

Article

HGCRec: A Heterogeneous Graph Contrastive Learning Framework for Cold-Start Multi-Modal Recommendation in Intelligent Information Systems

Huayu Li * and Yangchen Xu

Qingdao Institute of Software, College of Computer Science and Technology, China University of Petroleum (East China), Qingdao 266580, China; z23070033@s.upc.edu.cn

* Correspondence: lhyzj@upc.edu.cn

Abstract

Personalized recommendation has become an essential component of intelligent information systems and electronic multimedia platforms. However, cold-start items with limited user–item interactions remain difficult to model, especially when collaborative signals are sparse and heterogeneous side information is underutilized. To address this problem, this paper proposes Heterogeneous Graph Contrastive Recommendation (HGCRec), a heterogeneous graph contrastive learning framework for cold-start multi-modal recommendation. HGCRec constructs a heterogeneous structural graph containing users, items, categories, brands, and semantic entities, while visual and textual information is incorporated as item-side feature views. A relation-aware heterogeneous graph encoder captures relation-specific structural semantics, and a cross-modal graph contrastive learning objective coordinates structural, visual, and textual item representations. An interaction-sparsity-aware weighting strategy allocates stronger contrastive supervision to items with fewer training interactions. Furthermore, an item-specific adaptive fusion module integrates collaborative, structural, visual, and textual representations according to interaction sparsity and learned multi-source representation states. Experiments are conducted on three Amazon multi-modal recommendation datasets, including Baby, Sports, and Clothing. The results show that HGCRec consistently outperforms representative graph-based and multi-modal recommendation baselines. Compared with the strongest baseline, Freezing and Denoising Graph Structures for Multimodal Recommendation (FREEDOM), HGCRec improves Recall@20 by 9.65%, 10.44%, and 12.00% on Baby, Sports, and Clothing, respectively, and improves NDCG@20 by 11.14%, 11.35%, and 13.10%. Sparsity-aware analysis further shows larger relative improvements for items with extremely limited training interactions, demonstrating the effectiveness of HGCRec under the evaluated interaction-sparse and few-shot settings.



Academic Editor: George Angelos Papadopoulos

Received: 3 July 2026

Revised: 12 August 2026

Accepted: 12 August 2026

Published: 14 August 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: cold-start recommendation; multi-modal recommendation; heterogeneous graph learning; graph contrastive learning; adaptive fusion; intelligent information systems

1. Introduction

Artificial intelligence (AI)-driven intelligent information systems and electronic multimedia platforms increasingly rely on personalized recommendation techniques to improve information access efficiency and user experience. In practical e-commerce, content-sharing, and multimedia service platforms, user preference modeling commonly draws on heterogeneous behavioral, relational, visual, and textual information. Traditional collaborative

filtering methods mainly rely on historical user–item interactions to learn latent representations and generate recommendation lists. However, a large number of items are newly released, rarely interacted with, or located in the long-tail distribution, resulting in insufficient collaborative signals for reliable representation learning.

In this study, the item-side cold-start setting is defined from the perspective of training interaction sparsity. Specifically, items with zero training interactions are regarded as zero-direct-interaction items, whereas items with one or two training interactions are regarded as few-shot or extremely sparse items.

Graph-based collaborative filtering methods, such as the Light Graph Convolutional Network (LightGCN) [1], simplify graph convolution for recommendation by propagating user and item embeddings over the interaction graph, while knowledge-aware models such as the Knowledge Graph Attention Network (KGAT) [2] and the Knowledge-Graph-based Intent Network (KGIN) [3] further introduce knowledge graph connectivity to enhance high-order semantic reasoning. These advances motivate the use of complementary semantic and multi-modal evidence to enrich interaction-sparse item representations when collaborative connections are limited. Meanwhile, heterogeneous side information differs in relation type, semantic content, coverage, and representation space, which requires relation-aware modeling and coordinated multi-modal integration.

Graph-based modeling is particularly suitable for this data structure because users, items, categories, brands, and semantic entities naturally form typed objects connected through semantically distinct relations. Relation-aware message passing preserves graph topology and relation semantics and enables interaction-sparse items to aggregate collaborative and structural evidence from direct and high-order neighborhoods. Visual and textual content further supplies complementary item-level evidence through modality-specific feature representations.

Beyond recommendation systems, graph-theory-based learning has also been applied to large-scale, multi-dimensional, and uncertain non-physical systems. Tutsoy [4] formulated epidemiological modeling as a graph-based constrained parameter-learning problem, in which structural coupling and parameter constraints were incorporated into gradient-based and meta-heuristic optimization algorithms. This study demonstrates that graph structures and explicit constraints can organize complex interactions and support the optimization of large-scale coupled systems. Following this general principle, HGCRRec represents heterogeneous recommendation entities and semantic relations in a unified graph and jointly optimizes relation-aware transformations, cross-modal alignment, and adaptive fusion. Relation-wise sparse message passing and mini-batch optimization further provide an algorithmic basis for extending the framework to larger recommendation graphs.

From the perspective of information acquisition, structural, visual, and textual information exhibit distinct physical and digital properties. Structural information consists of digitally recorded behavioral and relational signals, including user–item interactions and item associations with categories, brands, and semantic entities. Visual information originates from product images and digitally represents observable physical characteristics, such as appearance, color, shape, and style. Textual information is obtained from product titles and descriptions and represents functional, contextual, and semantic attributes.

Before fusion, the reliability of these information sources is evaluated separately. Structural information is characterized using relation coverage, user and item degree distributions, and graph density. Visual and textual representations are assessed using modality coverage, missing-feature ratios, zero-feature ratios, and the mean and standard deviation of feature norms. Cross-modal cosine similarity is further examined to identify strong inconsistencies between visual content and textual descriptions. After quality screening, modality-specific features are normalized and projected into a shared latent space.

In the recommendation process, structural information supports relational preference propagation, visual information captures perceptual similarity, and textual information provides functional and semantic evidence, especially for interaction-sparse items [5].

To alleviate the sparsity issue, multi-modal recommendation has attracted increasing attention because item visual and textual contents can provide complementary semantic evidence beyond user–item interactions. Early studies such as Visual Bayesian Personalized Ranking (VBPR) [6] incorporate visual features to improve implicit-feedback recommendation, while Multimodal Attentive Metric Learning (MAML) [7] models user-specific attention over multi-modal item features to capture diverse preferences. With the development of graph neural networks, the Multi-modal Graph Convolution Network (MMGCN) [8] constructs modality-specific user–item graphs to learn modality-aware user preferences, and the Dual Graph Neural Network (DualGNN) [9] introduces dual graph modeling to exploit both user–item interactions and user co-occurrence relations for multimedia recommendation. Subsequent studies further improve graph-based multi-modal representation learning by refining noisy interaction graphs or mining latent semantic structures. For instance, the Graph-Refined Convolutional Network (GRCN) [10] adaptively refines the interaction graph to reduce the influence of false-positive implicit feedback, whereas Latent Structure Mining for Multimodal Recommendation (LATTICE) [11] mines latent item–item structures from multi-modal features to enhance item representations through graph convolution.

Existing multi-modal recommendation methods provide a strong methodological foundation, while three specific issues remain central to interaction-sparse item recommendation. First, many methods perform representation-level fusion through concatenation, addition, or coarse-grained attention. A relation-aware mechanism can further distinguish user–item interactions, item–category associations, item–brand relations, and item–entity links during transformation and aggregation, thereby preserving the semantic contribution of each relation type.

Second, knowledge-aware recommendation methods primarily emphasize structural propagation over interaction graphs and semantic graphs. An explicit cross-modal consistency objective can further coordinate the structural, visual, and textual representations of the same item and provide stable representation regularization for interaction-sparse items.

Third, contrastive recommendation methods improve the general quality and robustness of multi-modal representations. For example, MICRO [12] introduces contrastive modality fusion for multimedia recommendation, BM3 [13] bootstraps latent representations through self-supervised reconstruction, FREEDOM [14] combines interaction-graph denoising with frozen item–item structures, and MMSSL [15] learns modality-aware self-supervised signals. Item-specific collaborative sufficiency and side-information quality vary substantially across the training-interaction-sparsity spectrum. This variation motivates a unified mechanism that combines relation-aware propagation, cross-modal alignment, and item-specific information allocation.

To tackle these challenges, this paper proposes HGCRRec, a Heterogeneous Graph contrastive learning framework for item-side multi-modal recommendation under training interaction sparsity in intelligent information systems.

HGCRRec constructs an interaction-sparsity-oriented heterogeneous graph containing users, items, knowledge entities, categories, and brands as explicit node types. To maintain symbolic and implementation consistency, visual and textual information is attached to item nodes as modality-specific features during projection, contrastive alignment, and fusion. Categories, brands, and knowledge entities participate as explicit semantic nodes in heterogeneous relational message propagation. The graph relations include user–item interactions, item–entity links, item–category associations, item–brand links, and available

item–item semantic relations. This construction enables interaction-sparse items to connect with the existing preference space through semantic neighbors and multi-modal features.

Based on the constructed heterogeneous graph, HGCRec designs a relation-aware heterogeneous graph encoder to model different semantic relations separately. Each relation type is assigned relation-specific transformation parameters and attention coefficients, enabling user–item, item–entity, item–category, item–brand, and item–item semantic neighbors to contribute through distinct propagation channels. This design preserves relation heterogeneity and controls semantic interference during neighborhood aggregation.

Furthermore, HGCRec introduces a cross-modal graph contrastive learning objective to align structural, visual, and textual representations of the same item. Positive pairs are constructed from different views of the same item, while negative pairs are sampled from other items in the mini-batch. Inspired by graph contrastive recommendation studies such as SGL [16] and NCL [17], the contrastive objective serves as an auxiliary representation regularizer alongside the recommendation loss.

HGCRec further develops an item-adaptive fusion mechanism that estimates item-specific contribution weights for collaborative, structural, visual, and textual representations according to the available interaction evidence and learned multi-source representation states. The mechanism is designed to emphasize semantic and modality evidence for interaction-sparse items and to increase the contribution of collaborative representations as interaction evidence becomes richer.

The experimental analysis directly examines this adaptive behavior by recording λ_i^{CF} , λ_i^S , λ_i^V , and λ_i^T after model convergence and comparing their mean values and distributions across training-degree groups. The analysis includes the 0–2 interaction group and the group with at least 50 training interactions, together with correlation statistics between item degree and the learned fusion weights. Static, uniform, and item-adaptive fusion variants further separate the effect of dynamic weight allocation from the effects of heterogeneous encoding and cross-modal contrastive learning.

From the learning perspective, HGCRec adopts an end-to-end hybrid learning paradigm that combines weakly supervised recommendation learning with self-supervised cross-modal representation learning. Observed user–item interactions provide pairwise ranking supervision, while the structural, visual, and textual representations of the same item provide self-supervised cross-modal alignment signals. Relation-specific transformations, attention coefficients, modality projection parameters, and adaptive fusion parameters are jointly learned through gradient-based optimization. The intelligence of HGCRec is reflected in its ability to learn user preferences, distinguish heterogeneous semantic relations, align multi-modal representations, and allocate information sources according to item-specific representation conditions.

The final user–item preference score is optimized with a recommendation loss and cross-modal contrastive regularization. The main contributions of this paper are summarized as follows:

- We propose an interaction-sparsity-oriented heterogeneous graph construction strategy that represents users, items, knowledge entities, categories, and brands as explicit nodes and attaches visual and textual information to item nodes as modality-specific features. This strategy enables interaction-sparse items to obtain semantic and content evidence beyond direct user–item interactions.
- We design a relation-aware heterogeneous graph encoder that performs relation-specific transformation and attention aggregation over different edge types, thereby preventing heterogeneous semantic neighbors from being indiscriminately mixed.
- We introduce a cross-modal contrastive and item-adaptive fusion mechanism that aligns structural, visual, and textual item views and learns item-specific contribution

weights for collaborative, semantic, and modality information across different levels of training interaction sparsity.

- We provide direct empirical evidence for the adaptive fusion process through group-wise fusion-weight visualization, degree–weight correlation analysis, comparisons with static fusion variants, and paired significance tests against the strongest baseline.

The remainder of this paper is organized as follows. Section 2 reviews graph-based, knowledge-aware, multi-modal, and contrastive recommendation studies. Section 3 formulates the item-side multi-modal recommendation problem under training interaction sparsity, including strict zero-interaction cold-start and few-shot cases. Section 4 presents the proposed HGCRec framework in detail. Section 5 reports the experimental settings and results. Section 6 concludes the paper and discusses future research directions.

2. Related Work

2.1. Graph-Based and Knowledge-Aware Recommendation

Graph-based recommendation methods have been widely studied because user–item interactions naturally form bipartite graph structures. Compared with traditional collaborative filtering methods, graph neural networks can propagate preference signals over high-order neighbors, thereby capturing collaborative relationships that are difficult to model through shallow latent factor methods. Several surveys have systematically summarized the development of graph neural networks in recommender systems, including graph construction, message propagation, aggregation mechanisms, and optimization strategies [18,19]. Based on this research line, recent graph collaborative filtering methods have achieved competitive performance by simplifying graph convolution, introducing self-supervised learning signals, or enhancing neighborhood representation. However, most graph collaborative filtering models still mainly depend on user–item interaction graphs. When items have very few historical interactions, especially in cold-start or long-tail scenarios, the learned item embeddings may be unreliable because they cannot receive sufficient collaborative messages from the interaction graph.

To alleviate data sparsity and cold-start problems, knowledge-aware recommendation methods introduce external knowledge graphs or item-related semantic relations into recommendation models. Knowledge graphs provide structured side information, such as item attributes, categories, brands, entities, and relational paths, which can connect sparse items with semantically related objects. Knowledge-graph-based recommender systems can be roughly divided into embedding-based, path-based, and propagation-based methods [20]. Early propagation-based methods, such as RippleNet [21], propagate user preferences over knowledge graphs by expanding users’ potential interests from items they have historically interacted with. KGCN [22] and KGNN-LS [23] further apply graph neural networks to knowledge-aware recommendation by aggregating neighborhood information from knowledge graphs. Path-based methods, such as KPRN [24], model the sequential dependencies of entities and relations along user–item paths to enhance both recommendation accuracy and explainability. In addition, attention-based models, such as KGAT [2] and conditional graph attention networks [25], assign different importance to entities and relations during knowledge propagation.

Although knowledge-aware recommendation methods can enrich item representations, they still have several limitations for cold-start multi-modal recommendation. First, many methods focus on entity relations or path reasoning but do not sufficiently exploit visual and textual item contents. Second, different relations, such as user–item interaction, item–category relation, item–brand relation, and item–entity association, usually have different semantic functions. Simply aggregating them in a unified graph may introduce semantic noise. Third, knowledge-aware propagation does not explicitly align structural

semantics with visual and textual modality representations. Some recent studies begin to incorporate multi-modal knowledge graphs into recommendation [26], while heterogeneous graph attention methods such as DisenHAN [27] further show that relation heterogeneity is important for recommendation. However, existing methods still lack a cold-start-oriented heterogeneous graph construction strategy that jointly considers user–item interactions, semantic entities, item attributes, and multi-modal representations. Therefore, relation-aware heterogeneous graph encoding is required to provide more reliable semantic support for sparse items.

In summary, existing graph-based and knowledge-aware recommendation methods primarily emphasize collaborative propagation, knowledge-path modeling, or entity-level aggregation. Their remaining challenges concentrate on the differentiated modeling of heterogeneous relation semantics, the coordination of structural and multi-modal information, and semantic support for interaction-sparse items. HGCRec addresses these challenges through heterogeneous graph construction, relation-aware message aggregation, and cross-modal representation alignment.

2.2. Multi-Modal and Contrastive Recommendation

Multi-modal recommendation aims to improve recommendation performance by incorporating item contents from different modalities, such as images, textual descriptions, videos, and acoustic signals. Compared with collaborative filtering models that only rely on interaction records, multi-modal methods can provide additional semantic evidence for user preference modeling. Recent surveys on multi-modal and deep-learning-based recommender systems have summarized major technical directions, including modality encoder design, feature interaction, feature enhancement, and model optimization [28,29]. Early methods, such as VBPR [6], introduce visual features into personalized ranking, while attention-based methods, such as MAML [7], model user-specific modality preferences. These methods demonstrate the usefulness of multi-modal information, but they usually treat modality features as auxiliary item attributes and combine them with ID embeddings through concatenation, weighted summation, or attention mechanisms.

With the development of graph neural networks, many studies have attempted to integrate graph learning with multi-modal recommendation. MMGCN [8] constructs modality-specific user–item graphs to learn modality-aware user and item representations. DualGNN [9] jointly models user–item interactions and user co-occurrence relations to enhance multimedia recommendation. GRCN [10] introduces a graph refinement module to reduce the negative influence of noisy implicit feedback. LATTICE [11] mines latent item–item structures from multi-modal features and uses graph convolution to enhance item representations. Furthermore, multi-modal knowledge graph methods [26] construct semantic graphs from visual, textual, and relational information, while disentangled multi-modal representation learning methods attempt to separate different latent factors in user preferences [30]. These methods improve multi-modal representation learning from different perspectives. However, most of them mainly focus on general recommendation performance and do not explicitly consider how cold-start items should adaptively rely on multi-modal contents and semantic relations when collaborative signals are insufficient.

Contrastive learning has recently become an important technique for recommendation because it can improve representation robustness by maximizing the agreement between different views. In graph-based recommendation, SGL [16] constructs augmented graph views to enhance collaborative filtering representations, while NCL [17] introduces neighborhood-enriched contrastive signals from graph structure and semantic space. In multi-modal recommendation, contrastive learning has also been used to align modality-specific and modality-shared representations. For example, MGCL [31] captures collaborative sig-

nals from user–item interactions and conducts multi-modal graph contrastive learning to enhance multimedia recommendation. BM3 [13] bootstraps latent representations to simplify multi-modal recommendation without relying heavily on negative sampling. FREEDOM [14] freezes item–item graph structures and denoises user–item interaction graphs to improve robustness. MMSSL [15] designs modality-aware self-supervised signals to model cross-modal dependencies. Multi-view graph convolutional networks further separate behavior and modality views to reduce modality noise and improve preference modeling [32].

Despite these advances, existing multi-modal and contrastive recommendation methods still do not fully solve the cold-start problem. Some cold-start recommendation studies use contrastive learning to preserve collaborative signals in content representations [33], while contrastive collaborative filtering methods further exploit co-occurrence signals from warm items to improve cold-start item representation [34]. However, these methods usually focus on either content-collaboration alignment or collaborative signal transfer rather than jointly modeling heterogeneous semantic relations, multi-modal representations, and cold-start adaptive fusion. In real-world recommendation scenarios, an item with few interactions should rely more on category, brand, entity, image, and textual information, whereas a warm item with sufficient interactions can depend more on collaborative signals. This indicates that fixed fusion or global modality weighting is not suitable for items with different sparsity levels.

In summary, existing multi-modal and contrastive recommendation methods provide effective content representation and cross-view supervision, while cold-start recommendation further requires the joint utilization of heterogeneous structural semantics, modality-specific information, and item interaction sparsity. HGCRec combines cross-modal contrastive alignment, interaction-sparsity-aware contrastive weighting, and item-specific adaptive fusion to coordinate these complementary information sources.

3. Preliminaries and Problem Definition

This section introduces the mathematical notation and formulates the cold-start multi-modal recommendation problem. We define the interaction data, heterogeneous graph structure, item interaction sparsity, top- K recommendation task, and cross-modal contrastive learning objective. The concrete graph instantiation and representation-learning process are presented in Section 4.

Definition 1. *Heterogeneous Recommendation Graph and Multi-modal Features.*

Let $\mathcal{U}$ and $\mathcal{I}$ denote the sets of users and items, respectively. The implicit feedback data are represented by a user–item interaction matrix $\mathbf{Y} \in \{0, 1\}^{|\mathcal{U}| \times |\mathcal{I}|}$, where

$$y_{ui} = \begin{cases} 1, & \text{if user } u \text{ has interacted with item } i, \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

For multi-modal recommendation, let $\mathcal{M} = \{V, T\}$ denote the modality set, where V and T represent visual and textual modalities, respectively. The original visual and textual features of item i are denoted by $\mathbf{x}_i^V \in \mathbb{R}^{d_V}$ and $\mathbf{x}_i^T \in \mathbb{R}^{d_T}$.

Let $\mathcal{C}$, $\mathcal{B}$, and $\mathcal{E}$ denote the sets of category, brand, and semantic-entity nodes, respectively. The heterogeneous structural graph is defined as

$$\mathcal{G} = (\mathcal{V}, \mathcal{A}, \mathcal{R}), \quad (2)$$

where $\mathcal{V}$, $\mathcal{A}$, and $\mathcal{R}$ denote the node set, edge set, and relation-type set, respectively. The node set is defined as

$$\mathcal{V} = \mathcal{U} \cup \mathcal{I} \cup \mathcal{C} \cup \mathcal{B} \cup \mathcal{E}. \quad (3)$$

The relation-type set is defined as

$$\mathcal{R} = \{r_{ui}, r_{iu}, r_{ic}, r_{ci}, r_{ib}, r_{bi}, r_{ie}, r_{ei}\} \cup \mathcal{R}_{ii}, \quad (4)$$

where r_{ui} , r_{ic} , r_{ib} , and r_{ie} denote user–item, item–category, item–brand, and item–entity relations, respectively, and the corresponding reverse relations are represented by r_{iu} , r_{ci} , r_{bi} , and r_{ei} . $\mathcal{R}_{ii}$ denotes the available item–item semantic relation types. Visual and textual information is represented by the item-side feature views $\mathbf{x}_i^V$ and $\mathbf{x}_i^T$.

Definition 2. *Cold-Start and Sparsity-Aware Setting.*

Cold-start recommendation refers to the situation where users or items have insufficient historical interactions. This paper mainly focuses on item-side cold-start recommendation. For each item i , its training interaction degree is defined as

$$d_i^{tr} = \sum_{u \in \mathcal{U}} y_{ui}^{tr}, \quad (5)$$

where y_{ui}^{tr} denotes the interaction indicator in the training set.

All item-degree statistics and sparsity groups are calculated exclusively from the training split. In particular, $d_i^{tr} = 0$ identifies an item with zero direct user–item interactions in the training graph. Such an item can remain connected to the heterogeneous training graph through available category, brand, entity, or item–item semantic relations.

Following the sparsity-aware evaluation protocol, we adopt a progressively widening degree partition based on the empirical distribution of item interaction degrees in the training set after preprocessing. The boundaries 2, 4, 9, 19, and 49 are used to distinguish extremely sparse, sparse, low-frequency, medium-frequency, high-frequency, and warm or popular items, respectively. Finer intervals are retained in the low-degree region because performance differences are more sensitive to small changes in interaction frequency, whereas wider intervals are used in the high-degree region to maintain sufficient sample coverage and improve the stability of subgroup evaluation. These boundaries are fixed before test evaluation and are applied consistently to all compared methods. Items are divided into the following non-overlapping groups:

$$g(i) = \begin{cases} G_1, & 0 \leq d_i^{tr} \leq 2, \\ G_2, & 3 \leq d_i^{tr} \leq 4, \\ G_3, & 5 \leq d_i^{tr} \leq 9, \\ G_4, & 10 \leq d_i^{tr} \leq 19, \\ G_5, & 20 \leq d_i^{tr} \leq 49, \\ G_6, & d_i^{tr} \geq 50. \end{cases} \quad (6)$$

The numerical constants in Equation (6) serve as evaluation boundaries for sparsity-aware analysis, while the recommendation representations and fusion weights are learned through end-to-end optimization. In the reported experiments, d_i^{tr} is calculated from the fixed training snapshot to ensure reproducible comparisons.

G_1 represents the extremely sparse or few-shot group and contains items with zero, one, or two training interactions.

The cold-start item set is defined as

$$\mathcal{I}_{cs}(\delta) = \{i \in \mathcal{I} \mid d_i^{tr} \leq \delta\}, \quad (7)$$

where δ is determined from the training-degree partition and fixed before validation and test evaluation. In the reported experiments, $\delta = 2$ is adopted, corresponding to group G_1 , which contains items with no more than two training interactions.

The zero-interaction subset is further denoted by $\mathcal{I}_0 = \{i \in \mathcal{I} \mid d_i^{tr} = 0\}$. Accordingly, $\mathcal{I}_0$ characterizes the zero-direct-interaction setting, while $\mathcal{I}_{cs}(2)$ represents the broader extremely sparse cold-start setting used in the sparsity-aware evaluation.

Definition 3. *Top-K Recommendation Task.*

Given the training graph $\mathcal{G}^{tr}$, user-item interactions, item modality features, and semantic relations, the goal is to learn a scoring function

$$\hat{y}_{ui} = f_{\Theta}(u, i \mid \mathcal{G}^{tr}, \mathbf{x}_i^V, \mathbf{x}_i^T), \quad (8)$$

where Θ denotes the trainable model parameters and $\hat{y}_{ui}$ represents the predicted preference score of user u toward item i . For each user u , the candidate set for evaluation is defined as

$$\tilde{\mathcal{I}}_u = \mathcal{I} \setminus \mathcal{I}_u^{tr}, \quad (9)$$

where $\mathcal{I}_u^{tr}$ denotes the set of items interacted with by user u in the training set.

The recommendation model ranks candidate items according to $\hat{y}_{ui}$ and returns the top- K item list $\mathcal{L}_u^K$. Recommendation quality is evaluated using full-ranking metrics, including Recall@K and NDCG@K. Recall@K is defined as

$$\text{Recall@K} = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{|\mathcal{L}_u^K \cap \mathcal{I}_u^{te}|}{|\mathcal{I}_u^{te}|}. \quad (10)$$

NDCG@K is defined as

$$\text{NDCG@K} = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{1}{Z_u} \sum_{p=1}^K \frac{\mathbf{1}(i_p \in \mathcal{I}_u^{te})}{\log_2(p+1)}. \quad (11)$$

Here, $\mathcal{I}_u^{te}$ denotes the test item set of user u , i_p is the item ranked at position p , $\mathbf{1}(\cdot)$ is the indicator function, and Z_u is the ideal discounted cumulative gain for normalization. The full-ranking protocol uses the complete candidate set $\tilde{\mathcal{I}}_u$ for evaluation.

Definition 4. *Cross-Modal Graph Contrastive Learning.*

Graph contrastive learning learns discriminative representations by maximizing agreement between positive pairs and increasing representation discrimination across items [16,17,35]. Each item is represented through the heterogeneous structural, visual, and textual views. Their representations are denoted by $\mathbf{z}_i^S$, $\mathbf{z}_i^V$, and $\mathbf{z}_i^T$, respectively.

The visual and textual views represent item-side feature representations.

For any two views $a, b \in \{S, V, T\}$ with $a \neq b$, representations of the same item under views a and b form a positive pair, while representations of other items in the same mini-batch provide comparison samples. The view-level contrastive loss is defined as

$$\mathcal{L}_{a,b} = -\frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \log \frac{\exp(\text{sim}(\mathbf{z}_i^a, \mathbf{z}_i^b)/\tau)}{\sum_{j \in \mathcal{B}} \exp(\text{sim}(\mathbf{z}_i^a, \mathbf{z}_j^b)/\tau)}. \quad (12)$$

In Equation (12), $\mathcal{B}$ denotes a mini-batch of items, $\tau > 0$ is the temperature parameter controlling the concentration of the similarity distribution, and $\text{sim}(\cdot, \cdot)$ denotes cosine similarity, defined as

$$\text{sim}(\mathbf{p}, \mathbf{q}) = \frac{\mathbf{p}^T \mathbf{q}}{\|\mathbf{p}\|_2 \|\mathbf{q}\|_2}. \quad (13)$$

Accordingly, the overall cross-modal contrastive objective is formulated as

$$\mathcal{L}_{CL} = \lambda_{V,T} \mathcal{L}_{V,T} + \lambda_{S,V} \mathcal{L}_{S,V} + \lambda_{S,T} \mathcal{L}_{S,T}, \quad (14)$$

where $\lambda_{V,T}$, $\lambda_{S,V}$, and $\lambda_{S,T}$ specify the global regularization strengths of visual–textual, structural–visual, and structural–textual alignment, respectively. The visual–textual term captures shared content semantics, while the structural–visual and structural–textual terms co-ordinate modality-specific representations with the heterogeneous structural representation.

For any trainable parameter $\theta \in \Theta$, the gradient contributed by the cross-modal objective is expressed as

$$\frac{\partial \mathcal{L}_{CL}}{\partial \theta} = \lambda_{V,T} \frac{\partial \mathcal{L}_{V,T}}{\partial \theta} + \lambda_{S,V} \frac{\partial \mathcal{L}_{S,V}}{\partial \theta} + \lambda_{S,T} \frac{\partial \mathcal{L}_{S,T}}{\partial \theta}. \quad (15)$$

During each mini-batch, the three alignment losses are weighted and summed, and the resulting $\mathcal{L}_{CL}$ is jointly optimized with the recommendation loss through back-propagation. Each coefficient scales the gradient contribution of its corresponding alignment path. The resulting gradients jointly update the relation-aware graph encoder, modality-specific projection networks, and the shared representation parameters involved in the corresponding cross-modal pair.

The three coefficients are selected according to validation NDCG@20 and remain fixed within each training run to support stable and reproducible optimization. In the reported experiments, $\lambda_{V,T}$, $\lambda_{S,V}$, and $\lambda_{S,T}$ are set to 0.2, 0.3, and 0.4, respectively.

4. Methodology

This section presents the proposed HGCRec framework for cold-start multi-modal recommendation. Based on the mathematical formulation in Section 3, HGCRec combines heterogeneous structural relations with visual and textual item features. The framework consists of four major components: heterogeneous graph construction, relation-aware heterogeneous graph encoding, cross-modal contrastive learning, and cold-start adaptive fusion with joint optimization.

4.1. Overall Framework and Heterogeneous Graph Construction

The overall framework of HGCRec is shown in Figure 1. The model takes user–item interactions, item visual features, item textual features, and item metadata as inputs.

The heterogeneous graph defined in Equation (2) is instantiated from five principal node types: users, items, categories, brands, and semantic entities. A user–item edge corresponds to an observed interaction in the training data. Item–category and item–brand edges are obtained from item metadata, while item–entity edges connect items with their associated semantic entities. Available also-buy and also-viewed records are incorporated through $\mathcal{R}_{ii}$ as additional item–item semantic relations. Reverse relations support bidirectional relation-aware propagation.

The different relation types provide complementary structural information. User–item relations encode observed behavioral preferences, while category, brand, entity, and item–item semantic relations describe attribute-level and semantic associations among items. These relations jointly form the structural information channel processed by the relation-

aware heterogeneous graph encoder. Visual and textual information provides two item-side content channels that characterize perceptual and semantic properties, respectively.

Before graph construction, duplicate interactions, invalid identifiers, and relations with incomplete endpoints are filtered during preprocessing. The resulting user–item edges are generated exclusively from the training interaction split, while category, brand, entity, and item–item semantic relations are constructed from the metadata available for model training.

For visual and textual information, a modality-availability indicator is defined as

$$a_i^m = \begin{cases} 1, & \text{if } \mathbf{x}_i^m \text{ is available, finite, and } \|\mathbf{x}_i^m\|_2 > \epsilon, \\ 0, & \text{otherwise,} \end{cases} \quad m \in \{V, T\}. \quad (16)$$

Here, $\epsilon > 0$ is a numerical-stability threshold for identifying all-zero or numerically degenerate feature vectors. The threshold is fixed before model training according to floating-point precision and applied consistently across all datasets and data splits. Valid modality features are normalized as

$$\bar{\mathbf{x}}_i^m = \frac{a_i^m \mathbf{x}_i^m}{\|\mathbf{x}_i^m\|_2 + \epsilon}, \quad m \in \{V, T\}. \quad (17)$$

The availability indicator serves as the modality mask in subsequent representation learning. The normalized visual and textual features provide the inputs to their corresponding modality-specific projection networks.

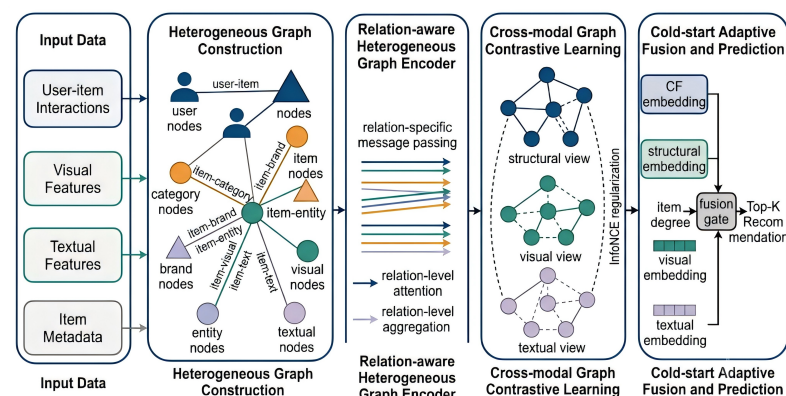


Figure 1. Overall framework of HGCRec, including heterogeneous information inputs, relation-aware graph encoding, cross-modal contrastive alignment, and adaptive multi-source fusion.

For each user $u \in \mathcal{U}$ and item $i \in \mathcal{I}$, trainable ID embeddings are initialized as $\mathbf{e}_u^{(0)} \in \mathbb{R}^d$ and $\mathbf{e}_i^{(0)} \in \mathbb{R}^d$. Trainable embeddings are also assigned to category, brand, and semantic-entity nodes.

The normalized visual and textual features are mapped into the common d -dimensional latent space through modality-specific projection networks:

$$\begin{aligned} \tilde{\mathbf{x}}_i^m &= a_i^m \phi_m(\bar{\mathbf{x}}_i^m) \\ &= a_i^m [\mathbf{W}_2^m \sigma(\mathbf{W}_1^m \bar{\mathbf{x}}_i^m + \mathbf{b}_1^m) + \mathbf{b}_2^m], \quad m \in \{V, T\}. \end{aligned} \quad (18)$$

In Equation (18), $\phi_m(\cdot)$ denotes the projection network for modality m , $\sigma(\cdot)$ denotes the nonlinear activation function, and $\tilde{\mathbf{x}}_i^m \in \mathbb{R}^d$ denotes the projected modality representation. $\mathbf{W}_1^m$, $\mathbf{W}_2^m$, $\mathbf{b}_1^m$, and $\mathbf{b}_2^m$ are modality-specific trainable parameters. The projected representations $\tilde{\mathbf{x}}_i^V$ and $\tilde{\mathbf{x}}_i^T$ constitute the visual and textual feature views associated with item i .

Heterogeneous graph construction provides interaction-sparse items with additional structural connections through shared metadata and semantic relations. An item with

limited direct user–item interactions can share a category, brand, semantic entity, also-buy relation, or also-viewed relation with other items in the training graph. Relation-aware propagation transfers structural information along these semantic paths, while the visual and textual feature views provide complementary item-level content evidence.

For items with $d_i^{tr} = 0$, structural representation learning proceeds through the available metadata and semantic connections in the training graph. Shared category, brand, entity, and item–item relations enable information propagation between zero-direct-interaction items and other structurally related items. All propagation paths are determined by the training graph and the metadata available at training time, while validation and test interactions are reserved for model selection and evaluation.

Accordingly, HGCRec combines heterogeneous structural propagation with visual and textual content representation to support interaction-sparse items. The structural, visual, and textual representations are subsequently coordinated through cross-modal contrastive alignment and adaptive multi-source fusion.

4.2. Relation-Aware Heterogeneous Graph Encoder

After constructing the heterogeneous recommendation graph, HGCRec employs a relation-aware heterogeneous graph encoder to learn user and item structural representations. Different relation types describe complementary semantics: user–item relations represent behavioral preferences, item–category and item–brand relations characterize attribute associations, item–entity relations provide semantic connections, and also-bought or also-viewed relations capture item-level associations. HGCRec therefore performs relation-specific transformation and attention-based aggregation to preserve these semantic distinctions during graph propagation.

The encoder follows four sequential operations at each graph layer: relation-specific message transformation, intra-relation neighbor attention, relation-level fusion, and residual node updating. Algorithm 1 provides a high-level overview of this layer-wise procedure before the detailed mathematical formulation.

Algorithm 1: Relation-Aware Heterogeneous Graph Encoding

Input: Graph $\mathcal{G}$, initial embeddings $\mathbf{H}^{(0)} = \{\mathbf{h}_i^{(0)}\}_{i \in \mathcal{V}}$, and layers L
Output: User representations $\{\mathbf{e}_u\}_{u \in \mathcal{U}}$ and structural item representations $\{\mathbf{z}_i^S\}_{i \in \mathcal{I}}$

- 1: **for** $l = 0$ to $L - 1$ **do**
- 2: **for each** node $i \in \mathcal{V}$ **do**
- 3: Determine the valid relation set $\mathcal{R}_i = \{r \in \mathcal{R} \mid \mathcal{N}_i^r \neq \emptyset\}$;
- 4: **for each** relation $r \in \mathcal{R}_i$ **do**
- 5: Transform neighbor messages using $\mathbf{m}_{j \rightarrow i}^{(r,l)} = \mathbf{W}_r^{(l)} \mathbf{h}_j^{(l)}$;
- 6: Compute relation-specific neighbor attention scores $e_{ij}^{(r,l)}$;
- 7: Normalize the attention scores to obtain $\alpha_{ij}^{(r,l)}$;
- 8: Aggregate the relation-specific neighborhood representation $\mathbf{h}_{i,r}^{(l+1)}$;
- 9: Compute the relation-level importance score $q_{i,r}^{(l)}$;
- 10: **end for**
- 11: Normalize the relation-level scores to obtain $\beta_{i,r}^{(l)}$;
- 12: Fuse the relation channels: $\hat{\mathbf{h}}_i^{(l+1)} = \sum_{r \in \mathcal{R}_i} \beta_{i,r}^{(l)} \mathbf{h}_{i,r}^{(l+1)}$;
- 13: Update the node representation: $\mathbf{h}_i^{(l+1)} = \text{LN}(\hat{\mathbf{h}}_i^{(l+1)} + \sigma(\hat{\mathbf{h}}_i^{(l+1)}))$;
- 14: **end for**
- 15: **end for**
- 16: Set $\mathbf{e}_u = \mathbf{h}_u^{(L)}$ for each user $u \in \mathcal{U}$;
- 17: Set $\mathbf{z}_i^S = \mathbf{h}_i^{(L)}$ for each item $i \in \mathcal{I}$;
- 18: **return** $\{\mathbf{e}_u\}_{u \in \mathcal{U}}$ and $\{\mathbf{z}_i^S\}_{i \in \mathcal{I}}$.

The following formulation details the operations summarized in Algorithm 1. The first stage performs relation-specific message transformation and neighbor-level attention within each semantic relation.

4.2.1. Relation-Specific Message Transformation and Neighbor Attention

Let $\mathbf{h}_i^{(l)} \in \mathbb{R}^d$ denote the representation of node i at the l -th graph layer, where $l = 0, \dots, L - 1$, and let $\mathcal{N}_i^r$ denote the set of neighboring nodes connected to node i through relation type r .

The initial representation $\mathbf{h}_i^{(0)}$ is obtained from the trainable graph-node embeddings introduced in Section 4.1.

For each relation r , the message passed from neighbor $j \in \mathcal{N}_i^r$ to node i is computed as

$$\mathbf{m}_{j \rightarrow i}^{(r,l)} = \mathbf{W}_r^{(l)} \mathbf{h}_j^{(l)}, \quad (19)$$

where $\mathbf{W}_r^{(l)} \in \mathbb{R}^{d \times d}$ is the trainable transformation matrix associated with relation r at layer l , and $\mathbf{m}_{j \rightarrow i}^{(r,l)} \in \mathbb{R}^d$ is the transformed message.

A separate transformation matrix is maintained for each relation type and graph layer. Each $\mathbf{W}_r^{(l)}$ is included in the trainable parameter set Θ , and its gradient is obtained by back-propagating the overall objective through the subsequent relation-aware attention, relation-level fusion, and node-update operations. The matrix is then updated jointly with the remaining model parameters during optimization.

After relation-specific message transformation, neighbor-level attention estimates the contribution of each neighbor within relation channel r .

The relation-aware attention score is calculated as

$$e_{ij}^{(r,l)} = \text{LeakyReLU} \left((\mathbf{a}_r^{(l)})^T \left[\mathbf{W}_r^{(l)} \mathbf{h}_i^{(l)} \parallel \mathbf{W}_r^{(l)} \mathbf{h}_j^{(l)} \right] \right), \quad (20)$$

where $\mathbf{a}_r^{(l)} \in \mathbb{R}^{2d}$ is the trainable attention vector associated with relation r at layer l , $\parallel$ denotes vector concatenation, and $e_{ij}^{(r,l)}$ measures the importance of neighbor j to node i under relation r .

The normalized attention coefficient is obtained by applying softmax over the neighbors associated with relation r :

$$\alpha_{ij}^{(r,l)} = \frac{\exp(e_{ij}^{(r,l)})}{\sum_{k \in \mathcal{N}_i^r} \exp(e_{ik}^{(r,l)})}. \quad (21)$$

The normalization is performed over all $k \in \mathcal{N}_i^r$, and therefore $\sum_{j \in \mathcal{N}_i^r} \alpha_{ij}^{(r,l)} = 1$. The coefficient $\alpha_{ij}^{(r,l)}$ represents the relative contribution of neighbor j within relation channel r .

The relation-specific neighborhood representation of node i is then computed as

$$\mathbf{h}_{i,r}^{(l+1)} = \sum_{j \in \mathcal{N}_i^r} \alpha_{ij}^{(r,l)} \mathbf{m}_{j \rightarrow i}^{(r,l)}. \quad (22)$$

where $\mathbf{h}_{i,r}^{(l+1)} \in \mathbb{R}^d$ represents the information aggregated from relation channel r .

The resulting relation-specific representations provide the inputs to the second stage, which evaluates the contributions of different semantic relation channels and updates the target-node representation.

4.2.2. Relation-Level Fusion and Node Updating

Since each node may receive messages from multiple relation types, HGCRRec introduces a relation-level fusion mechanism to combine the relation-specific representations.

Let $\mathcal{R}_i = \{r \in \mathcal{R} \mid \mathcal{N}_i^r \neq \emptyset\}$ denote the set of relation types incident to node i .

For each relation $r \in \mathcal{R}_i$, the relation-level importance score is defined as

$$q_{i,r}^{(l)} = \left(\mathbf{w}_\beta^{(l)}\right)^T \tanh\left(\mathbf{W}_\beta^{(l)} \left[\mathbf{h}_i^{(l)} \parallel \mathbf{h}_{i,r}^{(l+1)}\right] + \mathbf{b}_\beta^{(l)}\right), \quad (23)$$

where $\mathbf{W}_\beta^{(l)} \in \mathbb{R}^{d_\beta \times 2d}$, $\mathbf{w}_\beta^{(l)} \in \mathbb{R}^{d_\beta}$, and $\mathbf{b}_\beta^{(l)} \in \mathbb{R}^{d_\beta}$ are trainable relation-fusion parameters, and $q_{i,r}^{(l)} \in \mathbb{R}$ is the unnormalized importance of relation r for node i .

The relation-level fusion parameters $\mathbf{W}_\beta^{(l)}$, $\mathbf{w}_\beta^{(l)}$, and $\mathbf{b}_\beta^{(l)}$ are shared across node types and relation channels within the same graph layer, allowing the importance of different relations to be evaluated in a common latent space. Separate parameter sets are maintained for different graph layers.

The normalized relation weight is computed as

$$\beta_{i,r}^{(l)} = \frac{\exp\left(q_{i,r}^{(l)}\right)}{\sum_{r' \in \mathcal{R}_i} \exp\left(q_{i,r'}^{(l)}\right)}. \quad (24)$$

The relation weights satisfy $\sum_{r \in \mathcal{R}_i} \beta_{i,r}^{(l)} = 1$. The coefficient $\alpha_{ij}^{(r,l)}$ characterizes neighbor importance within relation r , while $\beta_{i,r}^{(l)}$ characterizes the contribution of relation channel r to node i .

The relation-specific representations are subsequently integrated according to $\beta_{i,r}^{(l)}$, followed by residual updating and layer normalization:

$$\mathbf{h}_i^{(l+1)} = \text{LN}\left(\mathbf{h}_i^{(l)} + \sigma\left(\sum_{r \in \mathcal{R}_i} \beta_{i,r}^{(l)} \mathbf{h}_{i,r}^{(l+1)}\right)\right), \quad (25)$$

where $\sigma(\cdot)$ denotes a nonlinear activation function and $\text{LN}(\cdot)$ denotes layer normalization. The residual connection and normalization operation are used to stabilize training and mitigate over-smoothing during multi-layer propagation.

Each graph layer therefore first models neighbor contributions within individual relation channels and then integrates the resulting relation-specific representations at the relation level. After L graph propagation layers, the final structural representation of item i is defined as $\mathbf{z}_i^S = \mathbf{h}_i^{(L)}$, and the final user representation is defined as $\mathbf{e}_u = \mathbf{h}_u^{(L)}$.

Through this two-level attention mechanism, HGCRRec captures both neighbor-level relevance and relation-level semantic importance. For interaction-sparse items, category, brand, semantic-entity, also-buy, and also-viewed relations provide structural information that complements the limited direct collaborative evidence and supports the subsequent cross-modal representation alignment.

4.3. Cross-Modal Graph Contrastive Learning

Although the relation-aware encoder can exploit heterogeneous structural information, cold-start items still suffer from weak collaborative supervision. To enhance item representation consistency, HGCRRec introduces cross-modal graph contrastive learning. Different from general graph augmentation-based contrastive learning, HGCRRec constructs item views from heterogeneous structure, visual content, and textual content. The structural view is obtained from the relation-aware graph encoder, while the visual and textual views are obtained from modality-specific projection networks.

For item i , the three view representations are defined as

$$\mathbf{z}_i^S = \mathbf{W}_S \mathbf{h}_i^{(L)}, \mathbf{z}_i^V = \mathbf{W}_V \tilde{\mathbf{x}}_i^V, \mathbf{z}_i^T = \mathbf{W}_T \tilde{\mathbf{x}}_i^T, \quad (26)$$

where $\mathbf{W}_S$, $\mathbf{W}_V$, and $\mathbf{W}_T$ are view-specific projection matrices.

The three vectors are normalized before contrastive learning. As defined in Equation (12), the same item under two different views is treated as a positive pair, while other items in the mini-batch are treated as negative samples. HGCRec adopts the contrastive objective in Equation (14) to align visual–textual, structural–visual, and structural–textual item representations.

To strengthen cross-modal representation learning for interaction-sparse items, HGCRec assigns degree-dependent contrastive weights to different items. The interaction-sparsity-aware weight of item i is defined as

$$\omega_i = 1 + \mu \exp\left(-\frac{d_i^{tr}}{\kappa}\right), \quad (27)$$

where d_i^{tr} is the training interaction degree of item i , μ controls the strength of sparsity-aware weighting, and κ controls the decay rate with respect to item degree. Accordingly, items with fewer training interactions receive stronger cross-modal consistency supervision, while the weight gradually approaches 1 as the interaction degree increases. The contribution of this weighting strategy is evaluated through an independent ablation variant, HGCRec w/o CS-weight, in which ω_i is uniformly set to 1.

When d_i^{tr} is small, ω_i becomes larger, which encourages the model to impose stronger cross-view consistency on cold-start items. Based on Equation (12), the weighted contrastive loss between two views a and b is written as

$$\mathcal{L}_{a,b}^{cs} = \frac{\sum_{i \in \mathcal{B}} \omega_i \ell_i^{a,b}}{\sum_{i \in \mathcal{B}} \omega_i}, \quad (28)$$

where $\ell_i^{a,b}$ is the item-level contrastive loss term defined by Equation (12).

The final cold-start-aware contrastive loss is

$$\mathcal{L}_{CL}^{cs} = \mathcal{L}_{V,T}^{cs} + \mathcal{L}_{S,V}^{cs} + \mathcal{L}_{S,T}^{cs}. \quad (29)$$

It encourages the structural, visual, and textual representations of the same item to be consistent, which can provide additional semantic constraints for sparse items. Similar ideas have been adopted in graph self-supervised learning and multi-modal graph contrastive recommendation [31,35], but HGCRec further combines this objective with cold-start-aware weighting and heterogeneous relation modeling.

4.4. Cold-Start Adaptive Fusion and Model Optimization

The cold-start adaptive fusion module is illustrated in Figure 2. HGCRec integrates collaborative, heterogeneous structural, visual, and textual representations through item-specific predictive information allocation. The interaction degree provides an explicit sparsity descriptor, while the learned representations characterize the information states derived from the corresponding collaborative, structural, and modality-specific sources. Their joint use enables the fusion module to learn item-specific representation combinations across different interaction levels.

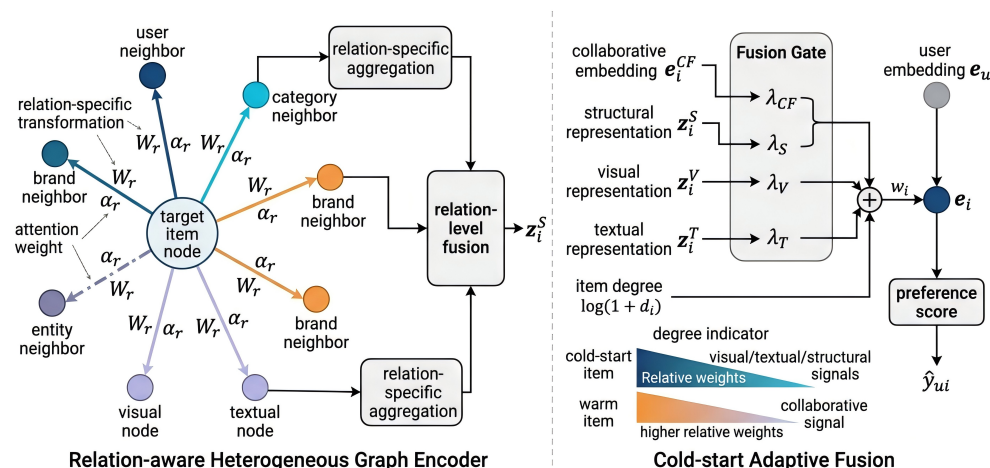


Figure 2. Relation-aware encoding and cold-start adaptive fusion.

Let $\mathbf{e}_i^{CF}$ denote the collaborative item embedding learned from user–item interactions, and let $\mathbf{z}_i^S$, $\mathbf{z}_i^V$, and $\mathbf{z}_i^T$ denote the structural, visual, and textual representations defined above. The fusion input of item i is constructed as

$$\mathbf{s}_i = \left[\log(1 + d_i^{tr}) \parallel \mathbf{e}_i^{CF} \parallel \mathbf{z}_i^S \parallel \mathbf{z}_i^V \parallel \mathbf{z}_i^T \right]. \quad (30)$$

Here, $\log(1 + d_i^{tr})$ quantifies the amount of direct interaction evidence available for item i in the training graph. The structural representation $\mathbf{z}_i^S$ summarizes the heterogeneous context obtained through relation-aware propagation, while $\mathbf{e}_i^{CF}$, $\mathbf{z}_i^V$, and $\mathbf{z}_i^T$ characterize the collaborative, visual, and textual information states, respectively.

The item-specific fusion coefficients are calculated as

$$\lambda_i = \text{softmax} \left(\mathbf{W}_g^2 \phi \left(\mathbf{W}_g^1 \mathbf{s}_i + \mathbf{b}_g^1 \right) + \mathbf{b}_g^2 \right), \quad (31)$$

where $\phi(\cdot)$ denotes the nonlinear activation function used in the fusion gate. The matrices $\mathbf{W}_g^1$ and $\mathbf{W}_g^2$ and the bias vectors $\mathbf{b}_g^1$ and $\mathbf{b}_g^2$ are jointly learned with the remaining model parameters.

The fusion-weight vector is

$$\lambda_i = \left[\lambda_i^{CF}, \lambda_i^S, \lambda_i^V, \lambda_i^T \right], \quad (32)$$

where $\lambda_i^m \geq 0$ and $\sum_{m \in \{CF, S, V, T\}} \lambda_i^m = 1$. The Softmax operation places the four information sources in a common normalized allocation space.

The final item representation is obtained by weighted fusion:

$$\mathbf{e}_i = \lambda_i^{CF} \mathbf{e}_i^{CF} + \lambda_i^S \mathbf{z}_i^S + \lambda_i^V \mathbf{z}_i^V + \lambda_i^T \mathbf{z}_i^T. \quad (33)$$

The fusion module and the four representation branches are optimized jointly. For compact notation, define $\mathcal{M} = \{CF, S, V, T\}$, $\mathbf{h}_i^{CF} = \mathbf{e}_i^{CF}$, $\mathbf{h}_i^S = \mathbf{z}_i^S$, $\mathbf{h}_i^V = \mathbf{z}_i^V$, and $\mathbf{h}_i^T = \mathbf{z}_i^T$. Let $\mathbf{H}_i = [\mathbf{h}_i^{CF}, \mathbf{h}_i^S, \mathbf{h}_i^V, \mathbf{h}_i^T]$ and $\delta_i = \partial \mathcal{L} / \partial \mathbf{e}_i$. The gradient received by information source m can then be expressed as

$$\frac{\partial \mathcal{L}}{\partial \mathbf{h}_i^m} = \lambda_i^m \delta_i + \left(\frac{\partial \lambda_i}{\partial \mathbf{h}_i^m} \right)^T \mathbf{H}_i^T \delta_i, \quad m \in \mathcal{M}. \quad (34)$$

The first term describes the gradient transmitted directly through the weighted representation, and the second term describes the gradient transmitted through the dependence of the fusion coefficients on the input representations. Together, the two terms coordinate representation learning and item-specific information allocation during joint optimization.

Let $\mathbf{g}_i$ denote the pre-Softmax gate logits in Equation (31). The gradient with respect to the logit corresponding to information source m is

$$\frac{\partial \mathcal{L}}{\partial g_i^m} = \lambda_i^m (\mathbf{h}_i^m - \mathbf{e}_i)^T \delta_i. \quad (35)$$

Equation (35) characterizes the local gradient guidance of the fusion gate at the current parameter state. The update of each gate logit depends on its current allocation coefficient, the difference between the corresponding source representation and the fused representation, and the gradient direction of the joint objective. This mechanism links the learned multi-source representations with item-specific predictive information allocation through end-to-end optimization.

The degree descriptor further conditions the allocation process on item interaction sparsity. As the interaction degree and learned representation states vary across items, the gate generates corresponding collaborative, structural, visual, and textual contribution coefficients for each item.

Given the final user representation $\mathbf{e}_u$ and item representation $\mathbf{e}_i$, the preference score is computed by the inner product:

$$\hat{y}_{ui} = \mathbf{e}_u^T \mathbf{e}_i. \quad (36)$$

For implicit feedback recommendation, HGCRc is optimized with a pairwise ranking objective. For each triplet (u, i, j) , where i is a positive item interacted with by user u and j is a sampled negative item, the recommendation loss is defined as

$$\mathcal{L}_{rec} = - \sum_{(u,i,j) \in \mathcal{O}} \log \sigma(\hat{y}_{ui} - \hat{y}_{uj}), \quad (37)$$

where $\mathcal{O}$ denotes the training triplet set and $\sigma(\cdot)$ is the sigmoid function.

The overall optimization objective combines the pairwise recommendation loss, the cross-modal contrastive objective defined in Equation (14), and L_2 regularization:

$$\mathcal{L} = \mathcal{L}_{rec} + \eta \mathcal{L}_{CL} + \lambda_{reg} \|\Theta\|_2^2, \quad (38)$$

where η controls the overall contribution of the cross-modal contrastive objective, λ_{reg} is the regularization coefficient, and Θ denotes all trainable parameters. The visual-textual, structural-visual, and structural-textual alignment terms are further weighted by $\lambda_{V,T}$, $\lambda_{S,V}$, and $\lambda_{S,T}$ in Equation (14).

For each mini-batch, $\mathcal{L}_{rec}$ is calculated from the sampled user-positive-item-negative-item triplets, and $\mathcal{L}_{CL}$ is calculated from the structural, visual, and textual item representations. The overall objective is optimized through back-propagation using AdamW. Gradients propagate through relation-aware encoding, cross-modal representation learning, and adaptive fusion, jointly updating the trainable parameters. The temperature τ , the overall contrastive coefficient η , and the three view-pair coefficients are selected according to validation NDCG@20, and the checkpoint with the highest validation NDCG@20 is used for test evaluation.

During inference, HGCRc ranks candidate items for each user according to Equation (36), and the items with the highest scores are returned as the recommendation list.

5. Experiments

5.1. Experimental Setup

5.1.1. Datasets and Preprocessing

To evaluate the effectiveness of the proposed HGCRec framework, experiments are conducted on real-world multi-modal recommendation datasets. Following previous multi-modal recommendation studies, we use Amazon product-recommendation datasets containing user–item interactions, item metadata, visual information, and textual information. The raw review and metadata files are obtained from the public Amazon product data released by McAuley et al. [36], which provides product reviews, product metadata, image-related information, category information, brand information, and also-viewed or also-bought links (<https://cseweb.ucsd.edu/~jmcauley/datasets/amazon/links.html> (accessed on 11 August 2026)). The selected datasets include Amazon Baby, Amazon Sports, and Amazon Clothing, corresponding to the Baby; Sports and Outdoors; and Clothing, Shoes and Jewelry categories in the Amazon product data (<https://snap.stanford.edu/data/amazon/productGraph/> (accessed on 11 August 2026)). These datasets represent small-scale, medium-scale, and relatively large-scale recommendation scenarios, respectively, and therefore provide a comprehensive benchmark for evaluating cold-start multi-modal recommendation.

The information used by HGCRec can be divided into structural, visual, and textual sources with different physical and digital properties. Structural information consists of digitally recorded behavioral and relational data, including user–item interactions and item associations with categories, brands, semantic entities, also-buy links, and also-viewed links. Visual information originates from item images and digitally represents observable physical properties, such as appearance, color, shape, and style. Textual information is derived from item titles and descriptions and represents functional, contextual, and semantic properties. In the recommendation process, structural information supports preference propagation and high-order relation modeling, visual information characterizes perceptual similarity, and textual information provides functional and semantic evidence for items with insufficient collaborative interactions.

For each dataset, user–item interactions are treated as implicit feedback. Repeated interactions between the same user and item are deduplicated, invalid identifiers and relations with missing endpoints are removed, and users and items with fewer than five interactions are filtered. The minimum-interaction threshold is fixed at 5 before data splitting to remove extremely low-frequency users and items that cannot provide sufficiently stable interaction evidence, and the same filtering rule is applied to all datasets and compared methods. All statistics reported below are calculated after preprocessing.

Visual features are extracted using the pre-trained checkpoint ViT-B/16. Each item image is resized to 224×224 , processed using resizing to 256×256 , center cropping to 224×224 , and standard ImageNet channel normalization and represented using the CLS-token. Textual inputs are constructed from the title and description, tokenized using the bert-base-uncased tokenizer, truncated or padded to a maximum length of 512, and encoded using the pre-trained checkpoint bert-base-uncased. The visual and textual encoders are frozen during recommendation-model training. Their output dimensions are $d_V = 768$ and $d_T = 768$, respectively.

Before modality projection and fusion, visual and textual features are checked for missing values, non-finite values, and all-zero vectors. Valid feature vectors are L_2 -normalized according to Equation (17), while unavailable modalities are identified using the availability indicator in Equation (16). The normalized features are then mapped into the shared d -dimensional latent space through the modality-specific projection networks in Equation (18). Cross-modal consistency is measured only after visual and textual repre-

sentations have been projected into the shared space, because cosine similarities between unaligned raw encoder outputs are not directly interpretable.

The reliability and information quality of the three sources are evaluated using descriptive statistics. For structural information, we report the mean, median, and standard deviation of user and item degrees, graph density, and the coverage of each semantic relation. For visual and textual information, we report modality coverage, missing-feature ratio, all-zero-feature ratio, and the mean and standard deviation of feature norms. These statistics are used to verify that the features entering the heterogeneous graph are sufficiently available, numerically valid, and comparable after standardization.

The implementation-level heterogeneous graph is constructed using the node and relation definitions introduced in Section 3. Category, brand, semantic-entity, also-buy, and also-viewed relations are obtained from item metadata independently of validation and test user–item interactions. Reverse edges are added for directed bipartite relations when required by message propagation. The exact numbers of nodes and directed edges for each relation type are reported together with the processed dataset statistics.

The detailed statistics of the datasets are shown in Table 1.

Table 1. Statistics of the processed Amazon recommendation datasets. Sparsity is calculated as $1 - |\mathcal{Y}|/(|\mathcal{U}||\mathcal{I}|)$.

Dataset	Users	Items	Interactions	Sparsity	Visual Dim.	Textual Dim.	Entity Nodes	Relation Types
Baby	19,445	7050	160,892	0.998829	768	768	28,632	6
Sports	35,598	18,357	296,741	0.999547	768	768	62,105	6
Clothing	78,264	42,713	612,487	0.999817	768	768	136,941	6

Table 2 reports the coverage, joint availability, all-zero ratios, and numerical quality of visual and textual features. All percentages are calculated over the processed item set, while feature norms are computed before L_2 normalization, which is applied before projection and contrastive alignment.

Table 2. Coverage and quality statistics of visual and textual information.

Dataset	Visual Coverage	Text Coverage	Both Available	All-Zero Visual	All-Zero Text	Raw Feature Norms Before L_2 Normalization (V/T)
Baby	96.5%	98.2%	95.1%	0.2%	0.1%	$14.2 \pm 3.1/9.8 \pm 2.4$
Sports	94.8%	97.5%	93.0%	0.3%	0.2%	$13.9 \pm 3.5/9.5 \pm 2.6$
Clothing	97.1%	99.1%	96.5%	0.1%	0.1%	$14.5 \pm 3.0/9.9 \pm 2.2$

To further characterize structural richness and heterogeneity, Table 3 reports the distribution of interaction degrees and the coverage of the principal metadata relations. The median and standard deviation are reported together with the mean because the interaction distributions are highly skewed and cannot be adequately described using the mean alone.

Table 3. Structural-information quality and relation coverage after preprocessing.

Dataset	User Degree	Item Degree	Graph Density	Category Cov.	Brand Cov.	Also-Buy Cov.	Also-View Cov.
Baby	8.2/5.0/6.1	22.8/11.0/35.4	0.00117	99.5%	65.4%	75.2%	82.1%
Sports	8.3/5.0/7.2	16.1/9.0/25.3	0.00045	99.2%	58.7%	70.1%	78.4%
Clothing	7.8/5.0/5.8	14.3/8.0/22.1	0.00018	99.8%	72.3%	68.5%	80.2%

Overall, the statistics in Tables 1–3 confirm that the three datasets exhibit high interaction sparsity and heterogeneous information distributions. Meanwhile, the high visual

and textual coverage, low all-zero-feature ratios, and diverse semantic relations demonstrate that the datasets provide rich and reliable complementary information for cold-start recommendation.

5.1.2. Implementation

The proposed HGCRec is compared with representative recommendation baselines, including graph-based and multi-modal recommendation methods. The baseline methods include LightGCN [1], VBPR [6], MMGCN [8], LATTICE [11], BM3 [13], and FREEDOM [14]. LightGCN is a strong graph collaborative filtering baseline based on user–item interaction propagation. VBPR introduces visual features into personalized recommendation. MMGCN and LATTICE further exploit multi-modal graph structures for item representation learning. BM3 and FREEDOM introduce self-supervised learning or graph denoising mechanisms and are regarded as strong multi-modal recommendation baselines. All compared methods use the same data split and evaluation protocol.

The embedding dimension is selected from $\{64, 128, 256, 512\}$, the learning rate is selected from $\{1 \times 10^{-4}, 5 \times 10^{-4}, 1 \times 10^{-3}, 5 \times 10^{-3}\}$, the number of graph layers is selected from $\{1, 2, 3, 4\}$, and the temperature parameter is selected from $\{0.1, 0.3, 0.5, 0.7, 1.0\}$. For HGCRec, the weights of different contrastive losses are selected according to validation performance. The model parameters are optimized using AdamW, and early stopping is applied based on validation NDCG@20. The best hyperparameters are set as follows: the embedding dimension is 256, the number of graph layers is 3, the learning rate is 1×10^{-3} , the temperature is 0.5, and the contrastive loss weights are 0.2, 0.3, and 0.4 for the three cross-modal objectives.

All experiments are conducted on a workstation equipped with an NVIDIA GeForce RTX 3090 GPU, an Intel Core i9-10900K CPU, and 128 GB of system memory. The software environment consists of Ubuntu 20.04, Python 3.9, PyTorch 2.0.1, and CUDA 11.8. The training batch size is set to 2048 for HGCRec and all compared methods whenever permitted by their original implementations. AdamW is used to optimize HGCRec, and early stopping is performed according to validation NDCG@20. The checkpoint achieving the highest validation NDCG@20 is retained for final test evaluation.

5.1.3. Evaluation Protocol

After preprocessing, the interactions of each user are divided using a user-level hold-out protocol, with 80%, 10%, and 10% assigned to the training, validation, and test sets, respectively. The split is performed independently for each user using a fixed split seed so that users appearing in the validation and test sets also have observable training interactions. The validation set is used only for hyperparameter selection and early stopping, whereas the test set is reserved exclusively for final performance evaluation.

The heterogeneous training graph $\mathcal{G}^{tr}$ contains only user–item interaction edges from the training set. Validation and test user–item edges are never included in graph propagation. Category, brand, semantic-entity, also-buy, also-viewed, visual, and textual relations are constructed from item metadata and modality content independently of the held-out validation and test interactions.

During negative sampling, validation and test positive items are excluded from the negative candidate set of the corresponding user. Specifically, for user u , negative items are sampled from $\mathcal{I} \setminus (\mathcal{I}_u^{tr} \cup \mathcal{I}_u^{val} \cup \mathcal{I}_u^{test})$. Therefore, held-out positive interactions are used neither as negative training samples nor as edges in the training graph. Item interaction degrees and sparsity groups are calculated exclusively from the training interaction set.

The evaluation follows the full-ranking protocol: for each user, all items that are not observed in the training set are treated as candidate items. The model ranks all candidate

items and returns the top- K recommendation list. The evaluation metrics include Recall@10, Recall@20, NDCG@10, NDCG@20, Precision@20, and MAP@20. The main performance table reports Recall@20, NDCG@20, Precision@20, and MAP@20. The main comparison experiments are repeated using five matched random seeds, namely 2024, 2025, 2026, 2027, and 2028, and their results are reported as mean $\pm$ standard deviation. The number of runs used for additional ablation, robustness, and efficiency experiments is stated separately in the corresponding subsection.

5.2. Baselines and Overall Performance

Table 4 reports the overall recommendation performance of HGCRc and the compared baselines. The relative improvement is calculated against the strongest baseline on each dataset.

Table 4. Overall performance comparison.

Dataset	Model	Recall@20	NDCG@20	Precision@20	MAP@20	Improvement (R@20/N@20)
Baby	LightGCN	0.1423 $\pm$ 0.0021	0.0786 $\pm$ 0.0015	0.0189 $\pm$ 0.0003	0.0421 $\pm$ 0.0009	–
Baby	VBPR	0.1571 $\pm$ 0.0024	0.0872 $\pm$ 0.0018	0.0205 $\pm$ 0.0004	0.0468 $\pm$ 0.0011	–
Baby	MMGCN	0.1645 $\pm$ 0.0026	0.0915 $\pm$ 0.0020	0.0217 $\pm$ 0.0004	0.0493 $\pm$ 0.0012	–
Baby	LATTICE	0.1682 $\pm$ 0.0028	0.0941 $\pm$ 0.0021	0.0224 $\pm$ 0.0005	0.0510 $\pm$ 0.0013	–
Baby	BM3	0.1736 $\pm$ 0.0030	0.0978 $\pm$ 0.0022	0.0231 $\pm$ 0.0005	0.0532 $\pm$ 0.0014	–
Baby	FREEDOM	0.1792 $\pm$ 0.0032	0.1014 $\pm$ 0.0024	0.0240 $\pm$ 0.0006	0.0557 $\pm$ 0.0015	–
Baby	HGCRc	0.1965 $\pm$ 0.0035	0.1127 $\pm$ 0.0026	0.0268 $\pm$ 0.0007	0.0624 $\pm$ 0.0017	9.65%/11.14%
Sports	LightGCN	0.1156 $\pm$ 0.0019	0.0634 $\pm$ 0.0013	0.0142 $\pm$ 0.0003	0.0335 $\pm$ 0.0008	–
Sports	VBPR	0.1284 $\pm$ 0.0022	0.0701 $\pm$ 0.0016	0.0158 $\pm$ 0.0003	0.0372 $\pm$ 0.0010	–
Sports	MMGCN	0.1347 $\pm$ 0.0024	0.0743 $\pm$ 0.0017	0.0166 $\pm$ 0.0004	0.0396 $\pm$ 0.0011	–
Sports	LATTICE	0.1389 $\pm$ 0.0025	0.0768 $\pm$ 0.0019	0.0171 $\pm$ 0.0004	0.0412 $\pm$ 0.0012	–
Sports	BM3	0.1442 $\pm$ 0.0027	0.0805 $\pm$ 0.0020	0.0178 $\pm$ 0.0004	0.0433 $\pm$ 0.0013	–
Sports	FREEDOM	0.1495 $\pm$ 0.0029	0.0837 $\pm$ 0.0021	0.0185 $\pm$ 0.0005	0.0454 $\pm$ 0.0014	–
Sports	HGCRc	0.1651 $\pm$ 0.0031	0.0932 $\pm$ 0.0023	0.0207 $\pm$ 0.0006	0.0511 $\pm$ 0.0016	10.44%/11.35%
Clothing	LightGCN	0.0927 $\pm$ 0.0017	0.0491 $\pm$ 0.0011	0.0105 $\pm$ 0.0002	0.0248 $\pm$ 0.0007	–
Clothing	VBPR	0.1035 $\pm$ 0.0020	0.0552 $\pm$ 0.0014	0.0118 $\pm$ 0.0003	0.0279 $\pm$ 0.0009	–
Clothing	MMGCN	0.1086 $\pm$ 0.0021	0.0584 $\pm$ 0.0015	0.0124 $\pm$ 0.0003	0.0297 $\pm$ 0.0010	–
Clothing	LATTICE	0.1124 $\pm$ 0.0023	0.0607 $\pm$ 0.0016	0.0129 $\pm$ 0.0003	0.0311 $\pm$ 0.0011	–
Clothing	BM3	0.1173 $\pm$ 0.0024	0.0641 $\pm$ 0.0017	0.0135 $\pm$ 0.0004	0.0330 $\pm$ 0.0012	–
Clothing	FREEDOM	0.1218 $\pm$ 0.0026	0.0672 $\pm$ 0.0019	0.0141 $\pm$ 0.0004	0.0349 $\pm$ 0.0013	–
Clothing	HGCRc	0.1364 $\pm$ 0.0028	0.0760 $\pm$ 0.0021	0.0162 $\pm$ 0.0005	0.0403 $\pm$ 0.0015	12.00%/13.10%

LightGCN provides a representative graph collaborative filtering baseline based on user–item interactions. VBPR incorporates visual features into collaborative recommendation. MMGCN and LATTICE further model multi-modal graph structures and item-level semantic associations. BM3 and FREEDOM introduce self-supervised representation learning and graph denoising strategies, respectively, and achieve the strongest baseline performance across the three datasets.

HGCRc achieves the highest performance on all four evaluation metrics across Amazon Baby, Amazon Sports, and Amazon Clothing. Compared with FREEDOM, HGCRc improves Recall@20 by 9.65%, 10.44%, and 12.00%, respectively, and improves NDCG@20 by 11.14%, 11.35%, and 13.10%. The gains become larger from Baby to Clothing, indicating consistent effectiveness across datasets with different interaction and modality characteristics.

These gains are associated with three complementary components. The heterogeneous graph provides semantic propagation paths through category, brand, entity, also-buy, and also-viewed relations, while visual and textual feature views provide complementary

content evidence. The relation-aware encoder performs relation-specific transformation, neighbor aggregation, and relation-level fusion. The item-specific fusion gate further coordinates collaborative, structural, visual, and textual representations according to the available item information.

Table 5 reports the paired significance analysis between HGCRRec and FREEDOM over five matched random seeds. For Recall@20, the paired mean differences on Baby, Sports, and Clothing are 0.0173, 0.0156, and 0.0146, respectively. For NDCG@20, the corresponding mean differences are 0.0113, 0.0095, and 0.0088. All six comparisons achieve $p < 0.01$, while the confidence intervals remain entirely above zero. The Cohen's d_z values further indicate stable performance differences across the matched runs.

Table 5. Paired significance tests between HGCRRec and FREEDOM over five matched random seeds.

Dataset	Metric	Mean Difference	95% CI	p -Value	Cohen's d_z
Baby	Recall@20	0.0173	[0.0142, 0.0204]	0.000108	6.82
Baby	NDCG@20	0.0113	[0.0093, 0.0133]	0.000090	7.15
Sports	Recall@20	0.0156	[0.0127, 0.0185]	0.000120	6.64
Sports	NDCG@20	0.0095	[0.0078, 0.0112]	0.000097	7.01
Clothing	Recall@20	0.0146	[0.0118, 0.0174]	0.000141	6.37
Clothing	NDCG@20	0.0088	[0.0072, 0.0104]	0.000110	6.79

Targeted Ablation of the Interaction-Sparsity-Aware Weight

To quantify the contribution of the interaction-sparsity-aware contrastive weight in Equation (27), we construct a controlled variant termed HGCRRec w/o CS-weight. This variant sets $\omega_i = 1$ for every item and applies uniform item-level averaging to each cross-modal contrastive loss. The heterogeneous graph, relation-aware encoder, modality projection networks, fusion gate, recommendation objective, hyperparameters, data partitions, and matched random seeds follow the full HGCRRec setting. The relative performance drop is calculated using the full HGCRRec result as the denominator.

As shown in Table 6, replacing the interaction-sparsity-aware weights with uniform item weights produces Recall@20 drops of 6.31%, 7.93%, and 9.82% on Baby, Sports, and Clothing, respectively. The corresponding NDCG@20 drops are 6.48%, 8.26%, and 10.79%. Precision@20 and MAP@20 exhibit the same performance trend across the three datasets. The increasing performance differences from Baby to Clothing further indicate that degree-dependent contrastive weighting provides greater benefits when item representations rely more strongly on sparse collaborative signals and complementary modality information. These results support the contribution of the proposed weighting strategy to interaction-sparsity item representation learning and overall recommendation accuracy.

Table 6. Targeted ablation of the interaction-sparsity-aware contrastive weight.

Dataset	Variant	Recall@20	NDCG@20	Precision@20	MAP@20	Recall@20 Drop	NDCG@20 Drop
Baby	HGCRRec	0.1965 ± 0.0035	0.1127 ± 0.0026	0.0268 ± 0.0007	0.0624 ± 0.0017	–	–
Baby	HGCRRec w/o CS-weight	0.1841 ± 0.0038	0.1054 ± 0.0029	0.0251 ± 0.0008	0.0583 ± 0.0019	6.31%	6.48%
Sports	HGCRRec	0.1651 ± 0.0031	0.0932 ± 0.0023	0.0207 ± 0.0006	0.0511 ± 0.0016	–	–
Sports	HGCRRec w/o CS-weight	0.1520 ± 0.0034	0.0855 ± 0.0026	0.0190 ± 0.0007	0.0468 ± 0.0018	7.93%	8.26%
Clothing	HGCRRec	0.1364 ± 0.0028	0.0760 ± 0.0021	0.0162 ± 0.0005	0.0403 ± 0.0015	–	–
Clothing	HGCRRec w/o CS-weight	0.1230 ± 0.0031	0.0678 ± 0.0024	0.0146 ± 0.0006	0.0358 ± 0.0017	9.82%	10.79%

5.3. Cold-Start and Sparsity-Aware Analysis

To further evaluate the effectiveness of HGCRec under cold-start conditions, items are divided into non-overlapping groups according to their training interaction degrees. The groups are defined as 0–2, 3–4, 5–9, 10–19, 20–49, and ≥ 50 interactions.

Following the definition in Section 3, the 0–2 group represents an extremely sparse or few-shot setting and contains items with zero, one, or two observed training interactions. For items with $d_i^{tr} = 0$, the training graph contains zero direct user–item interaction edges incident to the item. Their structural representations can still receive information through category, brand, entity, and available item–item semantic relations constructed from training-time metadata.

Table 7 reports the sparsity-aware and few-shot recommendation performance of HGCRec and the strongest baseline. All sparsity groups are determined exclusively from training interaction degrees. The “Test Inter.” column records held-out positive interactions used for evaluation. These held-out interactions are reserved for test measurement, while graph construction and representation propagation use the training interaction graph and training-time metadata. For example, the 1472 test interactions in the 0–2 group are evaluation targets associated with items having no more than two observed training interactions.

Table 7. Sparsity-aware and few-shot recommendation performance on Amazon Baby.

Group	Item Count	Test Inter.	Baseline	Base R@20	HGCRec R@20	Gain	Base N@20	HGCRec N@20	Gain
0–2	2146	1472	FREEDOM	0.0613	0.0789	28.71%	0.0325	0.0436	34.15%
3–4	1583	1629	FREEDOM	0.1147	0.1362	18.74%	0.0641	0.0773	20.59%
5–9	1305	1964	FREEDOM	0.1528	0.1741	13.94%	0.0863	0.0997	15.53%
10–19	927	2418	FREEDOM	0.1836	0.2015	9.75%	0.1042	0.1165	11.80%
20–49	702	3156	FREEDOM	0.2104	0.2253	6.99%	0.1216	0.1320	8.55%
≥ 50	387	5453	FREEDOM	0.2417	0.2526	4.51%	0.1409	0.1492	5.89%

The results in Table 7 show that HGCRec obtains the largest relative improvements in the extremely sparse item group. In the 0–2 interaction group, HGCRec improves Recall@20 by 28.71% and NDCG@20 by 34.15% compared with FREEDOM. This group jointly reflects items with zero, one, and two observed training interactions and therefore characterizes the broader extremely sparse or few-shot setting. Heterogeneous semantic relations provide additional structural propagation paths under limited direct interaction evidence, while visual and textual representations supply complementary item-level content information.

In the 3–4 and 5–9 interaction groups, HGCRec also achieves consistent improvements, indicating that relation-aware structural encoding, cross-modal alignment, and item-specific fusion remain effective as limited collaborative evidence gradually becomes available.

As the interaction degree increases, the performance gap between HGCRec and the strongest baseline gradually decreases. Items in these groups contain richer direct collaborative evidence, allowing collaborative recommendation methods to learn increasingly informative item representations. HGCRec maintains its performance advantage by jointly coordinating collaborative, structural, visual, and textual information through item-specific fusion.

Figure 3 illustrates the performance variation across different sparsity groups. The results show a progressively larger relative advantage of HGCRec as the amount of direct training interaction evidence decreases.

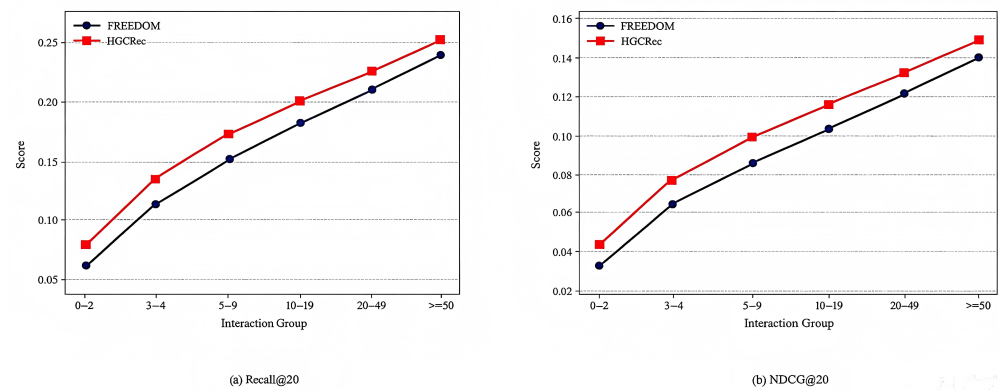


Figure 3. Recommendation performance across different item sparsity groups.

All structural propagation involved in this analysis is based on the training graph. For items with $d_i^{tr} = 0$, shared category, brand, entity, and item–item semantic relations provide training-graph paths connecting them with structurally related items. Validation and test user–item interactions are reserved for model selection and evaluation, respectively.

To further verify the contribution of the complete cross-modal contrastive objective, we compare HGCRec with a controlled variant, denoted as HGCRec w/o $\mathcal{L}_{CL}$, in which the overall contrastive coefficient η in Equation (38) is set to zero. The remaining model components and experimental settings are kept consistent. The corresponding validation and test results are reported in Table 8.

Table 8. Validation and test performance of the cross-modal contrastive objective.

Dataset	Variant	Validation Recall@20	Validation NDCG@20	Test Recall@20	Test NDCG@20
Baby	HGCRec	0.1985	0.1142	0.1965	0.1127
Baby	HGCRec w/o $\mathcal{L}_{CL}$	0.1890	0.1077	0.1871	0.1063
Sports	HGCRec	0.0765	0.0415	0.0751	0.0408
Sports	HGCRec w/o $\mathcal{L}_{CL}$	0.0728	0.0392	0.0715	0.0385
Clothing	HGCRec	0.0583	0.0274	0.0575	0.0269
Clothing	HGCRec w/o $\mathcal{L}_{CL}$	0.0558	0.0261	0.0549	0.0256

As shown in Table 8, HGCRec achieves consistent performance improvements over the controlled variant on both the validation and test sets. On the validation set, the relative improvements range from 4.48% to 5.08% for Recall@20 and from 4.98% to 6.04% for NDCG@20. On the test set, the corresponding improvements range from 4.74% to 5.03% and from 5.08% to 6.02%, respectively. These results support the effectiveness of cross-view representation alignment and demonstrate that the cross-modal contrastive objective contributes to the recommendation accuracy and generalization ability of HGCRec.

5.4. Parameter Sensitivity and Efficiency Analysis

This section analyzes the influence of key hyperparameters and compares the computational efficiency of HGCRec with representative baselines. The analyzed hyperparameters include embedding dimension, number of graph layers, contrastive temperature, and contrastive loss weight. In addition, we analyze the selection mechanism and variance of the dynamically learned fusion weights and discuss their effects on model optimization and convergence. The online graph-update latency associated with dynamically inserting new items is further evaluated to distinguish batch recommendation scoring from online representation updating.

5.4.1. Hyperparameter Sensitivity

Figure 4 shows the parameter sensitivity results. When the embedding dimension increases from 64 to 256, the performance continuously improves. However, further increasing the dimension to 512 leads to a slight decrease, which may be caused by overfitting or increased optimization difficulty. For graph layers, HGCRRec achieves the best performance when the layer number is 3. Using too few layers may limit high-order semantic propagation, while too many layers may introduce noise or over-smoothing.

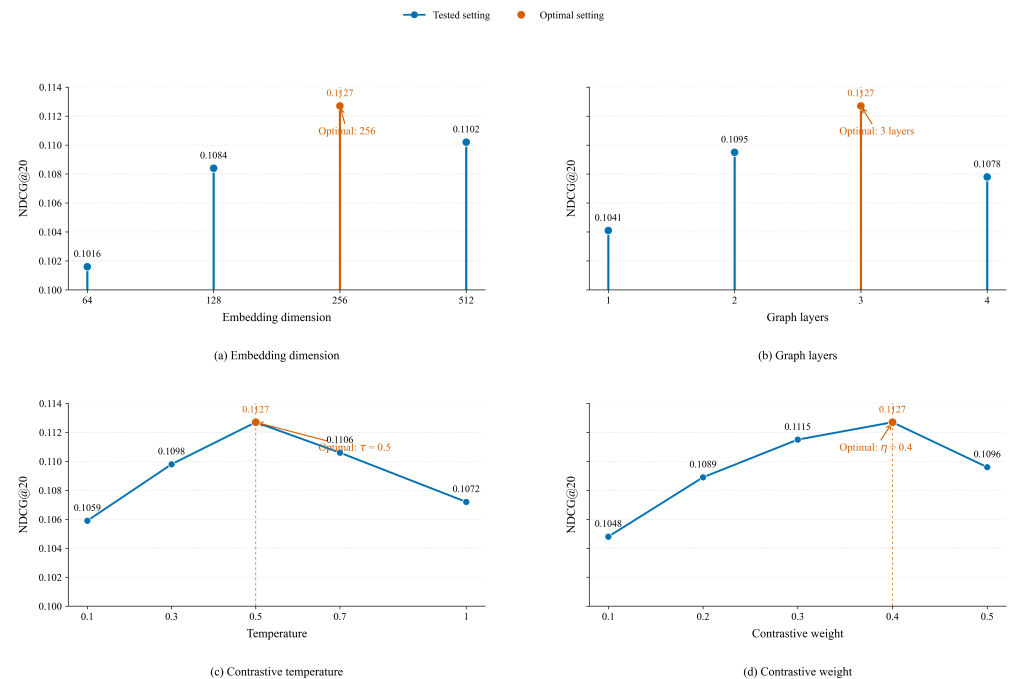


Figure 4. Parameter sensitivity analysis.

The contrastive temperature controls the sharpness of the similarity distribution in the contrastive objective. HGCRRec achieves the best result when the temperature is 0.5. A smaller temperature may make the model overemphasize hard negatives, while a larger temperature may weaken the discriminative effect of contrastive learning. The contrastive loss weight also influences performance. When the weight is too small, the cross-modal alignment effect becomes limited; when it is too large, the recommendation objective may be weakened. The best performance is obtained when the contrastive weight is 0.4.

5.4.2. Fusion-Weight Analysis

To directly examine whether the adaptive fusion module changes its information-source allocation according to item interaction sparsity, we conduct a fusion-weight analysis on Amazon Baby. The checkpoint achieving the highest validation NDCG@20 is fixed, and no additional parameter optimization is performed. We randomly sample 100 items from the 0–2 training-interaction group and 100 items from the ≥ 50 training-interaction group using a fixed random seed. A forward pass is then performed to record the four softmax-normalized fusion weights, namely λ_i^{CF} , λ_i^S , λ_i^V , and λ_i^T , for every sampled item.

The combined auxiliary-information weight is further defined as $\lambda_i^{Aux} = \lambda_i^S + \lambda_i^V + \lambda_i^T$. Table 9 reports the mean and standard deviation of the learned fusion weights in the two sparsity groups.

Table 9. Learned adaptive fusion weights for items with different training–interaction sparsity levels on Amazon Baby. Each group contains 100 randomly sampled items, and the values are reported as mean $\pm$ standard deviation.

Training–Interaction Group	λ^{CF}	λ^S	λ^V	λ^T	λ^{Aux}
0–2	0.182 $\pm$ 0.043	0.376 $\pm$ 0.051	0.265 $\pm$ 0.038	0.177 $\pm$ 0.034	0.818 $\pm$ 0.043
≥ 50	0.694 $\pm$ 0.067	0.142 $\pm$ 0.032	0.093 $\pm$ 0.026	0.071 $\pm$ 0.022	0.306 $\pm$ 0.067

As shown in Table 9, the mean collaborative weight of the 0–2 interaction group is 0.182, compared with 0.694 for the ≥ 50 interaction group. Meanwhile, the combined auxiliary–information weights of the two groups are 0.818 and 0.306, respectively. These results show that the adaptive gate assigns a lower proportion of collaborative information and a higher proportion of structural, visual, and textual information to extremely sparse and few-shot items. This observed redistribution provides direct evidence that the learned fusion weights vary with item interaction sparsity rather than remaining close to a fixed global weighting scheme.

Unlike the contrastive loss weight, the fusion weights λ_i^{CF} , λ_i^S , λ_i^V , and λ_i^T are not manually selected global hyperparameters. They are dynamically generated by the adaptive gate according to the interaction degree and multi-source representations of each item. The gate parameters are jointly learned by minimizing the overall objective, and the final checkpoint is selected according to validation NDCG@20. The softmax operation constrains the fusion weights to be non-negative and to satisfy $\lambda_i^{CF} + \lambda_i^S + \lambda_i^V + \lambda_i^T = 1$.

5.4.3. Computational Efficiency

Figure 5 compares the computational efficiency of HGCRec with representative baselines in terms of training time per epoch, total training time, batch inference time, GPU memory usage, and the number of trainable parameters. All efficiency experiments are conducted on a workstation equipped with an NVIDIA GeForce RTX 3090 GPU, an Intel Core i9-10900K CPU, and 128 GB of system memory. The reported inference time measures batch recommendation scoring after the user and item embeddings have been generated.

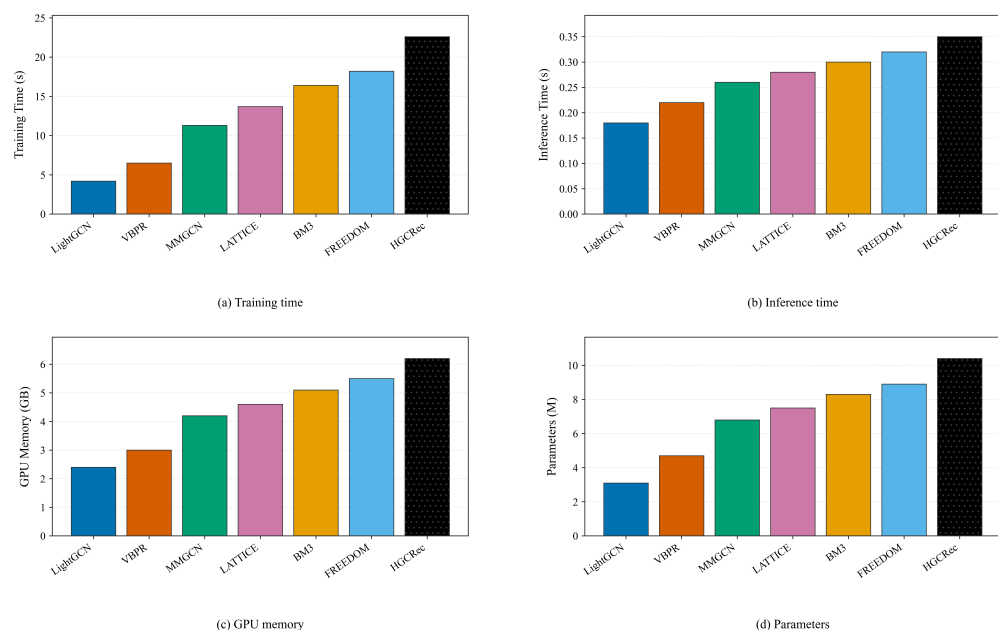


Figure 5. Efficiency comparison of HGCRec and representative baselines in terms of training time, inference time, GPU memory usage, and the number of trainable parameters.

As shown in Figure 5, heterogeneous relation encoding, modality projection, and cross-modal contrastive learning introduce additional computation compared with lightweight graph collaborative filtering methods such as LightGCN. Consequently, HGCRec requires more training time than lightweight baselines and slightly more training time than mainstream multi-modal baselines. Its GPU memory usage and parameter size also increase moderately because of the relation-specific transformation and modality projection parameters. For batch recommendation scoring with precomputed embeddings, HGCRec requires 0.35 s and remains in the same order of magnitude as the multi-modal baselines.

The batch-scoring latency characterizes recommendation generation with available embeddings, while the representation-update latency for newly arriving items involves graph construction and heterogeneous neighborhood propagation. We further conduct a new-item insertion experiment on Amazon Baby to quantify this update cost. A trained HGCRec checkpoint remains fixed throughout the experiment. We randomly select 100 held-out items and remove their item nodes, associated metadata edges, and visual and textual feature views from the initial graph. During insertion, the corresponding metadata relations and modality features are restored, while the held-out user–item interactions are reserved for test evaluation.

Two insertion modes are evaluated. Batch insertion adds all 100 items simultaneously and performs one L -layer heterogeneous graph propagation, whereas single-item online insertion processes the items sequentially and performs graph propagation after each arrival. The measured latency covers graph construction, neighborhood aggregation, and final item-representation generation, with data loading and model initialization completed before timing. Each mode is evaluated after 10 warm-up runs and repeated 50 times. The results are reported as mean $\pm$ standard deviation.

As shown in Table 10, batch insertion requires 25.0 ms for one graph update, corresponding to an amortized latency of 0.25 ms per item. Single-item insertion requires 18.5 ms per update, producing a 74-fold higher amortized cost per item. Sequentially processing the same 100 items requires approximately 1.85 s, while one batch update requires 25.0 ms. Batch insertion distributes the graph-propagation cost across multiple items, whereas sequential insertion repeatedly invokes the L -layer neighborhood aggregation procedure.

Table 10. Online graph-update latency of HGCRec for new-item insertion on Amazon Baby.

Insertion Mode	Items per Update	Latency per Update	Amortized Latency per Item
Batch insertion	100	25.0 $\pm$ 1.2 ms	0.250 $\pm$ 0.012 ms
Single-item online insertion	1	18.5 $\pm$ 0.9 ms	18.5 $\pm$ 0.9 ms

These results identify repeated heterogeneous neighborhood propagation as the primary online-update bottleneck under sequential item arrivals. The current latency profile favors periodic and mini-batch item updates in practical recommendation services. Neighbor-aggregation caching, propagation within affected L -hop neighborhoods, and incremental item-embedding updates provide direct optimization strategies for high-frequency online deployment.

5.5. Modality Noise Robustness

To evaluate the overall robustness of HGCRec under controlled random modality corruption, we conduct a noise-perturbation experiment on Amazon Baby. A single checkpoint selected according to validation NDCG@20 is fixed for all robustness evaluations, and the recommendation-model parameters remain unchanged. For visual perturbation, after resizing and center cropping, 20% of the non-overlapping 16×16 image patches are

randomly selected and replaced with the ImageNet channel mean before channel normalization. For textual perturbation, 20% of the valid input tokens are randomly replaced with the MASK token, excluding CLS, SEP, and padding tokens. The perturbed features are re-extracted using the same frozen ViT-B/16 and BERT-base encoders. The clean-feature result is evaluated once using the fixed checkpoint, whereas each noisy condition is repeated three times using independent random masks and reported as mean $\pm$ standard deviation.

As shown in Table 11, masking 20% of the image patches produces relative decreases of 2.54% and 2.84% in Recall@20 and NDCG@20, respectively. Text-token masking produces corresponding decreases of 1.88% and 2.13%. Under joint visual and textual perturbation, HGCRRec retains 94.76% of the clean-feature NDCG@20. The gradual performance changes indicate that HGCRRec maintains overall recommendation effectiveness under the controlled random corruption conditions considered here.

Table 11. Overall robustness of HGCRRec on Amazon Baby under controlled random visual-patch and textual-token masking.

Noise Condition	Image Mask	Text Mask	Recall@20	NDCG@20
Clean features	0%	0%	0.1965	0.1127
Image-patch masking	20%	0%	0.1915 $\pm$ 0.0012	0.1095 $\pm$ 0.0008
Text-token masking	0%	20%	0.1928 $\pm$ 0.0010	0.1103 $\pm$ 0.0009
Joint modality masking	20%	20%	0.1872 $\pm$ 0.0015	0.1068 $\pm$ 0.0011

The robustness observed at the aggregate level is supported by the joint use of structural, collaborative, visual, and textual information. Cross-modal alignment and item-specific fusion provide complementary representation channels when partial perturbations affect the modality features.

6. Conclusions

This paper proposes HGCRRec, a heterogeneous graph contrastive learning framework for cold-start multi-modal recommendation. HGCRRec constructs a heterogeneous structural graph containing users, items, categories, brands, and semantic entities, while visual and textual information are incorporated as item-side feature views. A relation-aware encoder captures heterogeneous structural semantics, and a cross-modal contrastive objective coordinates structural, visual, and textual item representations. An interaction-sparsity-aware weighting strategy further allocates stronger contrastive supervision to items with fewer training interactions. The item-specific adaptive fusion module integrates collaborative, structural, visual, and textual representations according to interaction sparsity and learned multi-source representation states.

Experiments on Amazon Baby, Amazon Sports, and Amazon Clothing show that HGCRRec consistently outperforms the selected graph-based and multi-modal baselines under the unified evaluation protocol. Compared with FREEDOM, HGCRRec improves Recall@20 by 9.65–12.00% and NDCG@20 by 11.14–13.10%. The sparsity-aware analysis on Amazon Baby further shows larger relative improvements for extremely sparse and few-shot items. In the 0–2 training-interaction group, which contains items with zero, one, or two observed training interactions, Recall@20 and NDCG@20 increase by 28.71% and 34.15%, respectively. The targeted ablation further shows that replacing the interaction-sparsity-aware weights with uniform item weights produces Recall@20 drops of 6.31–9.82% and NDCG@20 drops of 6.48–10.79% across the three datasets. Additional controlled evaluations support the contribution of cross-modal representation alignment, while the modality perturbation experiment shows that HGCRRec maintains overall recommendation effectiveness under the controlled random corruption conditions considered in this study.

HGCRec introduces additional training time, memory consumption, and parameter overhead through heterogeneous relation encoding, modality projection, and contrastive learning. Its batch recommendation scoring remains comparable in magnitude to main-stream multi-modal methods, while the current latency profile supports offline training and periodic or mini-batch item updates in practical recommendation services. Future work will investigate lightweight relation-aware propagation, incremental graph updating, more fine-grained evaluation of zero-direct-interaction and few-shot items, and robustness analysis under structured and non-IID modality degradation.

Author Contributions: Conceptualization, H.L.; methodology, H.L. and Y.X.; software, H.L. and Y.X.; validation, Y.X.; investigation, Y.X.; resources, H.L.; writing—original draft preparation, H.L. and Y.X.; writing—review and editing, H.L.; visualization, Y.X.; supervision, H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The raw data used in this study are openly available in the Amazon Product Data repository at <https://cseweb.ucsd.edu/~jmcauley/datasets/amazon/links.html> (accessed on 11 August 2026) and <https://snap.stanford.edu/data/amazon/productGraph/> (accessed on 11 August 2026). The processed datasets generated during the current study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. He, X.; Deng, K.; Wang, X.; Li, Y.; Zhang, Y.; Wang, M. 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, Virtual Event, China, 25–30 July 2020; pp. 639–648. [\[CrossRef\]](#)
2. Wang, X.; He, X.; Cao, Y.; Liu, M.; Chua, T.S. KGAT: Knowledge Graph Attention Network for Recommendation. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; pp. 950–958. [\[CrossRef\]](#)
3. Wang, X.; Huang, T.; Wang, D.; Yuan, Y.; Liu, Z.; He, X.; Chua, T.S. Learning Intents behind Interactions with Knowledge Graph for Recommendation. In Proceedings of the Web Conference 2021, Ljubljana, Slovenia, 19–23 April 2021; pp. 878–887. [\[CrossRef\]](#)
4. Tutsoy, O. Graph Theory Based Large-Scale Machine Learning With Multi-Dimensional Constrained Optimization Approaches for Exact Epidemiological Modeling of Pandemic Diseases. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 9836–9845. [\[CrossRef\]](#) [\[PubMed\]](#)
5. Bose, P.; Rana, P.; Ghosh, P. Attention-Based Multimodal Deep Learning on Vision-Language Data: Models, Datasets, Tasks, Evaluation Metrics and Applications. *IEEE Access* **2023**, *11*, 80624–80646. [\[CrossRef\]](#)
6. He, R.; McAuley, J. VBPR: Visual Bayesian Personalized Ranking from Implicit Feedback. In Proceedings of the AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016; Volume 30. [\[CrossRef\]](#)
7. Liu, F.; Cheng, Z.; Sun, C.; Wang, Y.; Nie, L.; Kankanhalli, M. User Diverse Preference Modeling by Multimodal Attentive Metric Learning. In Proceedings of the 27th ACM International Conference on Multimedia, Nice, France, 21–25 October 2019; pp. 1526–1534. [\[CrossRef\]](#)
8. Wei, Y.; Wang, X.; Nie, L.; He, X.; Hong, R.; Chua, T.S. MMGCN: Multi-Modal Graph Convolution Network for Personalized Recommendation of Micro-Video. In Proceedings of the 27th ACM International Conference on Multimedia, Nice, France, 21–25 October 2019; pp. 1437–1445. [\[CrossRef\]](#)
9. Wang, Q.; Wei, Y.; Yin, J.; Wu, J.; Song, X.; Nie, L. DualGNN: Dual Graph Neural Network for Multimedia Recommendation. *IEEE Trans. Multimed.* **2023**, *25*, 1074–1084. [\[CrossRef\]](#)
10. Wei, Y.; Wang, X.; Nie, L.; He, X.; Chua, T.S. Graph-Refined Convolutional Network for Multimedia Recommendation with Implicit Feedback. In Proceedings of the 28th ACM International Conference on Multimedia, Seattle, WA, USA, 12–16 October 2020; pp. 3541–3549. [\[CrossRef\]](#)
11. Zhang, J.; Zhu, Y.; Liu, Q.; Wu, S.; Wang, S.; Wang, L. Mining Latent Structures for Multimedia Recommendation. In Proceedings of the 29th ACM International Conference on Multimedia, Virtual Event, China, 20–24 October 2021; pp. 3872–3880. [\[CrossRef\]](#)
12. Zhang, J.; Zhu, Y.; Liu, Q.; Zhang, M.; Wu, S.; Wang, L. Latent Structure Mining with Contrastive Modality Fusion for Multimedia Recommendation. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 9154–9167. [\[CrossRef\]](#)

13. Zhou, X.; Zhou, H.; Liu, Y.; Zeng, Z.; Miao, C.; Wang, P.; You, Y.; Jiang, F. Bootstrap Latent Representations for Multi-Modal Recommendation. In Proceedings of the ACM Web Conference 2023, Austin, TX, USA, 30 April–4 May 2023; pp. 845–854. [\[CrossRef\]](#)
14. Zhou, X.; Zhou, H.; Liu, Y.; Zeng, Z.; Miao, C.; Wang, P.; You, Y.; Jiang, F. A Tale of Two Graphs: Freezing and Denoising Graph Structures for Multimodal Recommendation. In Proceedings of the 31st ACM International Conference on Multimedia, Ottawa, ON, Canada, 29 October–3 November 2023; pp. 935–943. [\[CrossRef\]](#)
15. Wei, W.; Huang, C.; Xia, L.; Zhang, C. Multi-Modal Self-Supervised Learning for Recommendation. In Proceedings of the ACM Web Conference 2023, Austin, TX, USA, 30 April–4 May 2023; pp. 790–800. [\[CrossRef\]](#)
16. Wu, J.; Wang, X.; Feng, F.; He, X.; Chen, L.; Lian, J.; Xie, X. Self-Supervised Graph Learning for Recommendation. In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, 11–15 July 2021; pp. 726–735. [\[CrossRef\]](#)
17. Lin, Z.; Tian, C.; Hou, Y.; Zhao, W.X. Improving Graph Collaborative Filtering with Neighborhood-Enriched Contrastive Learning. In Proceedings of the ACM Web Conference 2022, Virtual Event, Lyon, France, 25–29 April 2022; pp. 2320–2329. [\[CrossRef\]](#)
18. Wu, S.; Sun, F.; Zhang, W.; Xie, X.; Cui, B. Graph Neural Networks in Recommender Systems: A Survey. *ACM Comput. Surv.* **2022**, *55*, 1–37. [\[CrossRef\]](#)
19. Gao, C.; Zheng, Y.; Li, N.; Li, Y.; Qin, Y.; Piao, J.; Quan, Y.; Chang, J.; Jin, D.; He, X.; et al. A Survey of Graph Neural Networks for Recommender Systems: Challenges, Methods, and Directions. *ACM Trans. Recomm. Syst.* **2023**, *1*, 1–51. [\[CrossRef\]](#)
20. Guo, Q.; Zhuang, F.; Qin, C.; Zhu, H.; Xie, X.; Xiong, H.; He, Q. A Survey on Knowledge Graph-Based Recommender Systems. *IEEE Trans. Knowl. Data Eng.* **2022**, *34*, 3549–3568. [\[CrossRef\]](#)
21. Wang, H.; Zhang, F.; Wang, J.; Zhao, M.; Li, W.; Xie, X.; Guo, M. RippleNet: Propagating User Preferences on the Knowledge Graph for Recommender Systems. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management, Torino, Italy, 22–26 October 2018; pp. 417–426. [\[CrossRef\]](#)
22. Wang, H.; Zhao, M.; Xie, X.; Li, W.; Guo, M. Knowledge Graph Convolutional Networks for Recommender Systems. In Proceedings of the World Wide Web Conference, San Francisco, CA, USA, 13–17 May 2019; pp. 3307–3313. [\[CrossRef\]](#)
23. Wang, H.; Zhang, F.; Zhang, M.; Leskovec, J.; Zhao, M.; Li, W.; Wang, Z. Knowledge-Aware Graph Neural Networks with Label Smoothness Regularization for Recommender Systems. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; pp. 968–977. [\[CrossRef\]](#)
24. Wang, X.; Wang, D.; Xu, C.; He, X.; Cao, Y.; Chua, T.S. Explainable Reasoning over Knowledge Graphs for Recommendation. *Proc. AAAI Conf. Artif. Intell.* **2019**, *33*, 5329–5336. [\[CrossRef\]](#)
25. Tu, K.; Cui, P.; Wang, D.; Zhang, Z.; Zhou, J.; Qi, Y.; Zhu, W. Conditional Graph Attention Networks for Distilling and Refining Knowledge Graphs in Recommendation. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management, Virtual Event, Australia, 1–5 November 2021; pp. 1834–1843. [\[CrossRef\]](#)
26. Sun, R.; Cao, X.; Zhao, Y.; Wan, J.; Zhou, K.; Zhang, F.; Wang, Z.; Zheng, K. Multi-Modal Knowledge Graphs for Recommender Systems. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management, Virtual Event, Ireland, 19–23 October 2020; pp. 1405–1414. [\[CrossRef\]](#)
27. Wang, Y.; Tang, S.; Lei, Y.; Song, W.; Wang, S.; Zhang, M. DisenHAN: Disentangled Heterogeneous Graph Attention Network for Recommendation. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management, Virtual Event, Ireland, 19–23 October 2020; pp. 1605–1614. [\[CrossRef\]](#)
28. Liu, Q.; Wu, F.; Wang, X.; Chen, X.; Zhang, J.; Wang, L. Multimodal Recommender Systems: A Survey. *ACM Comput. Surv.* **2024**, *57*, 1–17. [\[CrossRef\]](#)
29. Zhang, S.; Yao, L.; Sun, A.; Tay, Y. Deep Learning Based Recommender System: A Survey and New Perspectives. *ACM Comput. Surv.* **2019**, *52*, 1–38. [\[CrossRef\]](#)
30. Liu, F.; Chen, H.; Cheng, Z.; Liu, A.; Nie, L.; Kankanhalli, M. Disentangled Multimodal Representation Learning for Recommendation. *IEEE Trans. Multimed.* **2023**, *25*, 7149–7159. [\[CrossRef\]](#)
31. Liu, K.; Xue, F.; Guo, D.; Sun, P.; Qian, S.; Hong, R. Multimodal Graph Contrastive Learning for Multimedia-Based Recommendation. *IEEE Trans. Multimed.* **2023**, *25*, 9343–9355. [\[CrossRef\]](#)
32. Yu, P.; Tan, Z.; Lu, G.; Bao, B.K. Multi-View Graph Convolutional Network for Multimedia Recommendation. In Proceedings of the 31st ACM International Conference on Multimedia, Ottawa, ON, Canada, 29 October–3 November 2023; pp. 6576–6585. [\[CrossRef\]](#)
33. Wei, Y.; Wang, X.; Li, Q.; Nie, L.; Li, Y.; Li, X.; Chua, T.S. Contrastive Learning for Cold-Start Recommendation. In Proceedings of the 29th ACM International Conference on Multimedia, Virtual Event, China, 20–24 October 2021; pp. 5382–5390. [\[CrossRef\]](#)
34. Zhou, Z.; Yang, N.; Wang, Y. Contrastive Collaborative Filtering for Cold-Start Item Recommendation. In Proceedings of the ACM Web Conference 2023, Austin, TX, USA, 30 April–4 May 2023; pp. 928–937. [\[CrossRef\]](#)

35. Liu, Y.; Jin, M.; Pan, S.; Zhou, C.; Zheng, Y.; Xia, F.; Yu, P.S. Graph Self-Supervised Learning: A Survey. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 5879–5900. [[CrossRef](#)]
36. McAuley, J.; Targett, C.; Shi, Q.; van den Hengel, A. Image-Based Recommendations on Styles and Substitutes. In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, 9–13 August 2015; pp. 43–52. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.