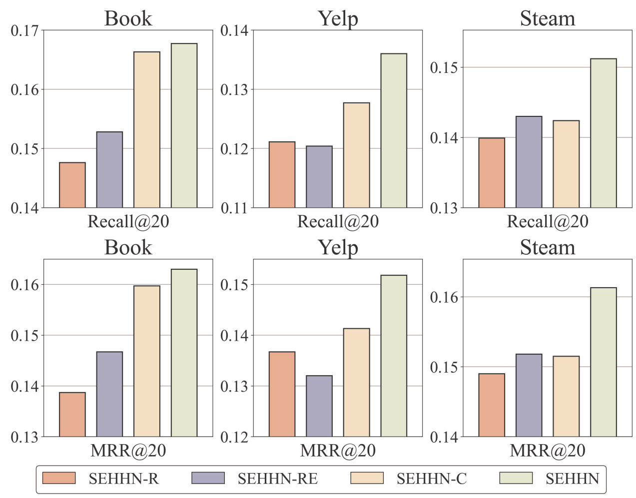

Figure 4: Ablation study on LLM-extracted semantics.

Finally, our proposed SEHHN surpassed the SOTA methods on all datasets. The improvement can be attributed to the use of semantic information to enhance the relationship modeling in CFs rather than directly augmenting entity representation, avoiding the degraded alignment caused by inconsistent semantics. The proposed graph autoencoder and LLM-based classifier leverage semantic information to enrich the collaboration and co-occurrence relationship in CFs, and better utilize the semantic information provided by LLM under the guidance of heterogeneous hypergraph.

Ablation Study

The Effect of the LLM-extracted Semantics As demonstrated in the previous sections, we utilize LLMs to extract review semantics and infer item categories for CF models. In this section, we explore the impact of the additional knowledge provided by LLMs on recommendation performance. SEHHN-R and SEHHN-C represent the versions of SEHHN without utilizing reviews and categories, re-

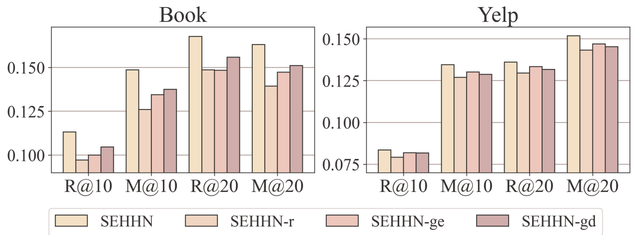

Figure 5: Ablation study on graph autoencoder.

spectively; SEHHN-RE replaces the review embeddings obtained through LLM with randomly initialized embeddings of the same dimension.

As shown in Figure 4, the performance of SEHHN-R and SEHHN-C decreases compared to the original model, indicating that the semantics of both review and category nodes are crucial for capturing user preferences and item features. The former can supplement contextual semantics for user-item collaboration, while the latter helps analyze the co-occurrence attributes of items. In addition, the performance of SEHHN-RE is inferior to that of SEHHN, demonstrating that LLMs can provide semantic representations that are beneficial for recommendation.

The Effect of Graph AutoEncoder We achieve alignment of review semantics to the collaborative space through a graph autoencoder. To verify its effectiveness, we design three variants: SEHHN-r removes the reconstruction task in the original model; SEHHN-ge replaces the graph encoder with linear transformation, which does not use observation or preference hyperedges to capture relationships between review nodes, i.e., replacing equations (3-4) with $e_r = \mathrm{MLP}_1(s)$; SEHHN-de replaces the graph decoder and directly reconstructs the review semantics with $\tilde{s} = \mathrm{MLP}_2(e_r)$.

Figure 5 reports the results on *Book* and *Yelp* datasets. We can observe that removing graph-level modeling from the encoder or decoder will lead to decreased performance, which suggests that leveraging the collaborative relationships on both sides of review nodes can more effectively bridge the gap between the semantic space and the collaboration space.

The Effect of LLM-based Embedder We extract high-dimensional semantic representations from reviews through LLMs. Building on this, we further experiment with open-source text embedding models BGE-large (Xiao et al. 2023) and Bloom-1b1 (Le Scao et al. 2023). As shown in Table 3, SEHHN can leverage the semantic information provided by any LLM to improve the recommendation performance, which demonstrates the generalization ability of the proposed heterogeneous hypergraph and graph autoencoder.

Hyper-parameter Study

Depth of GNN layers The number of GNN Layers L determines the degree of information fusion between heterogeneous nodes. We conduct experiments to evaluate the performance of varying numbers of layers on two datasets, as shown in Figure 6.

LLM	default	BGE	Bloom	AutoCF
Recall@20	0.1351	0.1322	0.1316	0.1295
MRR@20	0.1500	0.1484	0.1474	0.1432

Table 3: Experiments of LLM-based embedder on Yelp.

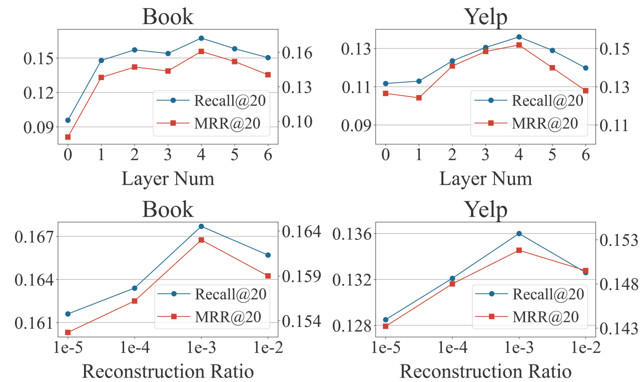

Figure 6: Impact of the number of GNN layers and the reconstruction ratio.

First, the absence of aggregation ($L = 0$) leads to the worst performance, highlighting the significance of message passing in fusing heterogeneous information. As L increases, the model's performance improves by incorporating neighboring semantics to a greater extent. However, too many layers might result in over-smooth representations and reducing recommendation performance.

Reconstruction Ratio The reconstruction ratio p_s determines the number of review nodes in the reconstruction task. Specifically, a higher probability means sampling more interaction hyperedges $\mathcal{E}^i$ to reconstruct the review semantics.

As shown in Figure 6, a lower p_s leads to significant performance degradation, indicating that insufficient learning of the graph encoder can result in poor alignment. Furthermore, excessive p_s may cause overfitting of the decoder, thus preventing the encoder from outputting semantic information that is valuable for collaborative filtering.

Conclusion

In this paper, we discuss the alignment degradation caused by semantic inconsistency when improving collaborative filtering with LLMs. To address these challenges, we propose a novel method named Semantic Enhanced Heterogeneous Hypergraph Network (SEHHN). The review semantics extracted by the graph autoencoder and feature-based item correlations mined by the LLM-based classifier enrich the collaboration and co-occurrence of CFs. In addition, we design a heterogeneous hypergraph network to achieve efficient alignment and propagation of heterogeneous information. Extensive experiments demonstrate the superiority of SEHHN and the effectiveness of each component.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant No. 62276110, No. 62172039 and in part by the fund of Joint Laboratory of HUST and Pingan Property & Casualty Research (HPL). The authors would also like to thank the anonymous reviewers for their comments on improving the quality of this paper.

References

Bao, K.; Zhang, J.; Wang, W.; Zhang, Y.; Yang, Z.; Luo, Y.; Feng, F.; He, X.; and Tian, Q. 2023a. A bi-step grounding paradigm for large language models in recommendation systems. *arXiv preprint arXiv:2308.08434*.

Bao, K.; Zhang, J.; Zhang, Y.; Wang, W.; Feng, F.; and He, X. 2023b. Tallrec: An effective and efficient tuning framework to align large language model with recommendation. *arXiv preprint arXiv:2305.00447*.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Chen, L.; Wu, L.; Hong, R.; Zhang, K.; and Wang, M. 2020. Revisiting graph based collaborative filtering: A linear residual graph convolutional network approach. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 27–34.

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*.

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

He, X.; Liao, L.; Zhang, H.; Nie, L.; Hu, X.; and Chua, T.-S. 2017. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*, 173–182.

Hou, Y.; Mu, S.; Zhao, W. X.; Li, Y.; Ding, B.; and Wen, J.-R. 2022a. Towards universal sequence representation learning for recommender systems. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 585–593.

Hou, Z.; Liu, X.; Cen, Y.; Dong, Y.; Yang, H.; Wang, C.; and Tang, J. 2022b. Graphmae: Self-supervised masked graph autoencoders. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 594–604.

Juan, W.; Yue-xin, L.; and Chun-ying, W. 2019. Survey of recommendation based on collaborative filtering. In *Journal of Physics: Conference Series*, volume 1314, 012078. IOP Publishing.

Kang, W.-C.; and McAuley, J. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, 197–206. IEEE.

Le Scao, T.; Fan, A.; Akiki, C.; Pavlick, E.; Ilić, S.; Hesslow, D.; Castagné, R.; Luccioni, A. S.; Yvon, F.; Gallé, M.; et al. 2023. Bloom: A 176b-parameter open-access multilingual language model.

Li, J.; Wang, M.; Li, J.; Fu, J.; Shen, X.; Shang, J.; and McAuley, J. 2023. Text Is All You Need: Learning Language Representations for Sequential Recommendation. *arXiv preprint arXiv:2305.13731*.

Lin, Z.; Tian, C.; Hou, Y.; and Zhao, W. X. 2022. Improving graph collaborative filtering with neighborhood-enriched contrastive learning. In *Proceedings of the ACM Web Conference 2022*, 2320–2329.

McAuley, J.; Targett, C.; Shi, Q.; and Van Den Hengel, A. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, 43–52.

Neelakantan, A.; Xu, T.; Puri, R.; Radford, A.; Han, J. M.; Tworek, J.; Yuan, Q.; Tezak, N.; Kim, J. W.; Hallacy, C.; et al. 2022. Text and code embeddings by contrastive pre-training. *arXiv preprint arXiv:2201.10005*.

Ren, X.; Wei, W.; Xia, L.; Su, L.; Cheng, S.; Wang, J.; Yin, D.; and Huang, C. 2024. Representation learning with large language models for recommendation. In *Proceedings of the ACM on Web Conference 2024*, 3464–3475.

Ren, X.; Xia, L.; Yang, Y.; Wei, W.; Wang, T.; Cai, X.; and Huang, C. 2023a. SSLRec: A Self-Supervised Learning Library for Recommendation. *arXiv preprint arXiv:2308.05697*.

Ren, X.; Xia, L.; Zhao, J.; Yin, D.; and Huang, C. 2023b. Disentangled Contrastive Collaborative Filtering. *arXiv preprint arXiv:2305.02759*.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y.; et al. 2017. Graph attention networks. *stat*, 1050(20): 10–48550.

Wang, L.; and Lim, E.-P. 2023. Zero-Shot Next-Item Recommendation using Large Pretrained Language Models. *arXiv preprint arXiv:2304.03153*.

Wang, X.; He, X.; Cao, Y.; Liu, M.; and Chua, T.-S. 2019a. Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 950–958.

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

Wang, Z.; Wei, W.; Cong, G.; Li, X.-L.; Mao, X.-L.; and Qiu, M. 2020. Global context enhanced graph neural networks for session-based recommendation. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, 169–178.

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

Xi, Y.; Liu, W.; Lin, J.; Cai, X.; Zhu, H.; Zhu, J.; Chen, B.; Tang, R.; Zhang, W.; Zhang, R.; et al. 2023. Towards open-world recommendation with knowledge augmentation from large language models. *arXiv preprint arXiv:2306.10933*.

Xia, L.; Huang, C.; Huang, C.; Lin, K.; Yu, T.; and Kao, B. 2023. Automated Self-Supervised Learning for Recommendation. In *Proceedings of the ACM Web Conference 2023*, 992–1002.

Xiao, S.; Liu, Z.; Zhang, P.; and Muennighoff, N. 2023. C-Pack: Packaged Resources To Advance General Chinese Embedding. arXiv:2309.07597.

Ye, Y.; Xia, L.; and Huang, C. 2023. Graph Masked Autoencoder for Sequential Recommendation. *arXiv preprint arXiv:2305.04619*.

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

Yue, Z.; Rabhi, S.; Moreira, G. d. S. P.; Wang, D.; and Oldridge, E. 2023. LlamaRec: Two-Stage Recommendation using Large Language Models for Ranking. *arXiv preprint arXiv:2311.02089*.

Zhang, F.; Yuan, N. J.; Lian, D.; Xie, X.; and Ma, W.-Y. 2016. Collaborative knowledge base embedding for recommender systems. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 353–362.

Zhang, Y.; Feng, F.; Zhang, J.; Bao, K.; Wang, Q.; and He, X. 2023. Collm: Integrating collaborative embeddings into large language models for recommendation. *arXiv preprint arXiv:2310.19488*.

Zhao, S.; Wei, W.; Mao, X.-L.; Zhu, S.; Yang, M.; Wen, Z.; Chen, D.; and Zhu, F. 2023a. Multi-view hypergraph contrastive policy learning for conversational recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 654–664.

Zhao, W. X.; Zhou, K.; Li, J.; Tang, T.; Wang, X.; Hou, Y.; Min, Y.; Zhang, B.; Zhang, J.; Dong, Z.; et al. 2023b. A survey of large language models. *arXiv preprint arXiv:2303.18223*.