HDAF-LLM: Hierarchical Decoupled Adaptive Fusion with Large Language Models for Multi-Domain Recommendation

Lichang Zhao[1], Baojun Tian[2,4✉], Pengyu Chen[3]
[1]*College of Intelligent Science and Technology, Inner Mongolia University of Technology*, Hohhot 010080, China
3173494859@qq.com https://orcid.org/0009-0004-2691-8781
[2]✉ *College of Intelligent Science and Technology, Inner Mongolia University of Technology*, Hohhot 010080, China
ngdtbj@126.com https://orcid.org/0009-0008-8129-5619
[3]*College of Intelligent Science and Technology, Inner Mongolia University of Technology*, Hohhot 010080, China
1053917487@qq.com https://orcid.org/0009-0001-2206-3278
[4]*Inner Mongolia Key Laboratory of Intelligent Perception and System Engineering*, Hohhot 010080, China

***Abstract*—Multi-domain recommendation (MDR) leverages shared user interests across domains to improve personalized recommendations. However, existing methods face two critical challenges: (1) cross-domain preference interference, where naive fusion causes negative transfer, and (2) semantic-structural misalignment between LLM-based features and graph-based collaborative patterns. Prior approaches address these separately through implicit regularization or static integration, lacking unified solutions with explicit guarantees. We propose HDAF-LLM (Hierarchical Decoupled Adaptive Fusion with Large Language Models), a novel framework that systematically tackles both challenges. First, we introduce explicit architectural decoupling through separate Global-HGCN and Domain-HGCN modules with isolated parameter sets, ensuring zero cross-domain gradient interference. Second, we design layer-wise contrastive learning with augmentation-based hard negatives to progressively align LLM semantics with graph embeddings. Third, we develop a two-stage adaptive fusion mechanism with density-aware weights that dynamically balances domain-specific and global information. Extensive experiments on six Amazon sub-domain datasets demonstrate that HDAF-LLM achieves statistically significant improvements of 1.15%–6.78% over state-of-the-art baselines (paired t-test, $p < 0.05$), with particularly notable gains of 7.8%–19.1% in cold-start scenarios.**

***Index Terms*—Multi-domain recommender, Large language models, Decoupled learning, Hierarchical contrastive learning**

I. Introduction

Multi-domain recommendation (MDR) focuses on delivering personalized recommendations across overlapping domains of users and items [1], [2]. MDR differs from Cross-Domain Recommendation (CDR) in scope: *CDR transfers knowledge between two specific domains, while MDR jointly models $K \geq 2$ domains with potentially complex inter-domain relationships*. MDR leverages user interests in one domain to improve recommendations in others, boosting user engagement and platform growth [3].

However, existing MDR methods face two critical challenges. **Challenge 1: Cross-domain preference interference.** Naive fusion of cross-domain information often leads to negative transfer [4], where domain-specific patterns are distorted by irrelevant cross-domain signals. Methods like TriCDR [5] and AMID [6] use contrastive objectives but lack explicit architectural mechanisms to prevent interference during message passing. **Challenge 2: Semantic-structural misalignment.** LLM-based methods [7], [8] capture rich semantic relationships but struggle to align with graph-based structural patterns [9], especially in multi-domain settings where domain-specific semantic contexts must be preserved.

Research Gap. Existing approaches address these challenges separately: graph-based methods [10], [11] focus on structural transfer but ignore semantic enrichment, while LLM-based methods [7], [12] enhance semantics but overlook structural dependencies. No prior work addresses *both* challenges through a unified hierarchical framework with explicit decoupling mechanisms and adaptive fusion weights.

To fill this gap, we propose HDAF-LLM with three key innovations: (1) **Explicit architectural decoupling** via separate parameter sets (W_d^l vs. W_{shared}^l) that prevents negative transfer at the message-passing level, unlike implicit regularization approaches. (2) **Layer-wise contrastive alignment** that progressively bridges semantic-structural gaps using augmentation-based hard negatives. (3) **Density-adaptive fusion weights** that automatically balance domain-specific and global information based on domain characteristics.

Our main contributions are:

1) A hierarchical framework with explicit architectural decoupling that models global and domain-specific graphs separately, preventing negative transfer through parameter isolation.
2) Integration of LLMs with layer-wise contrastive learning that progressively aligns semantic and structural information across GNN layers.
3) A two-stage adaptive fusion mechanism with closed-form weight functions based on interaction frequency, information gain, and domain density.

II. Related Work

A. Multi-Domain Recommendation

CDR tackles data sparsity and cold-start issues through knowledge transfer [13], serving as the foundation of MDR. GA-DTCDR [10] and C2DSR [11] incorporate graph structures, TriCDR [5] and AMID [6] employ contrastive objectives, D3 [14] utilizes attention-based fusion gates for domain modeling. However, these methods rely on implicit decoupling through regularization or soft-weighting mechanisms, which cannot explicitly separate domain-shared and domain-specific parameters, failing to guarantee interference-free optimization in complex multi-domain scenarios. Our explicit parameter separation provides stronger guarantees for cross-domain interference mitigation.

B. LLMs for Recommendation

LLMs capture fine-grained semantic relationships [15]. TALLRec [16] and LLM-ESR [7] refined LLM integration through instruction tuning and self-distillation. However, these methods treat LLM features as static inputs without progressive alignment to graph structures, leading to semantic-structural gaps in multi-domain settings.

III. Methodology

A. Problem Statement

Let U, I, and D denote the sets of users, items, and domains, with $|U| = M$, $|I| = N$, and $|D| = K$. For each domain $d \in D$, its interaction set is $R_d \subseteq U_d \times I_d$. This work aims to design HDAF-LLM that mitigates cross-domain interference and aligns semantic and structural information.

B. Overall Framework

HDAF-LLM (Fig. 1) consists of four main components: **(1) Global Heterogeneous Graph Convolutional Network (Global-HGCN):** Aggregates cross-domain information using shared parameters W^l_{shared}. **(2) Domain-specific Heterogeneous Graph Convolutional Network (Domain-HGCN):** Models domain-specific interactions using domain-specific parameters W^l_d. **(3) LLM-based Semantic Feature Extraction:** Uses ChatGLM2-6B to generate user preference and item attribute vectors through designed prompts. **(4) Layer-wise Contrastive Learning with Two-Stage Adaptive Fusion:** Aligns semantic features with structural embeddings, then dynamically fuses them.

C. Disentangled Convolutional Layer

The term "disentangled" refers to *explicit architectural separation* where Domain-HGCN operates independently within each domain's subgraph using parameters W^l_d, while Global-HGCN aggregates across all domains using W^l_{shared}.

1) Theoretical Justification: We justify explicit parameter separation over implicit regularization based on two observations: (1) **Gradient isolation:** Separate parameters ensure $\frac{\partial \mathcal{L}_d}{\partial W^l_{d'}} = 0$ for $d \neq d'$, preventing gradient interference. (2) **Capacity allocation:** Each domain has dedicated parameters to capture domain-specific patterns. Our experiments in Section 4.6 empirically validate this design.

2) Domain-Specific Graph Convolution (Domain-HGCN): For user u in domain d, Domain-HGCN computes:

$$h^{l+1}_{u,d} = \sigma\left(\text{Drop}\left(\sum_{i \in \mathcal{N}^d_u} \tilde{A}^d_{ui} h^l_{i,d} W^l_d + b_u, p\right)\right) \quad (1)$$

where $\tilde{A}^d_{ui} = 1/\sqrt{|\mathcal{N}^d_u||\mathcal{N}^d_i|}$ is the normalized adjacency, $\sigma(\cdot)$ is ReLU, $p = 0.1$ is dropout rate, $W^l_d \in \mathbb{R}^{d_l \times d_{l+1}}$ is domain-specific, and $\mathcal{N}^d_u$ is the neighbor set of u in domain d.

3) Domain-Shared Graph Convolution (Global-HGCN): For user u, Global-HGCN aggregates across all domains:

$$g^{l+1}_u = \sigma\left(\text{Drop}\left(\sum_{d \in \mathcal{D}} \sum_{i \in \mathcal{N}^d_u} \tilde{A}^g_{ui} g^l_i W^l_{shared} + b^s_u, p\right)\right) \quad (2)$$

where $\tilde{A}^g_{ui} = 1/\sqrt{|\mathcal{N}^g_u||\mathcal{N}^g_i|}$ and $|\mathcal{N}^g_u| = \sum_d |\mathcal{N}^d_u|$ is total neighbors.

4) LLM Integration: The LLM vectors $z_u, z_i \in \mathbb{R}^{4096}$ (from ChatGLM2-6B) are projected:

$$\tilde{z}_u = W_{proj} z_u + b_{proj}, \quad \tilde{z}_i = W_{proj} z_i + b_{proj} \quad (3)$$

where $W_{proj} \in \mathbb{R}^{4096 \times d_0}$ projects to $d_0 = 256$. The semantic-enhanced initial embeddings are:

$$h^0_{u,d} = \tilde{z}_u + e_u, \quad h^0_{i,d} = \tilde{z}_i + e_i \quad (4)$$

where $e_u, e_i \in \mathbb{R}^{d_0}$ are learnable ID embeddings. This additive combination serves as the input to both Domain-HGCN and Global-HGCN, enabling semantic information to propagate through graph convolution layers.

D. LLM-Based Adaptive Mechanism

1) LLM Deployment Strategy: **Offline Pre-computation:** All LLM vectors are pre-computed offline and cached in a FAISS index (IVF with 1024 clusters). Storage cost: 4096 dims $\times$ 4 bytes $\times$ 1M users $\approx$ 16GB. Pre-computation takes 0.8s per user/item on NVIDIA A100 (40GB).

LLM Configuration: ChatGLM2-6B with: temperature 0.1, max output 512 tokens, batch size 32, FP16 inference. We extract the last hidden state of the `[EOS]` token as the 4096-dim representation.

Periodic Updates: User vectors are refreshed every 24 hours for active users ($\geq$3 new interactions). During inference, only cached vectors are used with 95th percentile retrieval latency $<$ 0.5ms.

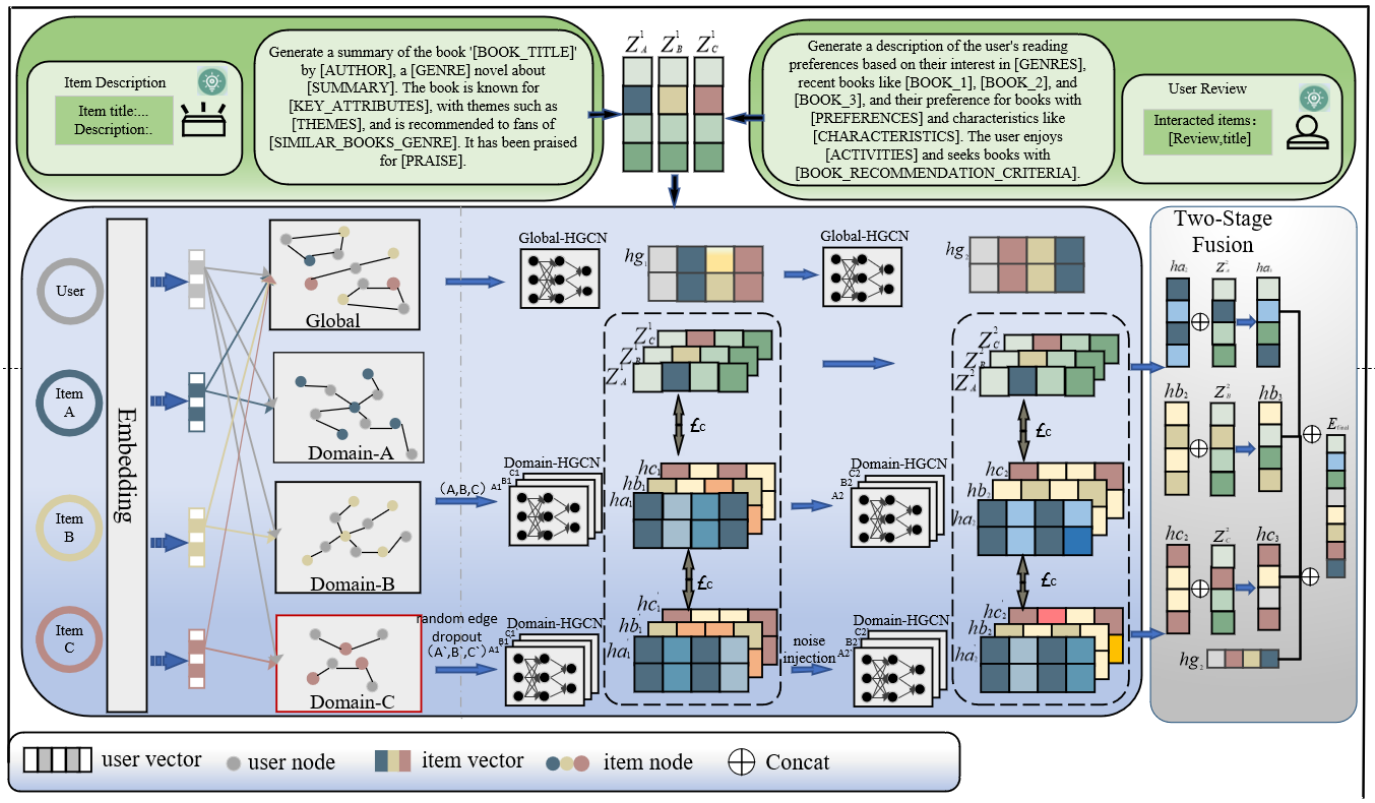

Fig. 1. Framework of HDAF-LLM. Global-HGCN and Domain-HGCN use separate parameters (W^l_{shared} vs. W^l_d) to prevent negative transfer. Layer-wise contrastive learning aligns LLM features with graph embeddings progressively. Two-stage fusion combines features with density-adaptive weights.

2) Prompt Templates: **User Prompt:** *"Based on the user's interaction history in [DOMAIN]: [ITEM_LIST]. Long-term interests: [CATEGORIES]. Recent items (last 7 days): [RECENT_ITEMS]. Generate a 50-word description of this user's preferences and needs in [DOMAIN]."*

Item Prompt: *"Item: [TITLE]. Category: [CATEGORY]. Description: [DESC]. Generate a 50-word summary of this item's key features and target users."*

The user vector combines enduring and recent preferences:

$$z_u = \frac{\alpha(t) z_{\text{end}} + \beta(t) z_{\text{rec}}}{\|\alpha(t) z_{\text{end}} + \beta(t) z_{\text{rec}}\|_2} \tag{5}$$

where $\alpha(t) = 1/(1+e^{-\tau(\bar{t}_{long} - t_{th})})$, $\beta(t) = 1-\alpha(t)$, $\tau = 0.1$, and t_{th} is learnable (initialized to 30 days).

E. Contrastive Learning for Multi-Domain Recommendations

1) Hard Negative Definition: A *hard negative* is a sample semantically or structurally similar to the positive sample but should be distinguished from it. We construct hard negatives through data augmentation (edge dropout and noise injection) rather than mining from the dataset.

Why Single Hard Negative: Unlike standard contrastive methods using large negative pools, we employ a single augmentation-based hard negative per sample for two reasons: (1) Our goal is semantic-structural *alignment* rather than discrimination, so one carefully constructed hard negative suffices. (2) Augmentation-based negatives maintain domain relevance without introducing noisy cross-domain samples.

2) Notation: Let $\mathbf{h}_d^{(l)} \in \mathbb{R}^{d_l}$ denote Domain-HGCN's output at layer l. Let $\tilde{\mathbf{z}}_{LLM} \in \mathbb{R}^{d_0}$ denote the projected LLM feature. Temperature $\tau = 0.5$, and $\text{sim}(\cdot,\cdot)$ is cosine similarity.

3) First Layer Contrastive Loss: Using edge dropout ($p = 0.1$), we construct hard negative $\mathbf{h}^{(1)}_{drop}$. Let $s_{pos} = \text{sim}(\mathbf{h}^{(1)}_d, \tilde{\mathbf{z}}_{LLM})$ and $s_{neg} = \text{sim}(\mathbf{h}^{(1)}_d, \mathbf{h}^{(1)}_{drop})$:

$$\mathcal{L}_1 = -\log \frac{\exp(s_{pos}/\tau)}{\exp(s_{pos}/\tau) + \exp(s_{neg}/\tau)} \tag{6}$$

F. Second Layer Contrastive Loss

Using noise injection, we construct $\mathbf{h}^{(2)}_{noise} = \mathbf{h}^{(2)}_d + \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, 0.1^2\mathbf{I})$. The momentum-updated target is:

$$\tilde{\mathbf{z}}^{(1)}_{LLM} = 0.9\tilde{\mathbf{z}}_{LLM} + 0.1\mathbf{h}^{(1)}_d \tag{7}$$

Let $s'_{pos} = \text{sim}(\mathbf{h}^{(2)}_d, \tilde{\mathbf{z}}^{(1)}_{LLM})$ and $s'_{neg} = \text{sim}(\mathbf{h}^{(2)}_d, \mathbf{h}^{(2)}_{noise})$:

$$\mathcal{L}_2 = -\log \frac{\exp(s'_{pos}/\tau)}{\exp(s'_{pos}/\tau) + \exp(s'_{neg}/\tau)} \tag{8}$$

The total contrastive loss is $\mathcal{L}_{cl} = \mathcal{L}_1 + \mathcal{L}_2$.

G. Two-Stage Fusion Mechanism

1) First Stage: LLM and Domain-Specific Fusion: The first stage fuses LLM semantic features with domain-specific graph embeddings:

$$\mathbf{z}_{f1}^{(d)} = \alpha \cdot \tilde{\mathbf{z}}_{LLM}^{(d)} + \beta \cdot \mathbf{h}_d^{(L)} \tag{9}$$

where $\mathbf{h}_d^{(L)}$ is Domain-HGCN's final output. The weights are computed as:

$$w_{freq}(u,d) = \frac{\log(1+|R_u^d|)}{\sum_{d'} \log(1+|R_u^{d'}|)} \tag{10}$$

$$w_{info}(u) = \sigma\left(\mathbf{w}_g^T[\tilde{\mathbf{z}}_{LLM}; \mathbf{h}_d^{(L)}] + b_g\right) \tag{11}$$

where $\mathbf{w}_g \in \mathbb{R}^{2d_0}$, $b_g \in \mathbb{R}$ are learnable. The normalized weights ($\alpha + \beta = 1$):

$$\alpha = \frac{w_{info}}{w_{info} + (1 - w_{info}) w_{freq} + \epsilon},\ \beta = 1 - \alpha \tag{12}$$

where $\epsilon = 10^{-8}$ ensures numerical stability.

Cold-start Handling: For entities with <5 interactions, we use $\alpha = 0.7$, $\beta = 0.3$. For zero-interaction users, we generate LLM vectors using demographic information; for zero-interaction items, we use category and title only.

H. Second Stage: Domain-Specific and Global Fusion

The second stage fuses first-stage output with global embeddings:

$$\mathbf{z}_{f2} = \gamma \cdot \mathbf{z}_{f1}^{(d)} + (1-\gamma) \cdot \mathbf{g}^{(L)} \tag{13}$$

where $\mathbf{g}^{(L)}$ is Global-HGCN's output, and γ is computed based on domain density:

$$\gamma = \sigma\left(\lambda \log\left(\frac{\rho(d)}{\bar{\rho}}\right)\right) \tag{14}$$

with $\lambda = 1.0$, $\rho(d) = |R_d|/(|U_d| \times |I_d|)$, and $\bar{\rho}$ the average density.

I. Fusion Loss

The fusion loss anchors fused representations to stable targets:

$$\mathcal{L}_{fus} = \|\mathbf{z}_{f1} - \hat{\mathbf{z}}_{f1}\|_2^2 + \|\mathbf{z}_{f2} - \hat{\mathbf{z}}_{f2}\|_2^2 \tag{15}$$

where the target vectors are defined as:

$$\hat{\mathbf{z}}_{f1} = \text{sg}(\tilde{\mathbf{z}}_{LLM}), \quad \hat{\mathbf{z}}_{f2} = \text{sg}(\mathbf{g}^{(L)}) \tag{16}$$

and $\text{sg}(\cdot)$ denotes stop-gradient. We anchor to LLM features because: (1) LLM embeddings are pre-computed and stable. (2) Anchoring to domain embeddings creates circular dependencies.

J. Learning and Training

The model optimizes the total loss combining four objectives:

$$\mathcal{L}_{tot} = \mathcal{L}_{BPR} + \lambda_1 \mathcal{L}_{cl} + \lambda_2 \mathcal{L}_{fus} + \lambda_3 \mathcal{L}_{reg} \tag{17}$$

where $\lambda_1 = 0.1$ balances semantic-structural alignment against recommendation accuracy, $\lambda_2 = 0.01$ anchors fused representations without dominating the primary objective, and $\lambda_3 = 10^{-5}$ provides mild weight decay to prevent overfitting. These values were determined via grid search on the validation set (details in Section **??**).

BPR Loss. We adopt Bayesian Personalized Ranking [**?**] as the primary recommendation objective:

$$\mathcal{L}_{BPR} = -\sum_{(u,i^+,i^-)\in\mathcal{O}} \log\sigma\left(\hat{y}_{u,i^+} - \hat{y}_{u,i^-}\right) \tag{18}$$

where $\mathcal{O} = \{(u,i^+,i^-) \mid (u,i^+) \in R_d, (u,i^-) \notin R_d\}$ is the pairwise training set, $\hat{y}_{u,i} = \mathbf{z}_{f2}^{(u)\top} \mathbf{z}_{f2}^{(i)}$ is the predicted preference score computed from the final fused representations, i^+ denotes an observed positive item, and i^- is a uniformly sampled negative item. BPR directly optimizes the ranking order rather than absolute scores, which is more appropriate for implicit feedback data where only positive interactions are observed.

Regularization. To prevent overfitting and improve generalization across domains, we apply Frobenius norm regularization on all learnable weight matrices:

$$\mathcal{L}_{reg} = \sum_{l=1}^{L} \left(\|W_d^l\|_F^2 + \|W_{shared}^l\|_F^2\right) + \|W_{proj}\|_F^2 \tag{19}$$

The regularization covers domain-specific parameters W_d^l, shared parameters W_{shared}^l across all GNN layers $l \in \{1,\ldots,L\}$, and the LLM projection matrix W_{proj}. Learnable ID embeddings e_u, e_i are excluded from regularization to preserve user/item identity representations. Note that LLM parameters (ChatGLM2-6B) are frozen throughout training, contributing no gradient updates and requiring no regularization.

Optimization. The full model is trained end-to-end using the Adam optimizer with learning rate $\eta = 10^{-3}$, batch size 64, and embedding dimension $d_0 = 256$. We apply early stopping with patience 10 based on HR@10 on the validation set. Negative items for BPR are sampled uniformly per batch; for contrastive learning, hard negatives are constructed via augmentation as described in Section **??**. The gradient of $\mathcal{L}_{tot}$ with respect to W_d^l is isolated from other domains by construction (i.e., $\frac{\partial \mathcal{L}_d}{\partial W_{d'}^l} = 0$ for $d \neq d'$), ensuring interference-free parameter updates consistent with our decoupling design.

IV. Experimental Evaluation

A. Datasets and Setup

We evaluate HDAF-LLM on six Amazon sub-domain datasets (Table I). These datasets exhibit diverse characteristics: Sport has the highest density (0.70%) while Electronic

has the lowest (0.01%), and Clothing contains the largest item set (193,304 items).

TABLE I
DATASET STATISTICS.

Dataset	Users	Items	Inter.	Density
Sport	4,263	1,021	30,512	0.70%
Electronic	20,728	10,492	25,660	0.01%
Phone	11,762	2,246	25,175	0.095%
All Beauty	24,752	8,788	33,929	0.016%
Clothing	24,752	193,304	370,840	0.008%
Luxury Beauty	24,752	6,980	56,356	0.033%

Environment: PyTorch 1.12.1, PyTorch Geometric 2.2.0, Python 3.9, CUDA 11.6, NVIDIA A100 (40GB). **Preprocessing:** (1) 5-core filtering. (2) Remove duplicate interactions (user-item-timestamp). (3) Filter interactions outside 2018-2023; (4) Chronological 8:1:1 split. **Training:** Adam optimizer, learning rate $\eta = 10^{-3}$, batch size 64, embedding dimension $d_0 = 256$, GNN layers $L = 3$, early stopping (patience 10). **Statistical Testing:** Two-sided paired t-test (3 runs, seeds 42–44), $n = 5$, $df = 4$. * denotes $p < 0.05$.

B. Baselines

We compare HDAF-LLM with three baseline categories: **SDSR:** GRU4Rec [17] models dynamic preferences via RNNs; SASRec [18] captures temporal dependencies using self-attention. **CDSR:** GA-DTCDR [10] combines graph structures with attention; DDTCDR [19] employs orthogonal mapping and autoencoders; C2DSR [11] propagates knowledge via GNNs; TriCDR [5] aligns preferences through contrastive learning; AMID [6] models multi-interest user groups. **LLM-SR:** SAID [12] fine-tunes LLMs for user-item interpretation; LLM-ESR [7] integrates LLM embeddings with self-distillation for long-tail recommendation.

C. Overall Performance

Tables II and III show performance across six domains. "Improv.%" is relative to the best baseline (underlined) per metric. Key observations: **(1) Consistent superiority across domains:** HDAF-LLM outperforms all baselines with statistically significant gains ($p < 0.05$), ranging from 1.15% to 6.78%. **(2) Larger gains in sparse domains:** Improvements are more significant in sparse domains (Electronic: 4.21% HR@10, All Beauty: 4.50% HR@10) than dense ones (Sport: 2.16% HR@10). This confirms our density-adaptive fusion: sparse domains gain more from Global-HGCN's cross-domain transfer (lower γ), while dense domains have adequate domain-specific signals. **(3) Superiority over LLM-based methods:** HDAF-LLM outperforms LLM-ESR and SAID by 2–4% on most metrics, thanks to layer-wise contrastive alignment and two-stage fusion bridging the semantic-structural gap. **(4) Complementary benefits:** HDAF-LLM combines the advantages of graph-based (C2DSR, TriCDR) and LLM-based (SAID, LLM-ESR) methods, capturing structural dependencies and semantic richness.

TABLE II
PERFORMANCE COMPARISON (PART 1). *: $p < 0.05$. BOLD: BEST; UNDERLINE: SECOND BEST. IMPROV.% IS RELATIVE TO THE BEST BASELINE.

Model	Sport		Electronic		Phone	
	H@10	N@10	H@10	N@10	H@10	N@10
GRU4Rec	0.581	0.489	0.335	0.200	0.323	0.190
SASRec	0.548	0.463	0.346	0.203	0.313	0.184
GA-DTCDR	0.609	0.502	0.371	0.236	0.305	0.191
DDTCDR	0.638	0.590	0.402	0.257	0.341	0.214
C2DSR	0.620	0.582	0.468	0.332	0.297	0.186
TriCDR	0.628	0.583	0.467	0.334	0.297	0.188
AMID	0.649	0.578	0.475	0.312	0.349	0.225
SAID	0.635	0.590	0.464	0.290	0.354	0.221
LLM-ESR	0.639	0.608	0.473	0.296	0.355	0.219
HDAF-LLM	**0.663***	**0.615***	**0.495***	**0.349***	**0.371***	**0.238***
Improv.%	2.16	1.15	4.21	4.49	4.51	5.78

TABLE III
PERFORMANCE COMPARISON (PART 2). *: $p < 0.05$. BOLD: BEST; UNDERLINE: SECOND BEST. IMPROV.% IS RELATIVE TO THE BEST BASELINE.

Model	All Beauty		Clothing		Luxury	
	H@10	N@10	H@10	N@10	H@10	N@10
GRU4Rec	0.223	0.141	0.493	0.438	0.725	0.650
SASRec	0.242	0.188	0.566	0.485	0.747	0.631
GA-DTCDR	0.281	0.187	0.618	0.567	0.738	0.631
DDTCDR	0.298	0.201	0.624	0.566	0.786	0.663
C2DSR	0.345	0.250	0.638	0.530	0.755	0.641
TriCDR	0.337	0.295	0.643	0.572	0.775	0.641
AMID	0.358	0.258	0.659	0.574	0.742	0.658
SAID	0.364	0.272	0.652	0.582	0.778	0.689
LLM-ESR	0.378	0.273	0.660	0.591	0.762	0.674
HDAF-LLM	**0.395***	**0.315***	**0.668***	**0.603***	**0.804***	**0.705***
Improv.%	4.50	6.78	1.21	2.03	2.29	2.32

D. Ablation Study

Table IV presents the ablation study validating each component's contribution. We remove one component at a time, keeping others unchanged: **(1) Impact of Global-HGCN:** Removing Global-HGCN causes the largest performance drop in sparse domains (21.0% All Beauty, 9.1% Luxury Beauty). Cross-referencing Table I, lower-density domains (All Beauty: 0.016%, Luxury: 0.033%) suffer more, as the global graph's cross-domain connectivity compensates for insufficient domain-specific interactions—sparse-history users leverage related domain information via shared neighbors. **(2) Impact of Domain-HGCN:** Removing Domain-HGCN leads to 14.9% (All Beauty) and 10.0% (Clothing) drops, demonstrating domain-specific modeling is essential for per-domain user preferences and validating explicit decoupling. **(3) Impact of LLM features:** Without LLM features, performance drops 12.7% (All Beauty) and 4.5% (Clothing). The larger drop in sparser domains shows LLM semantic features are valuable when collaborative signals are limited. **(4) Impact of Contrastive Learning:** Removing contrastive learning causes 8.6% (All Beauty) and 3.7% (Clothing) drops, showing layer-wise alignment is essential to bridge the semantic-structural

gap. **(5) Impact of Fusion Mechanism:** Without two-stage fusion, performance drops 5.8% and 2.4%, validating adaptive weight computation's importance.

TABLE IV
ABLATION STUDY RESULTS. DROP% SHOWS RELATIVE DECREASE FROM FULL MODEL.

Variant	All Beauty		Clothing		Luxury	
	H@10	N@10	H@10	N@10	H@10	N@10
w/o Global	0.312 (-21.0%)	0.231 (-26.7%)	0.589 (-11.8%)	0.512 (-15.1%)	0.731 (-9.1%)	0.623 (-11.6%)
w/o Domain	0.336 (-14.9%)	0.253 (-19.7%)	0.601 (-10.0%)	0.529 (-12.3%)	0.752 (-6.5%)	0.641 (-9.1%)
w/o LLM	0.345 (-12.7%)	0.250 (-20.6%)	0.638 (-4.5%)	0.530 (-12.1%)	0.755 (-6.1%)	0.641 (-9.1%)
w/o CL	0.361 (-8.6%)	0.276 (-12.4%)	0.643 (-3.7%)	0.555 (-8.0%)	0.775 (-3.6%)	0.661 (-6.2%)
w/o Fusion	0.372 (-5.8%)	0.289 (-8.3%)	0.652 (-2.4%)	0.572 (-5.1%)	0.783 (-2.6%)	0.676 (-4.1%)
Full	**0.395**	**0.315**	**0.668**	**0.603**	**0.804**	**0.705**

E. Cold-Start Performance

Table V evaluates cold-start performance (users/items with <5 interactions), comparing against C2DSR (best graph-based) and LLM-ESR (best LLM-based) baselines.

HDAF-LLM achieves 7.8%–19.1% improvements over LLM-ESR in cold-start scenarios. Gains stand out for cold-start items (16.0%–19.1%), where LLM-generated item descriptions provide semantic information compensating for missing interaction history. For cold-start users, 14.2%–16.7% improvement shows combining LLM-inferred preferences with global graph patterns solves the user cold-start problem.

These results validate our design: (1) higher α weights (0.7) for cold-start entities prioritize LLM features when collaborative signals are unavailable. (2) Global-HGCN's cross-domain connectivity helps cold-start entities leverage related domain information.

TABLE V
COLD-START PERFORMANCE. CS-USER/CS-ITEM: <5 INTERACTIONS. FORMAT: H@10/N@10.

Model	All Beauty		Clothing	
	CS-User	CS-Item	CS-User	CS-Item
C2DSR	0.215/0.142	0.189/0.123	0.452/0.346	0.385/0.289
LLM-ESR	0.253/0.168	0.231/0.152	0.512/0.391	0.446/0.341
HDAF-LLM	**0.289/0.196**	**0.268/0.181**	**0.552/0.423**	**0.491/0.376**
Improv.%	14.2/16.7	16.0/19.1	7.8/8.2	10.1/10.3

F. Design Choice Validation

We validate HDAF-LLM's key design choices via extensive experiments. **Decoupling Strategy:** We compare four decoupling strategies on the All Beauty dataset: shared parameters (HR@10 $= 0.361$), L_2 regularization on cross-domain gradients (HR@10 $= 0.372$), gradient reversal layer (HR@10 $= 0.376$), and our explicit parameter separation (HR@10 $= 0.395$). Our approach outperforms alternatives by 1.9%–3.4% in HR@10, ensuring zero cross-domain gradient interference with Domain-HGCN's separate parameter sets ($\frac{\partial \mathcal{L}_d}{\partial W^l_{d'}} = 0$ for $d \neq d'$).

Negative Sampling Strategy (Table VI): Our single hard negative approach (via augmentation) outperforms both random negatives (even with 256 samples) and multiple hard negatives, as: (1) augmentation-based negatives are domain-relevant and appropriately difficult. (2) our goal is semantic-structural *alignment*, not discrimination, so one carefully constructed negative suffices. (3) our approach is most efficient (86s vs. 312s per epoch).

TABLE VI
NEGATIVE SAMPLING COMPARISON ON ALL BEAUTY.

Strategy	H@10	N@10	Time
Random 64 neg.	0.382	0.291	142s
Random 256 neg.	0.386	0.298	389s
In-batch neg.	0.389	0.302	98s
4 hard neg. (aug.)	0.391	0.308	156s
16 hard neg. (aug.)	0.392	0.310	312s
Single hard neg. (Ours)	**0.395**	**0.315**	**86s**

Table VI shows our single hard negative strategy achieves the best performance and efficiency.

Fusion Strategy (Table VII): We compare five fusion strategies: concatenation, average pooling, single-stage fusion, attention-based fusion, and our two-stage fusion. Our two-stage method outperforms alternatives by 0.61%–3.4% HR@10, with a critical hierarchical design: the first stage aligns semantics and structure (LLM vs. Domain-HGCN), the second balances domain and global signals. Single-stage fusion conflates these gaps, while attention-based fusion lacks explicit density adaptation.

TABLE VII
FUSION STRATEGY COMPARISON ON ALL BEAUTY.

Fusion Strategy	H@10	N@10
Concatenation	0.361	0.276
Average pooling	0.365	0.281
Single-stage fusion	0.372	0.289
Attention-based fusion	0.389	0.305
Two-stage fusion (Ours)	**0.395**	**0.315**

Table VII shows our two-stage fusion outperforms alternatives. **Fusion Loss Target:** We investigate different anchoring targets for the fusion loss on All Beauty. Anchoring to LLM features yields the best performance (H@10=0.395, N@10=0.315), outperforming domain embeddings $\mathbf{h}_d^{(L)}$ (H@10=0.381, N@10=0.298) by 3.7% and global embeddings $\mathbf{g}^{(L)}$ (H@10=0.386, N@10=0.301) by 2.3%. This validates our rationale: LLM embeddings are pre-computed, stable supervision signals. **Contrastive Loss Weight:** Equal weighting (1:1) yields the best performance (H@10=0.395), outperforming 2:1 (H@10=0.389) and 1:2 (H@10=0.388) ratios.

G. Computational Cost

On Clothing, HDAF-LLM has moderate overhead: 5.6M trainable parameters (SASRec 2.1M, C2DSR 4.2M, LLM-ESR 3.8M), 2.3h training (0.8–1.5h for baselines), 12ms inference (5–10ms), 10GB GPU memory (4–8GB). LLM parameters (6B) are offline-precomputed (0.8s per entity, no updates), with

12ms inference latency well within real-time requirements (<100ms).

H. Sensitivity Analysis

Fig. 2 presents sensitivity analysis of key hyperparameters, revealing: (1) **Learning rate:** Optimal $\eta = 10^{-3}$ (smaller slows convergence; larger causes instability). (2) **Contrastive weight λ_1:** Optimal 0.1 (larger over-emphasizes alignment over recommendation accuracy). (3) **GNN layers L:** Optimal 3 (more causes over-smoothing; fewer limits receptive field).

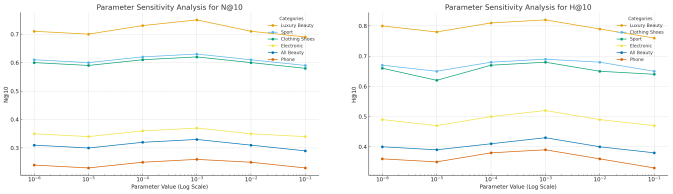

Fig. 2. Parameter Sensitivity Analysis. Left: NDCG@10; Right: HR@10. Performance is stable across a reasonable range of hyperparameters.

Embedding Dimension: Performance improves from $d_0 = 64$ (H@10=0.371) to $d_0 = 256$ (H@10=0.395), plateauing at $d_0 = 512$ (H@10=0.393), indicating 256 dimensions suffice.

V. Conclusion

We propose HDAF-LLM, addressing cross-domain interference via explicit architectural decoupling and semantic-structural misalignment via layer-wise contrastive learning with density-adaptive fusion. Experiments confirm statistically significant improvements, especially in cold-start scenarios.

Limitations: (1) Offline LLM pre-computation may miss rapidly evolving preferences. (2) Zero-interaction cold-start relies on demographic/category data. (3) Validation is limited to Amazon datasets. Future work will explore incremental LLM updates and cross-platform testing.

Acknowledgment

This work was supported by the Natural Science Foundation of Inner Mongolia Autonomous Region (Grant No. 2023LHMS06008) and the Inner Mongolia Key Laboratory of Intelligent Sensing and Systems Engineering (Grant No. 2025KYPT0088).

References

[1] Z. Zhang, S. Liu, J. Yu, Q. Cai, X. Zhao, C. Zhang, Z. Liu, Q. Liu, H. Zhao, L. Hu *et al.*, "M3oe: Multi-domain multi-task mixture-of experts recommendation framework," in *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2024, pp. 893–902.

[2] Y. Li, L. Hou, D. Li, and J. Li, "Hkgcl: Hierarchical graph contrastive learning for multi-domain recommendation over knowledge graph," *Expert Systems with Applications*, vol. 233, p. 120963, 2023.

[3] H. Ju, S. Kang, D. Lee, J. Hwang, S. Jang, and H. Yu, "Multi-domain recommendation to attract users via domain preference modeling," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 8, 2024, pp. 8582–8590.

[4] Z. Song, W. Zhang, L. Deng, J. Zhang, Z. Wu, K. Bian, and B. Cui, "Mitigating negative transfer in cross-domain recommendation via knowledge transferability enhancement," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024, pp. 2745–2754.

[5] H. Ma, R. Xie, L. Meng, X. Chen, X. Zhang, L. Lin, and J. Zhou, "Triple sequence learning for cross-domain recommendation," *ACM Transactions on Information Systems*, vol. 42, no. 4, pp. 1–29, 2024.

[6] W. Xu, Q. Wu, R. Wang, M. Ha, Q. Ma, L. Chen, B. Han, and J. Yan, "Rethinking cross-domain sequential recommendation under open-world assumptions," in *Proceedings of the ACM Web Conference 2024*, 2024, pp. 3173–3184.

[7] Q. Liu, X. Wu, Y. Wang, Z. Zhang, F. Tian, Y. Zheng, and X. Zhao, "Llm-esr: Large language models enhancement for long-tailed sequential recommendation," *Advances in Neural Information Processing Systems*, vol. 37, pp. 26 701–26 727, 2024.

[8] Y. Gao, T. Sheng, Y. Xiang, Y. Xiong, H. Wang, and J. Zhang, "Chat-rec: Towards interactive and explainable llms-augmented recommender system," *arXiv preprint arXiv:2303.14524*, 2023.

[9] Y. Wang, Y. Wang, Z. Fu, X. Li, W. Wang, Y. Ye, X. Zhao, H. Guo, and R. Tang, "Llm4msr: An llm-enhanced paradigm for multi-scenario recommendation," in *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, 2024, pp. 2472–2481.

[10] S. Soundariya, S. Ramakrishnan, S. Devadharshini, S. Madhumita, and L. Promodini, "Cross domain recommendation system using graph embedding and attention techniques," in *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. IEEE, 2024, pp. 1–8.

[11] J. Cao, X. Cong, J. Sheng, T. Liu, and B. Wang, "Contrastive cross-domain sequential recommendation," in *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 2022, pp. 138–147.

[12] J. Hu, W. Xia, X. Zhang, C. Fu, W. Wu, Z. Huan, A. Li, Z. Tang, and J. Zhou, "Enhancing sequential recommendation via llm-based semantic embedding learning," in *Companion Proceedings of the ACM Web Conference 2024*, 2024, pp. 103–111.

[13] A. K. Vajjala, D. F. Meher, S. Pothagoni, Z. Zhu, and D. S. Rosenblum, "Vietoris-rips complex: A new direction for cross-domain cold-start recommendation," in *Proceedings of the 2024 SIAM International Conference on Data Mining (SDM)*. SIAM, 2024, pp. 761–769.

[14] P. Jia, Y. Wang, S. Lin, X. Li, X. Zhao, H. Guo, and R. Tang, "D3: A methodological exploration of domain division, modeling, and balance in multi-domain recommendations," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 8, 2024, pp. 8553–8561.

[15] Z. Fu, X. Li, C. Wu, Y. Wang, K. Dong, X. Zhao, M. Zhao, H. Guo, and R. Tang, "A unified framework for multi-domain ctr prediction via large language models," *ACM Transactions on Information Systems*, vol. 43, no. 5, pp. 1–33, 2025.

[16] K. Bao, J. Zhang, Y. Zhang, W. Wang, F. Feng, and X. He, "Tallrec: An effective and efficient tuning framework to align large language model with recommendation," in *Proceedings of the 17th ACM conference on recommender systems*, 2023, pp. 1007–1014.

[17] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," *arXiv preprint arXiv:1511.06939*, 2015.

[18] W.-C. Kang and J. McAuley, "Self-attentive sequential recommendation," in *2018 IEEE international conference on data mining (ICDM)*. IEEE, 2018, pp. 197–206.

[19] P. Li and A. Tuzhilin, "Ddtcdr: Deep dual transfer cross domain recommendation," in *Proceedings of the 13th international conference on web search and data mining*, 2020, pp. 331–339.