Enhancing Recommender Systems via Fusion of Large Language Models and Heterogeneous Hypergraphs for Multi-view Contrastive Learning

1st Di Luo
Guangdong University of Foreign Studies
Guangzhou, China
dwyaneluo@gmail.com

2nd Fenfang Xie*
Guangdong University of Foreign Studies
Guangzhou, China
xiefragrance@gmail.com

3rd Feng Luo
Guangdong University of Foreign Studies
Guangzhou, China
feng72013@gmail.com

4th Mingdong Tang
Guangdong University of Foreign Studies
Guangzhou, China
mdtang@gdufs.edu.cn

5th Xiaohua Han
Guangdong University of Foreign Studies
Guangzhou, China
Hanxiaohua@gdufs.edu.cn

6th Liang Chen
Sun Yat-sen University
Guangzhou, China
chenliang6@mail.sysu.edu.cn

***Abstract*—Recommender systems are essential for personalized content filtering, yet existing approaches based on large language models (LLMs), graph neural networks, and contrastive learning still face key challenges: (1) over-smoothing in deep propagation, (2) sensitivity to noisy and sparse interactions, and (3) incomplete integration of textual and structural signals. To overcome these issues, we propose LLMHRec, a unified framework that integrates LLM-enhanced semantics with heterogeneous hypergraph learning for robust multi-view representation. Specifically, LLMs are employed to extract fine-grained semantic embeddings from reviews, while a heterogeneous hypergraph models user–item–review relations. A hyperedge attention mechanism derives a denoised hypergraph, and both original and denoised structures are encoded with LightHGCN. Their embeddings are fused via gated attention, and further aligned with semantic embeddings through a co-attention module. Finally, we apply a multi-view contrastive learning strategy on the fused embeddings, alongside augmented views created by random edge dropout and noise injection, to enhance representation robustness and generalization. Experiments on two real-world datasets show that LLMHRec consistently outperforms state-of-the-art baselines, validating its effectiveness and robustness.**

***Index Terms*—Large Language Models, Heterogeneous Hypergraph, Co-Attention, Contrastive Learning, Recommender Systems**

I. Introduction

Recommender systems have become indispensable for personalized information filtering, enabling the identification of user preferences from behavioral data and the delivery of tailored content. They play a pivotal role across diverse domains, including e-commerce [1], social media [2], music streaming [3], [4], and live broadcasting platforms [5]. Among various techniques, Collaborative Filtering (CF) remains a cornerstone due to its ability to exploit user–user and item–item similarity patterns [6]. However, its performance is severely hindered by the data sparsity problem, where limited user–item interactions fail to provide sufficient collaborative signals, ultimately degrading recommendation accuracy.

In recent years, Graph Neural Networks (GNNs) have significantly advanced recommendation by modeling user–item interactions through neighborhood aggregation. Representative methods include GraphSAGE [7], which inductively aggregates feature information from sampled neighbors. NGCF [8] extends this idea by explicitly propagating embeddings over high-order connectivity, while LightGCN [9] further simplifies the architecture by discarding feature transformations and nonlinearities, focusing solely on neighborhood aggregation for improved efficiency and effectiveness. However, bipartite GNNs are limited in modeling higher-order group-wise relations beyond pairwise interactions. To address this, Hypergraph Neural Networks (HGNNs) [10] connect multiple nodes via hyperedges to capture richer semantics. Extensions such as DHCF [11] introduce dual channels to separately model user- and item-centric signals, while HHGNN [12] incorporates heterogeneous hypergraphs with type-specific attention to learn more expressive representations.

Meanwhile, large language models (LLMs) have achieved remarkable success in natural language processing [13] and are increasingly applied to recommender systems for modeling unstructured textual data. User reviews and product descriptions provide rich semantic cues about user preferences and item properties, and encoding them with LLMs yields expressive embeddings that complement interaction signals. Early work such as BERT4Rec [14] exploits bidirectional transformer modeling to capture user behavior sequences, while CL4SRec [15] introduces contrastive learning with sequence augmentations for robustness. More recently, LLMRG [16] leverages LLMs to construct personalized reasoning graphs that integrate user profiles and behavioral sequences for enhanced interpretability and recommendation quality.

Contrastive learning (CL), a key self-supervised paradigm, has also been widely applied in recommender systems to enforce representation consistency across multiple graph views while preserving structural semantics [17]. Unlike supervised methods, graph contrastive learning (GCL) leverages augmented graph views to capture latent relations. Representative works include SGL [18], which generates augmentations for contrastive learning; SimGCL [19], which adopts a simplified strategy for robustness; and MVGCL [20], which constructs local and global diffusion views to enhance message passing and mitigate sparsity and noise.

Although the aforementioned methods have advanced recommendation performance, they still suffer from one or more of the following limitations:

- **Over-Smoothing in Deep Representation Propagation.** GNNs aggregate neighbor information to enrich node embeddings, but deeper layers often cause over-smoothing, making embeddings less distinguishable. In contrastive learning, multi-view augmentations can worsen this by discarding informative features, reducing inter-view variability and weakening the contrastive signal.
- **Sensitivity to Noisy and Sparse Interactions.** Sparse or noisy user–item interactions degrade embedding quality as errors propagate through message passing. While contrastive learning partially alleviates sparsity, most methods lack mechanisms for robust, noise-resistant representations and uniform latent spaces.
- **Incomplete Integration of Textual and Interaction Signals.** Existing GNN- and CL-based approaches mainly model interaction structures but underutilize textual content like reviews, which carry fine-grained semantic cues. LLMs capture rich semantics but struggle with complex relational patterns, leaving a gap between semantic and structural representations.

To address these challenges, we propose LLMHRec, a unified framework that integrates large language models with heterogeneous hypergraph contrastive learning to improve robustness and generalization. It extracts semantic embeddings from user reviews via LLMs, while a user–item–review heterogeneous hypergraph captures high-order interactions. To mitigate sparsity and noise, edge attention is used to prune low-confidence links, producing a denoised heterogeneous hypergraph. Both original and denoised heterogeneous hypergraphs are encoded with LightHGCNs, and their embeddings are fused through a mutil-head attention module. A co-attention mechanism then aligns interaction embeddings with semantic embeddings, yielding unified user and item representations. Finally, multi-view contrastive learning applied to fused embeddings and augmented graph views via edge dropout and noise injection enforces invariance across views, enhancing generalization. Extensive experiments on two real-world datasets show that LLMHRec outperforms strong baselines, with ablation confirming the effectiveness of its components.

The main contributions are summarized as follows:

- We present LLMHRec, which integrates LLM-derived textual semantics with heterogeneous hypergraph modeling of user–item–review relations. To further enhance representation learning, it employs a co-attention fusion mechanism that dynamically aligns and fuses interaction representations with semantic embeddings, thereby capturing complementary signals across different views.
- We apply a multi-view contrastive learning framework with augmented data views to enforce embedding consistency and improve model generalization.
- Extensive experiments demonstrate that our approach outperforms baseline models across multiple datasets, especially in scenarios with sparse interaction data. This highlights the model's capacity to preserve recommendation accuracy by integrating semantic information with structural insights, thereby providing a robust and personalized recommendation solution.

II. PRELIMINARIES

This section provides an overview of the key concepts forming the foundation of our proposed **LLMHRec** framework, including the implicit interaction matrix, LLM-based textual embeddings, and the heterogeneous user–item–review hypergraph.

Definition 1: Implicit Interaction Matrix. Let $U = \{u_1, u_2, \ldots, u_M\}$ and $I = \{i_1, i_2, \ldots, i_N\}$ represent the set of users and items, respectively, where $|U| = M$ and $|I| = N$. We denote the implicit feedback by $\mathbf{R} \in \{0,1\}^{M\times N}$, where each element is defined as:

$$r_{u,i} = \begin{cases} 1, & \text{if user } u \text{ has interacted with item } i; \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

where 1 indicates that user u has interacted with item i, and 0 indicates no such interaction.

Definition 2: Heterogeneous User–Item–Review Hypergraph. Unlike traditional pairwise graphs, a heterogeneous hypergraph can connect multiple nodes of different types in a single hyperedge. We define $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the vertex set $\mathcal{V} = \{U \cup I \cup \mathcal{R}\}$ includes users, items, and review nodes, and $\mathcal{E}$ is the set of hyperedges. Each hyperedge $e \in \mathcal{E}$ can connect a subset of nodes, e.g., $e = \{u, i, r\}$, representing a user–item interaction and its associated review. The vertex–hyperedge relationships are represented by an incidence matrix $H \in \{0,1\}^{|\mathcal{V}|\times|\mathcal{E}|}$, where:

$$h(v, e) = \begin{cases} 1, & \text{if } v \in e; \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

where 1 denotes a connection between vertex v and hyperedge e, while 0 indicates no connection.

III. METHODOLOGY

In this section, we present our LLMHRec framework, whose overall architecture is shown in Figure 1. First, we employ an LLM-based semantic encoder to extract semantic representations from user and item reviews. Second, we design a heterogeneous hypergraph encoder to capture high-order user–item interactions and learn interaction-aware embeddings. Third, a cross-view co-attention module integrates semantic and structural information. Finally, a multi-view contrastive learning strategy is introduced to enhance robustness by aligning perturbed views.

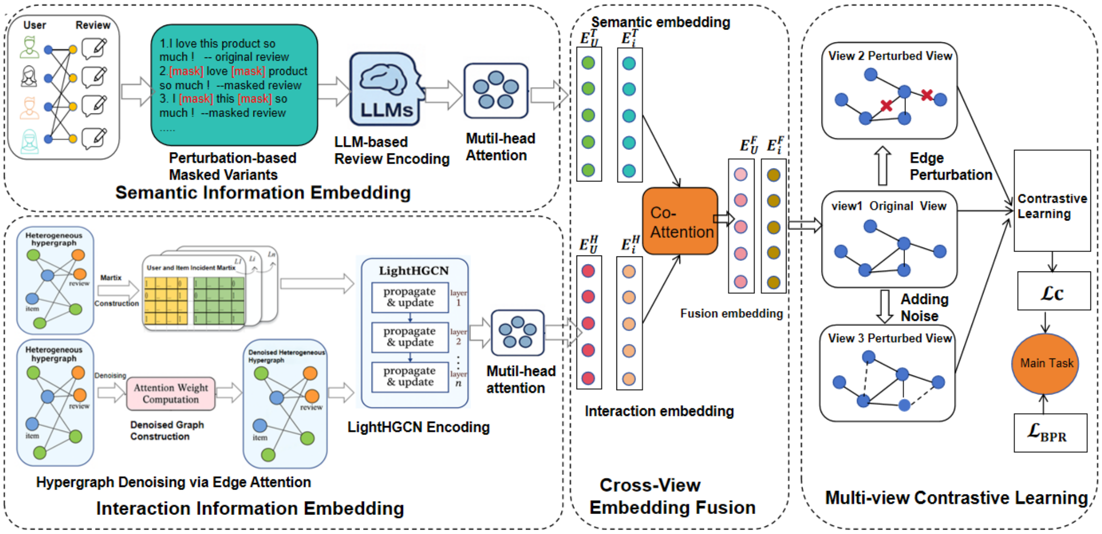

Fig. 1. Overall framework of the proposed **LLMHRec** model.

A. Semantic Information Embedding

User reviews provide rich semantic cues that reflect both fine-grained user preferences and latent item characteristics. To fully exploit these signals, we design a multi-view semantic embedding module that leverages a pre-trained large language model (LLM) combined with perturbation-based augmentation and multi-head attention fusion.

1) Perturbation-aware LLM Review Encoding: Given a review text $r = \{w_1, w_2, \ldots, w_n\}$ consisting of n tokens, we aim to generate robust token-level representations that capture both the original semantics and potential perturbations.

We first construct a set of perturbed variants using a stochastic masking operator $\mathcal{M}(\cdot)$, which selectively masks salient content tokens while preserving syntactic structure:

$$\tilde{R} = \{r, r^{(1)}, r^{(2)}, \ldots, r^{(m)}\}, \quad r^{(j)} \sim \mathcal{M}(r),\ j = 1, \ldots, m, \quad (1)$$

where m denotes the number of perturbed views, and $r^{(j)}$ represents the j-th variant of the original review r sampled from the masking distribution $\mathcal{M}(r)$. This formulation ensures that both the original review and its masked counterparts are considered jointly.

Next, the entire set $\tilde{R}$ is encoded by a pre-trained large language model (LLM) encoder $\mathcal{E}_{\text{LLM}}$ to obtain token-level contextualized representations:

$$H^{(j)} = \mathcal{E}_{\text{LLM}}(r^{(j)}), \quad (2)$$

$$H^{(j)} = \{h_1^{(j)}, h_2^{(j)}, \ldots, h_n^{(j)}\} \in R^{n\times d}, \quad j = 0, \ldots, m. \quad (3)$$

where d is the hidden dimension and $H^{(0)}$ corresponds to the original review.

Finally, we aggregate the original and perturbed representations to construct a multi-view semantic representation set:

$$\mathcal{H} = \{H^{(0)}, H^{(1)}, \ldots, H^{(m)}\}. \quad (4)$$

This design allows the model to jointly learn from both intact and perturbed semantics, enhancing robustness and generalization of review representations. Optionally, multi-head attention can be applied over $\mathcal{H}$ to fuse information across views for richer semantic embedding.

2) Multi-Head Attention Fusion: The multiple semantic views $\mathcal{H}$ are then integrated via a multi-head attention fusion mechanism to capture complementary semantic patterns from different perturbations. For a given view $H^{(v)}$, the attention for each head is computed as:

$$Q = H^{(v)}W_Q, \quad K = H^{(v)}W_K, \quad V = H^{(v)}W_V, \quad (5)$$

$$A = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right), \quad O^{(v)} = AV, \quad (6)$$

where $W_Q, W_K, W_V \in R^{d\times d_k}$ are learnable projection matrices. Q, K, and V denote the query, key, and value representations projected from the same input $H^{(v)}$, which allow the model to measure relevance (via $QK^\top$) and aggregate information (via AV) across different tokens in the view. Outputs from h heads are concatenated and linearly projected:

$$z^{(v)} = \text{Concat}(O_1^{(v)}, O_2^{(v)}, \ldots, O_h^{(v)})W_O. \quad (7)$$

Finally, the view-level representations $\{z^{(1)}, z^{(2)}, \ldots, z^{(m)}\}$ are averaged to obtain the textual embedding:

$$e_{\text{text}} = \frac{1}{m}\sum_{v=1}^{m} z^{(v)}. \quad (8)$$

We then aggregate these review-level embeddings separately for users and items. Let $\mathcal{R}_u$ and $\mathcal{R}_i$ denote the sets of reviews authored by user u and associated with item i, respectively. The final user- and item-level textual embeddings are given by:

$$E_{u/i}^T = \frac{1}{|\mathcal{R}_{u/i}|}\sum_{r\in\mathcal{R}_{u/i}} e_{\text{text}}(r), \quad (9)$$

where $E_u^T, E_i^T \in R^d$ encode the semantic preferences of user u and the latent textual characteristics of item i, respectively. These embeddings, enriched with perturbation-invariant semantic cues, are subsequently integrated with interaction-based representations via our multi-view fusion and contrastive learning framework.

B. Interaction Information Embedding

To capture high-order collaborative signals and complex semantic relationships between users, items, and reviews, we construct a heterogeneous hypergraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the node set $\mathcal{V}$ contains three types of nodes: users $\mathcal{U}$, items $\mathcal{I}$, and reviews $\mathcal{R}$, and the hyperedge set $\mathcal{E}$ models their interactions. Each hyperedge connects one user, one item, and its associated review, thus preserving triadic relations in the data.

1) Hypergraph Denoising via Edge Attention: Given the adjacency tensor of hypergraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, we first estimate the reliability of each hyperedge using an attention mechanism:

$$\alpha_e = \frac{\exp\left(\mathbf{a}^\top \sigma\left(W_\alpha \mathbf{h}_e\right)\right)}{\sum_{e' \in \mathcal{E}} \exp\left(\mathbf{a}^\top \sigma\left(W_\alpha \mathbf{h}_{e'}\right)\right)}, \quad (10)$$

where $\mathbf{h}_e$ denotes the concatenated embedding of nodes in hyperedge e, W_α and $\mathbf{a}$ are learnable parameters, and $\sigma(\cdot)$ is a non-linear activation.

Subsequently, hyperedges with attention scores below a threshold τ are removed to obtain a denoised hypergraph:

$$\tilde{\mathcal{E}} = \{e \in \mathcal{E} \mid \alpha_e \geq \tau\}, \quad \tilde{\mathcal{G}} = (\mathcal{V}, \tilde{\mathcal{E}}). \quad (11)$$

2) LightHGCN Encoding: Traditional Hypergraph Convolutional Networks (HGCNs) rely on nonlinear activations and feature transformations at each layer, which increase complexity and risk overfitting. Inspired by the design philosophy of LightGCN [9], we simplify hypergraph convolution by removing these components and retaining only pure neighborhood aggregation. This design reduces computation, mitigates over-smoothing, and better models high-order connectivity in large-scale heterogeneous hypergraphs. Hence, we adopt Light Hypergraph Convolutional Networks (LightHGCNs) to encode both the original and denoised hypergraphs. For a given hypergraph $\mathcal{G}$, the propagation rule is:

$$E^{(l+1)} = \tilde{D}_v^{-\frac{1}{2}} \tilde{H} \tilde{D}_e^{-1} \tilde{H}^\top \tilde{D}_v^{-\frac{1}{2}} E^{(l)}, \quad (12)$$

where $\tilde{H}$ is the incidence matrix, $\tilde{D}_v$ and $\tilde{D}_e$ are diagonal degree matrices of nodes and hyperedges, and $\mathbf{E}^{(l)}$ denotes node embeddings at layer l. The final embedding is obtained by averaging over all L layers:

$$E_{\mathcal{G}} = \frac{1}{L+1} \sum_{l=0}^{L} E^{(l)}. \quad (13)$$

The same procedure is applied to $\mathcal{G}$ and $\tilde{\mathcal{G}}$, resulting in $E_{\mathcal{G}}$ and $E_{\tilde{\mathcal{G}}}$.

3) User- and Item-level Interaction Embeddings: To obtain interaction-aware embeddings for users and items, we first integrate structural signals from both the original and denoised heterogeneous hypergraphs using a multi-head attention mechanism. For each attention head $h = 1, \ldots, H$, we compute:

$$Q_h = E_{\mathcal{G}} W_Q^{(h)}, \quad K_h = E_{\tilde{\mathcal{G}}} W_K^{(h)}, \quad V_h = E_{\tilde{\mathcal{G}}} W_V^{(h)}, \quad (14)$$

$$A_h = \text{softmax}\left(\frac{Q_h K_h^\top}{\sqrt{d_k}}\right), \quad O_h = A_h V_h, \quad (15)$$

where d_k is the dimension of each head. Q_h, K_h, V_h are the query, key, and value representations projected through the head-specific parameters $W_Q^{(h)}, W_K^{(h)}, W_V^{(h)}$. The outputs of all heads are then aggregated to obtain node-level interaction embeddings:

$$Z_v = \frac{1}{H} \sum_{h=1}^{H} O_h[v], \quad v \in \mathcal{U} \cup \mathcal{I}, \quad (16)$$

where $O_h[v]$ denotes the row corresponding to node v in O_h.

Finally, to generate user- and item-specific embeddings, we apply an attention-based aggregation over the multi-head outputs:

$$E_{u/i}^I = \sum_{h=1}^{H} \alpha_h^{(u/i)} O_h[u/i], \quad u/i \in \mathcal{U} \cup \mathcal{I}, \quad (17)$$

where the attention weights $\alpha_h^{(v)}$ are computed as

$$\alpha_h^{(v)} = \frac{\exp(a^\top \tanh(W O_h[v]))}{\sum_{k=1}^{H} \exp(a^\top \tanh(W O_k[v]))}. \quad (18)$$

These embeddings E_u^I and E_i^I capture interaction-aware structural information from the heterogeneous hypergraph and can be fused with the textual embeddings E_u^T, E_i^T in the subsequent co-attention module.

4) Cross-View Embedding Fusion: To capture the complementary signals between textual semantics and interaction structures, we employ a co-attention mechanism to jointly model user/item embeddings from the two modalities. Given the textual embeddings E_u^T, E_i^T and interaction embeddings E_u^I, E_i^I, we first compute the attention weights as:

$$\alpha_u = \text{softmax}\Big((E_u^T W_Q)(E_u^I W_K)^\top / \sqrt{d}\Big), \quad (19)$$

$$\alpha_i = \text{softmax}\Big((E_i^T W_Q)(E_i^I W_K)^\top / \sqrt{d}\Big), \quad (20)$$

where $W_Q, W_K \in R^{d \times d}$ are learnable projection matrices.

The attended representations are then derived by weighting the value projections:

$$\tilde{E}_u = \alpha_u (E_u^I W_V), \quad (21)$$

$$\tilde{E}_i = \alpha_i (E_i^I W_V), \quad (22)$$

where $W_V \in R^{d \times d}$ is a value projection.

Finally, the unified user and item embeddings are obtained by fusing the textual and interaction-aware signals:

$$E_u^F = \tanh(W_f [E_u^T \parallel \tilde{E}_u]), \quad (23)$$

$$E_i^F = \tanh(W_f [E_i^T \parallel \tilde{E}_i]), \quad (24)$$

where $W_f \in R^{2d \times d}$ is a fusion projection, and $\parallel$ denotes concatenation. Specifically, the concatenation of the textual and interaction embeddings produces a $2d$-dimensional vector, which is subsequently projected back into the d-dimensional space through W_f. In this way, the unified embeddings $E_u^F, E_i^F \in R^d$ are obtained and used for multi-view contrastive learning.

C. Multi-view Contrastive Learning

To further improve the robustness and generalization of fused embeddings E_u^F and E_i^F, we design a multi-view contrastive learning framework that leverages structural and stochastic perturbations. Specifically, we construct three complementary views: (i) the original fused embedding view V_1, (ii) an edge-dropped view V_2 obtained by randomly removing a subset of hyperedges from the interaction graph, and (iii) a noise-perturbed view V_3 generated by injecting Gaussian perturbations into the embedding space. Formally,

$$V_2 = \tilde{\Theta}^{(\mathrm{drop})} E^F, \qquad V_3 = E^F + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma^2 I), \tag{25}$$

where $\tilde{\Theta}^{(\mathrm{drop})}$ is the propagation operator under random edge dropout, and $\boldsymbol{\epsilon}$ denotes i.i.d. Gaussian perturbations with variance σ^2. These perturbations generate alternative but semantically consistent embedding views $\{E^{F,v1}, E^{F,v2}, E^{F,v3}\}$.

To enforce invariance across views, we employ a contrastive objective that maximizes agreement between corresponding user–item pairs across different perturbed views. For a user–item pair (u, i), the loss between two views v_p and v_q is defined as:

$$\mathcal{L}_{pq} = -\sum_{(u,i)} \log \frac{\exp\left(\cos(E_u^{F,vp}, E_i^{F,vq})/\tau\right)}{\sum_j \exp\left(\cos(E_u^{F,vp}, E_j^{F,vq})/\tau\right)}, \tag{26}$$

where $p, q \in \{1, 2, 3\}$, and τ is a temperature coefficient controlling the sharpness of similarity distributions, and j enumerates all candidate negatives in the embedding space. This formulation aligns semantically consistent representations while repelling mismatched ones.

The overall multi-view contrastive loss is obtained by jointly optimizing across all view pairs:

$$\mathcal{L}_c = \rho\big(\mathcal{L}_{12} + \mathcal{L}_{13} + \mathcal{L}_{23}\big), \tag{27}$$

with ρ serving as a balancing hyperparameter. This design ensures that the learned embeddings remain stable under both structural perturbations and stochastic noise, thereby improving their generalization to unseen interactions.

Finally, we aggregate the representations across all views to form the final user and item embeddings:

$$E_u^* = \frac{1}{3}\sum_{k=1}^{3} \bar{E}_u^{F,k}, \qquad E_i^* = \frac{1}{3}\sum_{k=1}^{3} \bar{E}_i^{F,k}, \tag{28}$$

where $\bar{E}_u^{F,k}$ and $\bar{E}_i^{F,k}$ denote the pooled embeddings of user u and item i from view V_k, $K \in \{1, 2, 3\}$. The final interaction prediction is given by:

$$\hat{r}_{u,i} = {E_u^*}^\top E_i^* + b_u + b_i, \tag{29}$$

where b_u and b_i represent user and item-specific biases. This formulation captures both invariant representations across perturbations and personalized interaction tendencies.

D. Model Training

The Bayesian Pairwise Ranking (BPR) loss [21] serves as the primary loss function, utilizing interaction data to guide the optimization process. Specifically, for each observed user-item pair (u, i^+), a random item i^- with little interaction with u is sampled to construct a triplet (u, i^+, i^-):

$$\mathcal{L}_{BPR} = \sum_{(u,i^+,i^-)\in Q} -\log \sigma(\hat{r}_{u,i^+} - \hat{r}_{u,i^-}) \tag{30}$$

where $Q = \{(u, i^+, i^-) \mid (u, i^+) \in Q^+, (u, i^-) \in Q^-\}$ is the training data, Q^+ represents the positive sample set, which includes user u and item i pairs having interactions, Q^- represents the negative sample set, which includes user u and item i pairs without interactions, and σ is the sigmoid function.

Finally, the contrastive loss and the BPR loss are integrated into a unified objective function for training optimization:

$$\mathcal{L} = \mathcal{L}_{BPR} + \mathcal{L}_C \tag{31}$$

IV. EXPERIMENTS

This section provides a comprehensive overview of the experimental evaluation, guided by the following research questions:

- **RQ1:** How does LLMHRec perform compared with state-of-the-art recommendation baselines?
- **RQ2:** What are the contributions of the individual components in LLMHRec?
- **RQ3:** Is LLMHRec robust under varying levels of data sparsity?
- **RQ4:** Can LLMHRec effectively alleviate the over-smoothing issue of deep graph-based models?
- **RQ5:** How sensitive is LLMHRec to the choice of key hyperparameters?

We begin by detailing the experimental setup, including dataset statistics, evaluation metrics, and implementation configurations. We then conduct experiments corresponding to each research question.

TABLE I
STATISTICS OF THE EXPERIMENTAL DATASETS.

Datasets	User #	Item #	Interaction #	Density #
Amazon	37,275	17,869	54,742	8.22×10^{-5}
Yelp	43,204	29,018	192,504	1.54×10^{-4}

A. Experimental Settings

We conduct experiments on two benchmark datasets, Yelp and Amazon. Yelp contains user ratings, reviews, and business metadata in local service domains, while Amazon includes reviews, ratings, and product metadata across multiple categories. The detailed statistics are shown in Table I. The proposed LLMHRec model is implemented in PyTorch, with each dataset randomly partitioned into training, validation, and testing sets in an 8:1:1 ratio. All embeddings are initialized using Xavier initialization [22], and the batch size is fixed at 1024. The number of multi-heads in the co-attention module is set to 3. For hyperparameter tuning, the temperature parameter τ in the multi-view contrastive loss is searched over $\{0.07, 0.1, 0.3, 0.5, 0.8, 1\}$, while the weight coefficient ρ is selected from $\{0.05, 0.1, 0.2, 0.3, 0.5, 1\}$. The learning rate is set to 0.001 for the Amazon dataset and 0.0001 for the Yelp dataset. We employ the Adam optimizer [23] to facilitate efficient convergence. This systematic parameter tuning ensures a balanced trade-off between model complexity and generalization performance.

B. Baseline Models

The performance of the proposed LLMHRec method is thoroughly evaluated through a comprehensive comparison with several baseline approaches. We adopt Recall@N and Normalized Discounted Cumulative Gain (NDCG)@N as the metrics, which are widely used in recommendation. The specific details of these baseline methods are provided below:

Language Model Based Approaches.

- **BERT4Rec [14]:** It adapts the Transformer architecture for sequential recommendation by modeling user behavior as a bidirectional sequence.
- **LLM-Rec [24]:** It is a text-based recommendation model that leverages LLMs with diverse prompting strategies to enrich item descriptions and align them more effectively with user preferences.
- **GenRec [25]:** A generative recommendation framework that fine-tunes LLaMA with user–item interaction text and uses prompts to directly generate target items instead of ranking candidates.
- **RecSysLLM [26]:** It unifies LLMs' reasoning and domain knowledge through specially designed data, training, and inference pipelines for efficient recommendation.

Graph Neural Network Based Approaches.

- **NGCF [8]:** It models user–item interactions as a bipartite graph and propagates high-order collaborative signals by stacking multiple layers of graph convolutions.
- **LightGCN [9]:** It simplifies layer propagation by removing non-linear transformations and feature mapping.
- **HGCN [27]**: It extends GCN by using hypergraphs to capture complex relationships between entities for better performance on tasks.
- **HGRec [28]:** It is a heterogeneous graph neural network that improves node embeddings by aggregating multi-hop meta-path neighbors and fusing semantics through an attention mechanism.
- **WaveHDNN [29]:** It is a wavelet-enhanced hypergraph diffusion model that captures heterophilic and high-order user-item interactions using multi-scale structure encoding.

Multi-view Learning Based Approaches.

- **SimGCL [19]:** It creates contrastive views by adding uniform noise to the embeddings, simplifying the contrastive learning process, improving recommendation accuracy and training efficiency.
- **LightGCL [30]:** It is a lightweight graph contrastive learning approach, and it ensures computational efficiency and scalability while maintaining high performance.
- **HCCF [31]:** It utilizes hypergraph structures to model high-order interactions and employs contrastive learning objectives to improve the robustness of embeddings.
- **AMGCR [32]:** It introduces adaptive multi-view graph contrastive learning with four view generators and preference reconstruction to enhance robustness against noise.
- **HeLLM [25]:** It enhances LLM-based recommendation by integrating multimodal user/item hypergraphs with sequential user behaviors to capture higher-order relations and contextual signals.

C. RQ1:Overall Performance

To evaluate the effectiveness of the proposed LLMHRec model, we compare it with a wide range of state-of-the-art baselines, including language model-based, graph neural network-based, and multi-view contrastive learning-based approaches, on two real-world datasets (Amazon and Yelp). The results are summarized in Table II, and the following key observations can be made:

- **Overall Performance.** As shown in Table II, LLMHRec consistently achieves the best results across both datasets and all evaluation metrics. For instance, it improves Recall@15 by **8.76%** and NDCG@15 by **20.65%** over the strongest baseline HeLLM on Amazon, while yielding **15.37%** and **7.90%** gains on Yelp. These improvements can be attributed to three major components: i) the perturbation-aware semantic encoder that leverages LLMs to extract robust, fine-grained semantic cues from reviews; ii) the heterogeneous hypergraph encoder that captures high-order collaborative structures while mitigating noise through hyperedge attention and denoising; and iii) the cross-view co-attention fusion and multi-view contrastive learning framework, which effectively align and regularize textual and interaction signals for more generalizable representations.
- **Effectiveness of Heterogeneous Hypergraph Structure Modeling.** Hypergraph-based methods such as HGCN and HGRe coutperform traditional GNN models by better capturing high-order dependencies. Building upon this, LLMHRec constructs a heterogeneous hypergraph where users, items, and reviews are connected via triadic hyperedges to capture richer semantics and interactions. A hyperedge attention mechanism is further applied to derive a denoised hypergraph, and both graphs are jointly encoded with LightHGCN before being fused through attention, enabling robust and expressive representation learning. Compared with these hypergraph-based baselines, LLMHRec consistently delivers superior performance, demonstrating the benefit of combining heterogeneous hypergraph construction, noise-aware denoising, and efficient LightHGCN propagation.

TABLE II

OVERALL PERFORMANCE COMPARISON ON AMAZON AND YELP DATASETS. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD.

Baseline	Amazon						Yelp					
	R@5	R@10	R@15	N@5	N@10	N@15	R@5	R@10	R@15	N@5	N@10	N@15
Language Model Based Approaches												
BERT4Rec	0.0305	0.0427	0.0548	0.0243	0.0325	0.0391	0.0152	0.0219	0.0286	0.0119	0.0160	0.0195
LLM-Rec	0.0439	0.0603	0.0749	0.0387	0.0502	0.0589	0.0251	0.0358	0.0464	0.0199	0.0264	0.0327
GenRec	0.0433	0.0589	0.0731	0.0372	0.0491	0.0573	0.0264	0.0375	0.0486	0.0212	0.0279	0.0345
RecSysLLM	0.0478	0.0637	0.0798	0.0411	0.0539	0.0627	0.0276	0.0384	0.0502	0.0221	0.0286	0.0354
Graph Neural Network Based Approaches												
NGCF	0.0345	0.0487	0.0619	0.0282	0.0376	0.0451	0.0191	0.0273	0.0354	0.0153	0.0206	0.0252
LightGCN	0.0396	0.0549	0.0698	0.0327	0.0439	0.0528	0.0214	0.0305	0.0402	0.0172	0.0231	0.0284
HGCN	0.0417	0.0572	0.0724	0.0348	0.0461	0.0547	0.0236	0.0334	0.0439	0.0185	0.0248	0.0309
HGRec	0.0448	0.0605	0.0763	0.0376	0.0497	0.0586	0.0248	0.0352	0.0461	0.0194	0.0258	0.0317
WaveHDNN	0.0485	0.0645	0.0805	0.0419	0.0545	0.0634	0.0281	0.0390	0.0509	0.0226	0.0291	0.0361
Multi-view Learning Based Approaches												
SimGCL	0.0435	0.0596	0.0748	0.0371	0.0489	0.0575	0.0248	0.0351	0.0462	0.0196	0.0258	0.0321
LightGCL	0.0457	0.0617	0.0773	0.0385	0.0506	0.0592	0.0262	0.0367	0.0475	0.0209	0.0273	0.0337
HCCF	0.0473	0.0631	0.0792	0.0401	0.0521	0.0611	0.0271	0.0379	0.0491	0.0217	0.0281	0.0346
AMGCR	0.0489	0.0656	0.0817	0.0415	0.0538	0.0627	0.0284	0.0393	0.0506	0.0228	0.0294	0.0360
HeLLM	0.0512	0.0685	0.0857	0.0442	0.0576	0.0671	0.0301	0.0416	0.0532	0.0239	0.0310	0.0382
Ours												
LLMHRec	**0.0586**	**0.0742**	**0.0931**	**0.0512**	**0.0659**	**0.0782**	**0.0325**	**0.0472**	**0.0619**	**0.0256**	**0.0323**	**0.0396**
Gains (%)	14.45%	8.20%	8.76%	14.74%	14.75%	20.65%	7.87%	11.81%	15.37%	7.06%	4.33%	7.90%

- **Superiority of Multi-view Fusion and Contrastive Learning.** While LLM-based approaches (e.g., RecSysLLM, GenRec) excel at extracting textual semantics, they struggle to fully model collaborative signals. Conversely, graph-based approaches capture structural patterns but lack semantic expressiveness. LLMHRec bridges this gap through a co-attention based fusion of semantic and interaction embeddings, ensuring complementary signals are jointly captured. Moreover, the proposed multi-view contrastive learning, which introduces stochastic and structural perturbations, further enforces representation invariance and enhances generalization. As a result, LLMHRec achieves noticeable gains over both semantic-only and interaction-only baselines, highlighting the benefit of a unified multi-view representation learning framework.

D. RQ2:Ablation Study

To evaluate the contribution of each core component in LLMHRec, we conduct a comprehensive ablation study by designing several degraded variants of our model. Specifically, we consider the following versions:

- **LLMHRec$_{w/o\ RandMask}$**: The random masking strategy is removed, and the large language model directly encodes review texts.
- **LLMHRec$_{w/o\ LLM}$**: The entire LLM-based semantic encoder is discarded, and the model only relies on interaction embeddings.
- **LLMHRec$_{w/o\ Denoise}$**: The denoised hypergraph is excluded, and LightHGCN only operates on the raw heterogeneous hypergraph.
- **LLMHRec$_{w/o\ Interact}$**: The interaction embedding module is removed, and recommendations are made solely based on semantic embeddings.
- **LLMHRec$_{Concat}$**: The co-attention fusion module is replaced with a simple concatenation of semantic and interaction embeddings.
- **LLMHRec$_{MLP}$**: The co-attention module is substituted by a multi-layer perceptron for feature fusion, serving as another comparison to highlight the effectiveness of the proposed co-attention mechanism.
- **LLMHRec$_{w/o\ MVCL}$**: The multi-view contrastive learning module is removed, and the model is trained only with recommendation loss.

TABLE III

ABLATION STUDY ON KEY COMPONENTS OF LLMHREC.

Dataset	Model Variant	R@5	N@5	R@10	N@10
Amazon	LLMHRec$_{\mathrm{w/o\ RandMask}}$	0.0621	0.0489	0.0897	0.0614
	LLMHRec$_{\mathrm{w/o\ LLM}}$	0.0554	0.0432	0.0825	0.0569
	LLMHRec$_{\mathrm{w/o\ Denoise}}$	0.0614	0.0482	0.0890	0.0607
	LLMHRec$_{\mathrm{w/o\ Interact}}$	0.0542	0.0425	0.0813	0.0561
	LLMHRec$_{\mathrm{Concat}}$	0.0598	0.0467	0.0872	0.0592
	LLMHRec$_{\mathrm{MLP}}$	0.0605	0.0473	0.0879	0.0597
	LLMHRec$_{\mathrm{w/o\ MVCL}}$	0.0602	0.0470	0.0875	0.0590
	LLMHRec	**0.0640**	**0.0501**	**0.0921**	**0.0630**
Yelp	LLMHRec$_{\mathrm{w/o\ RandMask}}$	0.0537	0.0362	0.0669	0.0478
	LLMHRec$_{\mathrm{w/o\ LLM}}$	0.0475	0.0315	0.0602	0.0431
	LLMHRec$_{\mathrm{w/o\ Denoise}}$	0.0531	0.0358	0.0663	0.0474
	LLMHRec$_{\mathrm{w/o\ Interact}}$	0.0468	0.0311	0.0594	0.0427
	LLMHRec$_{\mathrm{Concat}}$	0.0514	0.0347	0.0647	0.0461
	LLMHRec$_{\mathrm{MLP}}$	0.0520	0.0352	0.0653	0.0465
	LLMHRec$_{\mathrm{w/o\ MVCL}}$	0.0518	0.0350	0.0651	0.0463
	LLMHRec	**0.0551**	**0.0372**	**0.0685**	**0.0489**

The experimental results in Table III highlight the effectiveness of each component in LLMHRec, with the following conclusions:

- **Advantage of LLM-based Semantic Encoding.** Removing the random masking strategy (LLMRec$_{w/o\ RandMask}$) or the LLM encoder itself (LLMHRec$_{w/o\ LLM}$) both result in clear performance drops. This demonstrates that perturbation-enhanced semantic embeddings from the large language model are crucial for capturing robust and context-aware user preferences.

- **Validity of Heterogeneous Hypergraph Modeling.** The degradation of $\text{LLMHRec}_{\text{w/o Denoise}}$ and $\text{LLMHRec}_{\text{w/o Interact}}$ confirms the importance of denoised hypergraph construction and interaction embeddings. These modules effectively filter noisy relations and encode higher-order structural patterns, complementing semantic information.
- **Impact of Cross-view Fusion Strategy.** Replacing co-attention with simple concatenation or MLP (LLMHRec_{Concat}, LLMHRec_{MLP}) leads to weaker performance, showing that cross-attention provides a more fine-grained and adaptive integration between semantic and structural embeddings.
- **Benfit of Multi-view Contrastive Learning.** The removal of the contrastive learning module ($\text{LLMHRec}_{w/o\ MVCL}$) degrades performance on both datasets, verifying that multi-view alignment enhances representation robustness and improves generalization.

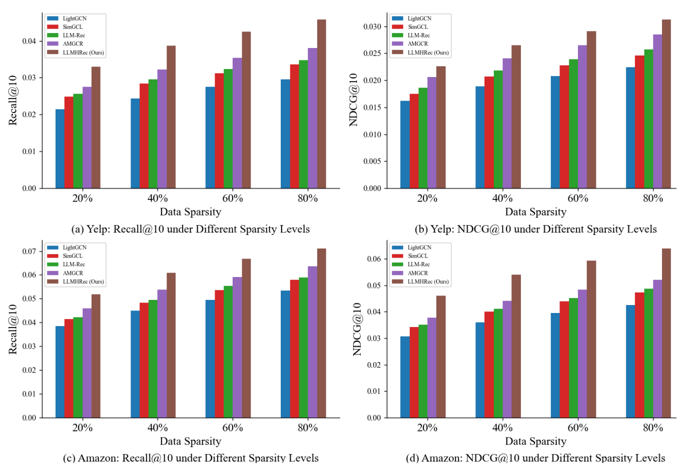

Fig. 2. **Performance Comparison under Varying Data Sparsity.**

E. RQ3:Data Sparsity Analysis

To evaluate the robustness of LLMHRec under varying levels of data sparsity, we simulate five sparsity settings by randomly retaining 20%, 40%, 60%, and 80% of the original training interactions, while keeping the test set fixed to ensure fair comparison. Each setting is repeated five times with different random seeds, and the average performance is reported. We compare LLMHRec with representative a GNN-based method (LightGCN), a recent LLM-based model (LLM-Rec), and multi-view contrastive learning approaches (SimGCL, AMGCR) and as illustrated in Figure 2, LLMHRec consistently outperforms all baselines across all sparsity levels on both Amazon and Yelp datasets, showing particularly strong performance under more sparse conditions. For instance, even at 40% training data, LLMHRec achieves Recall@10 around 0.06 and NDCG@10 around 0.05 on both amazon, whereas baselines exhibit more significant performance degradation and the robustness of LLMHRec stems from four key components: random content-word masking with LLM-based semantic encoding for richer personalized text features, a LightHGCN encoder that captures high-order interactions on both original and denoised hypergraphs to reduce noise, a co-attention fusion module that integrates semantic and interaction embeddings into more discriminative representations, and multi-view contrastive learning that aligns different representation views to enhance generalization. Together, these design choices enable LLMHRec to maintain strong and stable recommendation performance even under sparse training interactions.

F. RQ4:Over-smoothing Issue Analysis

To evaluate the robustness of LLMHRec against over-smoothing—a key limitation of deep GNNs—we compare it with NGCF, LightGCN, and HGRec using the Mean Average Distance (MAD) metric [33], which measures embedding diversity (higher MAD indicates less smoothing). MAD is computed at layers 0, 2, 4, 6, and 8 to track changes with propagation depth and as shown in Figure 3, LLMHRec consistently achieves the highest MAD values across all graph layers, reflecting its strong ability to preserve structural diversity in node embeddings. While NGCF and LightGCN exhibit rapid declines in MAD with increasing propagation depth—indicating severe over-smoothing—HGRec mitigates this effect to some extent by modeling higher-order hypergraph structures, but its MAD values still decrease noticeably (from 0.239 to 0.127). In contrast, LLMHRec maintains relatively stable MAD scores (from 0.262 to 0.196) across layers, which can be attributed to its combination of LightHGCN-based propagation on both original and denoised hypergraphs, along with multi-head attention for integrating structural signals. This demonstrates that LLMHRec effectively retains discriminative interaction-aware embeddings even in deeper graph layers, highlighting its robustness against over-smoothing and superior structural generalization.

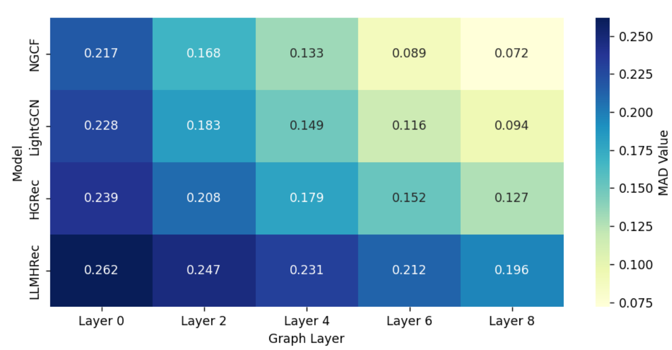

Fig. 3. **Over-smoothing Sensitivity Analysis**.

G. RQ5:Hyperparameter Sensitivity Analysis

To evaluate the robustness and tuning behavior of our LLMHRec model, we perform a comprehensive sensitivity analysis on four critical hyperparameters: the number of message passing layers L in the LightHGCN module, the number

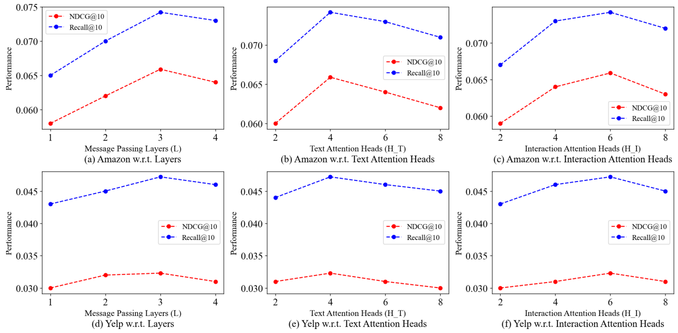

Fig. 4. **Hypermeters Analysis to Layers and Mutil-Attention Heads**

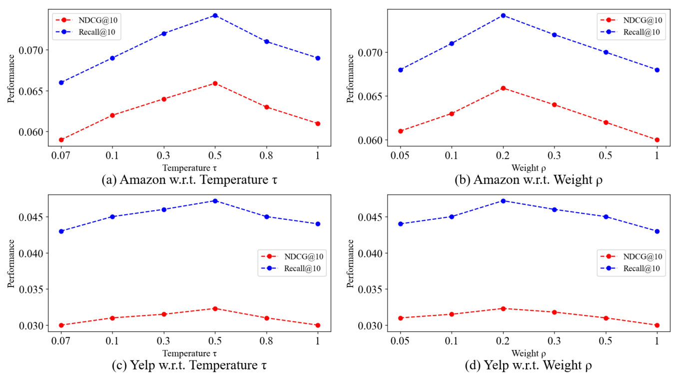

Fig. 5. **Study of Temperature and the Weight Hyperparameter ρ in Contrastive Learning of the LLMHRec**

of attention heads H_s in the semantic multi-head attention module, the number of attention heads H_i in the interaction multi-head attention module, and the multi-view contrastive learning parameters τ and ρ.

Impact of Message Passing Layers (L). We vary $L \in \{1, 2, 3, 4\}$ to examine structural depth. As shown in Figures 4(a) and (d), both Amazon and Yelp datasets peak at $L = 3$. Performance increases from $L = 1$ to 3, reflecting more effective high-order interaction modeling, but declines slightly beyond three layers due to potential over-smoothing.

Impact of Multi-Head Attention (H_T, H_I). We conduct two separate experiments for attention heads in semantic fusion (H_T) and interaction fusion (H_I), varying $H \in \{2, 4, 6, 8\}$. Figures 4(b), (c), (d) and (e) show that semantic and interaction attention modules benefit from increasing heads up to 4 and 6, beyond which performance gains plateau or decrease slightly. This indicates that moderate attention head counts are sufficient to capture complementary patterns without over-fragmenting representation capacity.

Impact of Contrastive yperparameters (τ, ρ). We vary the multi-view contrastive temperature $\tau \in \{0.07, 0.1, 0.3, 0.5, 0.8, 1\}$ and weight $\rho \in \{0.05, 0.1, 0.2, 0.3, 0.5, 1\}$. Figures 5 indicate that the best performance is achieved at $\tau = 0.5$ and $\rho = 0.2$ on both datasets. Smaller τ leads to overly sharp similarity distributions, while larger τ reduces contrastive pressure. Excessive ρ destabilizes training, whereas too small ρ underutilizes auxiliary views.

Overall, these results confirm that optimal hyperparameters of $L = 3$, $H_T = 4$, $H_I = 6$, $\tau = 0.5$, and $\rho = 0.2$ provide a good balance between high-order structural modeling, attention aggregation, and multi-view contrastive regularization, ensuring robust and stable recommendation performance.

V. Related Work

A. Large Language Model based Approaches

Recent advances in large language models (LLMs) have greatly influenced personalized recommendation, especially when rich textual information is available. BERT4Rec [14] adapts the Transformer for sequential recommendation by modeling user behavior bidirectionally, capturing complex temporal dependencies. LLM-Rec [24] uses diverse prompting strategies to enhance item representations and align them with user preferences. GenRec [25] adopts a generative approach by fine-tuning LLaMA on user–item interaction text to directly generate target items. Despite these advances, existing LLM-based methods often overlook structured user–item relationships and multi-view interactions. In contrast, LLMHRec introduces a multi-view semantic encoder with syntactic masking, generating diverse and robust semantic embeddings.

B. Graph Neural Network based Approaches

Graph Neural Networks (GNNs) are widely used in collaborative filtering for modeling high-order user–item relationships. NGCF [8] propagates embeddings across a bipartite user–item graph to capture collaborative signals via stacked convolutions. LightGCN [9] simplifies graph convolutions by removing feature transformations and non-linearities, demonstrating that pure neighborhood aggregation suffices for effective recommendation. HGRec [28] extends GNNs to heterogeneous graphs, fusing multi-hop neighbor information via attention to enhance node representations. While effective, these models may suffer from over-smoothing. LLMHRec addresses this by explicitly incorporating review nodes and constructing a heterogeneous hypergraph encoding user, item, and review interactions. LightHGCN with edge-level denoising and multi-head attention ensures scalable, expressive embeddings for downstream recommendation.

C. Multi-View Learning Based Approaches

Multi-view learning improves robustness and generalization in recommender systems. SimGCL [19] generates contrastive views by adding uniform noise to embeddings. LightGCL [30] proposes a lightweight graph contrastive learning framework for efficiency without sacrificing performance. AMGCR [32] creates diverse views via structured perturbations and fuses them adaptively. However, these methods often rely on simple perturbations or predefined structures. LLMHRec unifies semantic and structural perspectives under a multi-view framework, integrating perturbation-aware textual embeddings with high-order hypergraph-based interaction embeddings via a co-attention mechanism. A multi-view contrastive learning objective further enhances robustness and generalization of user–item representations.

VI. Conclusion

This paper proposes LLMHRec, a unified recommendation framework that integrates LLM-enhanced semantic embeddings with heterogeneous hypergraph-based interaction modeling. Textual representations are learned via a perturbation-aware multi-view encoder, while a lightweight hypergraph convolution captures interaction-aware embeddings. A co-attention module aligns semantic and structural signals, and multi-view contrastive learning improves robustness. Experiments on two real-world datasets demonstrate that LLMHRec outperforms state-of-the-art baselines, especially under sparse and noisy interactions. In the future we will investigate temporal dynamics and advanced hypergraph architectures to further enhance multi-view representations.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant 61976061 and Grant 62102461.

References

[1] J. Wang, R. Louca, D. Hu, C. Cellier, J. Caverlee, and L. Hong, "Time to shop for valentine's day: Shopping occasions and sequential recommendation in e-commerce," in *Proceedings of the 13th International Conference on Web Search and Data Mining*, 2020, pp. 645–653.

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

[3] A. Van den Oord, S. Dieleman, and B. Schrauwen, "Deep content-based music recommendation," *Advances in neural information processing systems*, vol. 26, 2013.

[4] M. Bevec, M. Tkalčič, and M. Pesek, "Hybrid music recommendation with graph neural networks," *User Modeling and User-Adapted Interaction*, pp. 1–38, 2024.

[5] C.-Y. Lin, T.-S. Chen, J. Chen, and C.-Y. Chen, "Personalized live streaming channel recommendation based on most similar neighbors," *Multimedia Tools and Applications*, vol. 80, pp. 19 867–19 883, 2021.

[6] I. Marzuki, M. Hariadi, R. F. Rachmadi, and Y. M. Arif, "Neural collaborative filtering for improved tourism destination recommendation," in *2024 8th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*. IEEE, 2024, pp. 481–486.

[7] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," *Advances in neural information processing systems*, vol. 30, 2017.

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

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

[10] Y. Feng, H. You, Z. Zhang, R. Ji, and Y. Gao, "Hypergraph neural networks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 3558–3565.

[11] S. Ji, Y. Feng, R. Ji, X. Zhao, W. Tang, and Y. Gao, "Dual channel hypergraph collaborative filtering," in *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, R. Gupta, Y. Liu, J. Tang, and B. A. Prakash, Eds. ACM, 2020, pp. 2020–2029.

[12] Y. Li, Z. Fan, and J. Zhang, "Heterogeneous hypergraph neural network for friend recommendation with human mobility," in *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022*. ACM, 2022, pp. 4209–4213.

[13] A. H. Nasution and A. Onan, "Chatgpt label: Comparing the quality of human-generated and llm-generated annotations in low-resource language nlp tasks," *IEEE Access*, 2024.

[14] F. Sun, J. Liu, J. Wu, C. Pei, X. Lin, W. Ou, and P. Jiang, "Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer," in *Proceedings of the 28th ACM international conference on information and knowledge management*, 2019, pp. 1441–1450.

[15] X. Xie, F. Sun, Z. Liu, S. Wu, J. Gao, J. Zhang, B. Ding, and B. Cui, "Contrastive learning for sequential recommendation," in *2022 IEEE 38th international conference on data engineering (ICDE)*. IEEE, 2022, pp. 1259–1273.

[16] Y. Wang, Z. Chu, X. Ouyang, S. Wang, H. Hao, Y. Shen, J. Gu, S. Xue, J. Y. Zhang, Q. Cui, L. Li, J. Zhou, and S. Li, "Enhancing recommender systems with large language model reasoning graphs," *CoRR*, vol. abs/2308.10835, 2023.

[17] Z. Liu and L. Zhang, "Enhanced self-supervised graph learning algorithm for social recommendation," in *2024 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2024, pp. 1–8.

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

[19] C. Liu, C. Yu, N. Gui, Z. Yu, and S. Deng, "Simgcl: graph contrastive learning by finding homophily in heterophily," *Knowledge and Information Systems*, vol. 66, no. 3, pp. 2089–2114, 2024.

[20] G. Zou, S. Lin, S. Hu, Y. Gan, B. Zhang, and Y. Chen, "Mvgcl: Multi-view graph contrastive learning for service recommendation," in *2023 IEEE International Conference on Web Services (ICWS)*, 2023.

[21] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "Bpr: Bayesian personalized ranking from implicit feedback," *arXiv preprint arXiv:1205.2618*, 2012.

[22] X. Ding, H. Yang, R. H. Chan, H. Hu, Y. Peng, and T. Zeng, "A new initialization method for neural networks with weight sharing," in *Mathematical Methods in Image Processing and Inverse Problems: IPIP 2018, Beijing, China, April 21–24*. Springer, 2021, pp. 165–179.

[23] S. Y. Şen and N. Özkurt, "Convolutional neural network hyperparameter tuning with adam optimizer for ecg classification," in *2020 innovations in intelligent systems and applications conference (ASYU)*. IEEE, 2020, pp. 1–6.

[24] H. Lyu, S. Jiang, H. Zeng, Y. Xia, Q. Wang, S. Zhang, R. Chen, C. Leung, J. Tang, and J. Luo, "Llm-rec: Personalized recommendation via prompting large language models," 2024. [Online]. Available: https://arxiv.org/abs/2307.15780

[25] X. Guo, T. Zhang, Y. Wang, C. Wang, F. Wang, X. Wang, X. Zhang, X. Liu, and Z. Cui, "Multi-modal hypergraph enhanced LLM learning for recommendation," *CoRR*, vol. abs/2504.10541, 2025.

[26] Z. Chu, H. Hao, X. Ouyang, S. Wang, Y. Wang, Y. Shen, J. Gu, Q. Cui, L. Li, S. Xue *et al.*, "Leveraging large language models for pre-trained recommender systems," *arXiv preprint arXiv:2308.10837*, 2023.

[27] N. Yadati, M. Nimishakavi, P. Yadav, V. Nitin, A. Louis, and P. Talukdar, "Hypergcn: A new method for training graph convolutional networks on hypergraphs," *Advances in neural information processing systems*, vol. 32, 2019.

[28] J. Shi, H. Ji, C. Shi, X. Wang, Z. Zhang, and J. Zhou, "Heterogeneous graph neural network for recommendation," 2020. [Online]. Available: https://arxiv.org/abs/2009.00799

[29] D. Sakong, T. T. Huynh, and J. Jo, "Hypergraph diffusion for high-order recommender systems," *arXiv preprint arXiv:2501.16722*, 2025.

[30] X. Cai, C. Huang, L. Xia, and X. Ren, "Lightgcl: Simple yet effective graph contrastive learning for recommendation," *arXiv preprint arXiv:2302.08191*, 2023.

[31] L. Xia, C. Huang, Y. Xu, J. Zhao, D. Yin, and J. Huang, "Hypergraph contrastive collaborative filtering," in *Proceedings of the 45th International ACM SIGIR conference on research and development in information retrieval*, 2022, pp. 70–79.

[32] M. Yang, Y. Yuan, J. Zhou, M. Xi, X. Pan, Y. Li, Y. Wu, J. Zhang, and J. Yin, "Adaptive fusion of multi-view for graph contrastive recommendation," in *Proceedings of the 18th ACM Conference on Recommender Systems*, ser. RecSys '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 228–237.

[33] D. Chen, Y. Lin, W. Li, P. Li, J. Zhou, and X. Sun, "Measuring and relieving the over-smoothing problem for graph neural networks from the topological view," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 04, 2020, pp. 3438–3445.