RESEARCH-ARTICLE

STARLINE: Contrastive Learning with Modality-Aware Graph Refinement for Effective Multimedia Recommendation

TAERI KIM, Hanyang University, Seoul, South Korea

SOHEE BAN, Hanyang University, Seoul, South Korea

HYUNJOON KIM, Hanyang University, Seoul, South Korea

SANG-WOOK KIM, Hanyang University, Seoul, South Korea

Open Access Support provided by:

Hanyang University

Published: 03 August 2025

Citation in BibTeX format

KDD '25: The 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining
August 3 - 7, 2025
Toronto ON, Canada

Conference Sponsors:
SIGMOD
SIGKDD

https://doi.org/10.1145/3711896.3737136
ISBN: 9798400714542

STARLINE: Contrastive Learning with Modality-Aware Graph Refinement for Effective Multimedia Recommendation

Taeri Kim
Hanyang University
Seoul, Korea
taerik@hanyang.ac.kr

Sohee Ban
Hanyang University
Seoul, Korea
soheeb@hanyang.ac.kr

Hyunjoon Kim
Hanyang University
Seoul, Korea
hyunjoonkim@hanyang.ac.kr

Sang-Wook Kim*
Hanyang University
Seoul, Korea
wook@hanyang.ac.kr

Abstract

Beyond using multimodal features of items in addition to user–item interactions, researchers have additionally utilized Contrastive Learning (CL) in recent multimedia recommender systems to highly alleviate the data sparsity problem. CL-based methods generate at least two embeddings (*i.e.*, views) for each instance and enrich the information of each instance from various perspectives via the views, thereby alleviating the data sparsity problem. Therefore, CL-based methods have focused on generating views that effectively represent the characteristics of each instance for their downstream tasks. Similarly, CL-based multimedia recommender systems have made efforts to effectively generate their user/item views by leveraging items' multimodal features. However, we point out the following two limitations that they have overlooked in generating their views: (1) they either have *not* attempted to identify the *influence of each modality feature* of an item on user–item interactions, or have identified it by *randomly* masking or dropping user–item interactions, and (2) they have *not* attempted to identify *non-interactions likely to result in interactions* in the future. To overcome these limitations, we propose a novel multimedia recommendation framework, named **STARLINE**, utilizing contra**ST**ive le**AR**ning with moda**LI**ty-aware graph refi**NE**ment. Extensive experiments on five real-world datasets validate the effectiveness and validity of STARLINE, especially showing consistently higher accuracy by up to 13.24% compared to the best competitor.

CCS Concepts

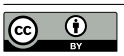
• **Information systems** → Recommender systems.

Keywords

multimedia recommendation, contrastive learning, graph refinement

ACM Reference Format:
Taeri Kim, Sohee Ban, Hyunjoon Kim, and Sang-Wook Kim. 2025. STARLINE: Contrastive Learning with Modality-Aware Graph Refinement for Effective Multimedia Recommendation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD '25), August 3–7, 2025, Toronto, ON, Canada.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3711896.3737136

*Corresponding author.



KDD '25, Toronto, ON, Canada

ACM ISBN 979-8-4007-1454-2/2025/08
https://doi.org/10.1145/3711896.3737136

KDD Availability Link:
The source code of this paper has been made publicly available at https://doi.org/10.5281/zenodo.15501421.

1 Introduction

The rapid increase of items on e-commerce platforms is making it difficult for users to find their preferred items. To deal with this information overload, numerous platforms have utilized recommender systems that predict the items a user is likely to prefer. Collaborative Filtering (CF), which is the most widely used recommender technique, predicts the user's preferences for items based on user–item interactions (*e.g.*, click logs) [2, 6, 15, 17, 21, 25]. However, the very sparse nature of interactions makes CF methods have difficulty in accurately predicting the user's preferences for items.

Multimedia recommender systems have emerged to alleviate this data sparsity problem. They utilize not only user–item interactions but also the multimodal features of items, which are pre-obtained by applying deep-learning models (*e.g.*, Convolutional Neural Network [16] and Recurrent Neural Network [8]) to the multimodal data of items (*e.g.*, item images and descriptions)[1] [5, 11–13, 35, 36]. From the multimodal features, multimedia recommender systems capture the *user's preferred feature in each modality* and utilize these features to predict items she is likely to prefer. As a result, they have significantly outperformed pure CF methods in terms of accuracy.

Recent multimedia recommender systems have attempted to further alleviate the data sparsity problem by additionally utilizing *Contrastive Learning* (CL) [3, 22, 26, 32, 36]. CL is a learning method that generates at least two embeddings (*i.e.*, *views*) for each instance (*e.g.*, user/item), and then pulls the views of the same instance close while pushing the views of different instances apart in a representation space. Since the information of each instance is enriched from various perspectives via the views, CL can help alleviate the data sparsity problem, thereby improving the accuracy of the model. Therefore, CL-based methods have focused on generating views that effectively represent the *characteristics of each instance* for their downstream tasks [3, 18, 22, 26, 27, 29, 32–34, 36].

Most CL-based multimedia recommender systems generate *user/item views* by utilizing the multimodal features of items via either a *non-graph-based approach* [26, 36] or a *graph-based approach* [3, 22, 32]. For instance, in a non-graph-based approach, MMSSL [26] generates dense user–item interaction matrices predicted from *all* modality features of items, performs adversarial learning [20] between each of these matrices and the original user–item interaction matrix, and uses the user/item embeddings obtained from the adversarial learning as the user/item views. In a graph-based

[1] Multimedia recommender systems do not focus on how to extract the multimodal features of items [5, 12, 13, 26, 35, 36].

approach, SLMRec [22] and MMGCL [32] first construct a bipartite graph (*modality graph*, for short) by using user–item interactions for each modality. In other words, all modality graphs have the *same* graph structure. Next, they randomly mask a whole modality graph [22, 32], or randomly drop some edges from each modality graph [32], referred to as a *random strategy*. Then, they generate user/item views by applying Graph Convolutional Network (GCN) [14] to the modified modality graphs.

Although existing CL-based multimedia recommendation methods in the above two approaches have made efforts to effectively generate user/item views, we claim that they have the following *two limitations in generating their views*. First, they either have *not* attempted to identify the *influence of each modality feature* of an item on user–item interactions [3, 26, 36], or have identified the influence via an *indiscriminate random strategy* [22, 32]. The indiscriminate random strategy can lead to an undesirable scenario where the influential modality features of an item are dropped while the uninfluential modality features are retained for each interaction, rather resulting in poorer outcomes than a scenario in which no attempt is made to identify the influence. Second, they have *not* attempted to identify the *item a user has not interacted with but is likely to interact with in the future*, given the user's preferred feature in each modality and each modality feature of the item [3, 22, 26, 32, 36]. Our claims about these limitations are supported by our empirical findings, which will be detailed in Section 3.

In this paper, we aim to recommend accurate items to a user by generating *user/item views* that *accurately capture the user's preferred feature in each modality* based on the following two ideas in a *graph-based approach*.

(Idea 1) Interaction Refinement. Based on a user's behavior, her interaction may have arisen from one of the following three causes:

- **Cause (i)**: A preference for *all* modality features of the item she interacted with.
- **Cause (ii)**: A preference for *some* modality feature(s) of the item she interacted with.
- **Cause (iii)**: *No* preference for *any* modality features of the item, but external factors (*e.g.*, a friend's recommendation).

For each interaction, accurately identifying the influence of each modality feature of the item, and then generating user/item views by using only the influential modality feature(s) of the item will help accurately capture the user's preferred feature in each modality.

(Idea 2) Non-Interaction Refinement. We note that there may be the following two cases of a user–item pair, *i.e.*, *non-interactions*, such that the user has not interacted with the item:

- **Case (i)**: A case that should *remain* a non-interaction (*e.g.*, the user is aware of the existence of an item but does not interact with it because she does *not prefer* any of all modality features of the item).
- **Case (ii)**: A case that is *likely to result in interaction* in the future (*e.g.*, the user is unaware of the existence of an item, while *potentially preferring* all modality features of the item).

The cause of each case can differ (*e.g.*, case (ii) may occur as the user potentially prefers some modality feature(s) of the non-interacted item). Therefore, for each non-interaction, identifying the case and its cause, and then generating user/item views by using user–item non-interactions for *case* (ii) will help accurately capture the user's preferred feature in each modality.

To this end, we propose a novel multimedia recommendation framework, named **STARLINE**, utilizing contra**ST**ive le**AR**ning with moda**LI**ty-aware graph refi**NE**ment (LINE) that effectively refines each modality graph by *carefully* and *flexibly* dropping original edges and adding necessary edges. It is worth mentioning that in *other domains*, such as CF recommendation, some graph-based methods [10, 24, 31] have attempted to refine interactions and non-interactions, mostly relying *solely* on the similarity between user and item embeddings (*similarity-based strategies*, for short); however, a similarity-based strategy *alone* is insufficient for careful and flexible refinement because (1) an interaction (*resp.* a non-interaction) with high similarity between the embeddings of a user and an item may need to be dropped (*resp.* to remain unadded) in some cases, and (2) this approach has *constraints*, such as fixed and equal numbers of interactions to drop and non-interactions to add. These claims will be empirically validated in Section 5 and Appendix A. If necessary, STARLINE can drop edges (*i.e.*, interactions) with high similarity from all modality graphs and add new edges for non-interactions with high similarity to only some modality graph(s), allowing a varying number of edges in each refined modality graph. As a result, STARLINE accurately captures the three causes of a user's interactions, and the two cases and their causes in her non-interactions.

Our contributions are summarized as follows:

- **Important Discovery**: We discovered that most existing CL-based multimedia recommender systems have difficulty in generating user/item views that accurately capture a user's preferred feature in each modality, due to not considering (Idea 1) and (Idea 2) in terms of interactions and non-interactions, respectively.
- **Novel Framework**: To overcome these limitations, we propose a novel multimedia recommendation framework, STARLINE, which utilizes CL with LINE. STARLINE is the first to carefully and flexibly drop and add interactions and non-interactions to generate effective user/item views to get higher accuracy.
- **Extensive Evaluation**: We validate the effectiveness and validity of STARLINE via extensive experiments on five real-world datasets. Above all, STARLINE consistently outperforms all thirteen state-of-the-art competitors, achieving up to 13.24% higher accuracy compared to the best competitor.

2 Related Work

In this section, we briefly review existing recommender systems that modify or refine graphs and discuss their limitations.

Non-CL-Based Recommender Systems. These methods generally modify or refine the user–item interaction bipartite graph (*user–item interaction graph*, for short) to generate final user/item embeddings for prediction. Specifically, the CF method EGLN [31] calculates the similarity between (learned) user and item embeddings, and then connects each user node to its top-q most similar item nodes with additional weighted edges. Multimedia recommendation methods GRCN [28] and FREEDOM [37] also refine the user–item interaction graph (rather than the modality graphs). GRCN first obtains user and item embeddings by applying GCN to the modality graphs. Then, it uses the similarities between these

user and item embeddings as edge weights in the user–item interaction graph. FREEDOM drops edges that connect high-degree nodes in the user–item interaction graph.

CL-Based Recommender Systems. These methods generally modify or refine the user–item interaction graph or modality graphs to generate user and item views. One CF method SGL [29] applies a random strategy (in Section 1) to the user–item interaction graph. Another CF method RGCF [24] calculates the similarity between the (learned) user and item embeddings. Then, it connects the following user–item pairs with edges: (1) user–item interactions with their similarities exceeding a threshold, and (2) randomly-selected user–item non-interactions with top-q similarities. A further CF method LightGCL [1] reconstructs the adjacency matrix of the user–item interaction graph by employing truncated Singular Value Decomposition (SVD). Social recommendation method SHaRe [10], which utilizes the social relations of users, drops edges with negative similarity based on the similarity between user embeddings in the user–user social graph, then adds as many new user–user edges with the highest similarities as dropped pairs, and assigns the similarity to them as edge weights. Multimedia recommendation methods SLMRec [22] and MMGCL [32] apply a random strategy to modality graphs, as mentioned in Section 1.

Discussions. Most of the above methods modify graphs using a random strategy [22, 29, 32] or refine them via similarity-based strategies [10, 24, 31]. These methods may result in unnecessary edge drops/additions while missing necessary ones. This is because a random strategy performs edge drops/additions without considering the user's preferred feature in each modality and each modality feature of the item, while similarity-based strategies rely *solely* on high similarity; both random and similarity-based strategies are constrained by fixed and equal numbers of interactions/non-interactions to be refined for all modality graphs. Note that the influence of each modality feature of an item on a user is represented as an edge between the user and item nodes in each modality graph. Yet, degree-based or SVD-based graph refinement methods [1, 37] deterministically refine original modality graphs that have the same topology in common, thereby producing the same modality graphs after refinement. In other words, these methods cannot reflect such varying influences depending on the modality.

3 Motivation

In this section, we demonstrate the limitations of existing CL-based multimedia recommender systems in generating *user/item views* via experiments answering the following preliminary questions (PQs):

- **PQ1**: Does identifying the influence of modality features of items via a *random strategy* hinder existing methods in a graph-based approach from improving accuracy?
- **PQ2**: Does treating *all* user–item non-interactions *solely* as case (i) of (Idea 2) in Section 1 hinder the existing models from improving accuracy?

Experimental Settings. We used five real-world Amazon datasets (*spec.*, Baby, Beauty, Toys & Games, Women Clothing, and Men Clothing categories) [19] and the accuracy metrics precision@10, recall@10, and normalized discounted cumulative gain (NDCG)@10, which are commonly used in (CL-based) multimedia recommendation [3, 5, 12, 13, 26, 35, 36]. We present only the results on Baby in terms of recall@10 in this section, as the results for other categories and metrics show similar tendencies. We conducted comparative experiments between the MMGCL [32] model in a graph-based approach and its variants, as MMGCL is the only model that randomly drops q'% of the edges (q edges, for short) from each modality graph to generate user/item views. The *variants* of MMGCL are equipped with *different view generation approaches*, inspired by the *similarity-based graph refinement methods* of EGLN [31], RGCF [24], and SHaRe [10] in Section 2. All variants will be explained in more detail in each PQ. For a fair comparison, we tuned the hyperparameters of MMGCL via extensive grid search and used the same best hyperparameters in all variants. We report the average of values obtained by five independent evaluations. Please refer to Section 5 for more detailed information on experimental settings.

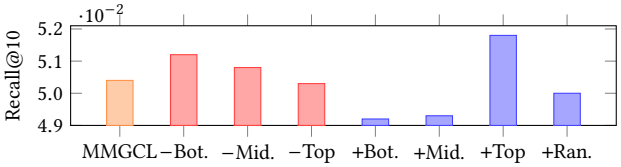


Figure 1: Accuracies of MMGCL when varying view generation approaches. All improvements are statistically significant with a p-value ≤ 0.05.

PQ1: Analysis of Interaction Refinement Approach. To answer PQ1, we considered three MMGCL variants −Bot., −Mid., and −Top, which *drop* the q edges with the lowest, median, and the highest cosine similarities, respectively, between the user and item embeddings for every interaction from each modality graph.

Figure 1 shows the accuracies of MMGCL and the three variants. From Figure 1, our empirical findings are summarized as follows:

(i) MMGCL shows lower accuracy than −Bot. and −Mid.. This is because an indiscriminate random edge drop of MMGCL can drop not only the uninfluential modality features of items but also the influential modality features. From these results, we have validated that generating user/item views via an indiscriminate random strategy limits the improvement of accuracy.

(ii) The accuracy improves in the order of −Top, −Mid., and −Bot.. This supports the importance of dropping uninfluential modality features of the item carefully by comparing the user's preferred feature (*i.e.*, user embedding) with the item's feature (*i.e.*, item embedding) for each modality and for each interaction.

PQ2: Effect of Non-Interaction Refinement. To answer PQ2, we considered four MMGCL variants +Bot., +Mid., +Top, and +Ran.. Variants +Bot., +Mid., and +Top *add* q non-interactions with the lowest, median, and the highest cosine similarities, respectively, between the user and item embeddings as additional edges in each modality graph; variant +Ran. adds q non-interactions selected randomly as additional edges in each modality graph.[2]

Figure 1 also shows the accuracies of the four variants, which give the following empirical findings:

(i) MMGCL shows lower accuracy than +Top. This indicates that refining non-interactions likely to result in interactions (*i.e.*, case (ii) of (Idea 2) in Section 1) improves accuracy, validating that

[2]All these four variants add the edges for non-interactions to the modality graphs from which MMGCL has randomly dropped q edges (as per the original method).

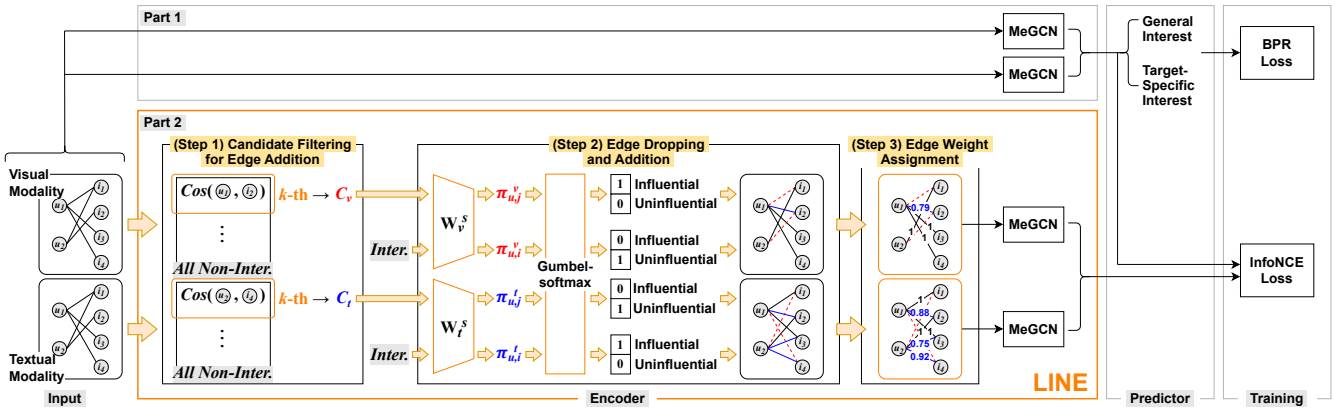


Figure 2: Overview of STARLINE composed of three key components: an encoder using LINE, a predictor considering both general and target-specific interests of users, and training based on both BPR loss and InfoNCE loss.

generating user/item views solely from user–item interactions limits the improvement of accuracy.

(ii) +Bot., +Mid., and +Ran. show lower accuracies than MMGCL, with the accuracy gap being *particularly pronounced* in +Bot. and +Mid.. These results suggest that the variants are confused by mistaking the modality features of items in non-interactions unlikely to result in interactions for those likely to result in interactions in the future. Hence, carefully selecting the modality features from non-interactions likely to result in interactions is crucial for improving accuracy.

Inspired by all the above findings, we design our refined modality graphs, which involve carefully dropping original edges and carefully adding potentially necessary but missing edges to generate effective user/item views. Although the above similarity-based strategies derived from EGLN, RGCF, and SHaRe performed better than the random-based one in graph refinement, we aim to propose a novel method that refines the graph *more carefully* and *flexibly beyond similarity-based ones*, incorporating new strategies to meticulously handle interactions/non-interactions without constraints.

4 Proposed Framework: STARLINE

In this section, we detail our proposed framework, STARLINE, which utilizes CL with *modality-aware graph refinement* (LINE).

4.1 Problem Definition

Let $\mathcal{U}$ and $\mathcal{I}$ denote a set of all users and a set of all items, respectively. For each user $u \in \mathcal{U}$, $\mathcal{N}_u$ denotes a set of items that u has interacted with. For each item $i \in \mathcal{I}$, $\mathcal{N}_i$ denotes a set of users who have interacted with i. In this paper, we assume that each item i is associated with its pre-obtained visual and textual modality features [3, 12, 13, 26, 35, 36].[3] Formally, an item i has modality feature $\tilde{\mathbf{e}}_{i,m} \in \mathbb{R}^{d_m}$, where modality $m \in \{v, t\}$ denotes a modality indicator, v and t denote visual and textual modalities, respectively, and d_m denotes the dimensionality of feature w.r.t. modality m. The goal of multimedia recommendation is to recommend the top-N items that a user u is most likely to prefer among her non-interacted items $j \in \mathcal{I} \setminus \mathcal{N}_u$, by using not only the user–item interactions but also the multimodal features of items. We summarize the key notations used in this paper in Table 1.

[3]We obtain modality features using the same methods as in previous studies [3, 12, 13, 35]. The additional modality features can also be incorporated easily into STARLINE.

Table 1: Key notations used in this paper

Notation	Description
$\tilde{\mathbf{e}}_{i,m}$	Modality m's feature of item i
$\mathbf{e}^0_{u,m}, \mathbf{e}^0_{i,m}$	Initial embeddings of user u and item i learned w.r.t. modality m
$\mathbf{e}^l_{u,m}, \mathbf{e}^l_{i,m}$	Embeddings of user u and item i w.r.t. modality m at the l-th MeGCN-layer
$\mathbf{e}_u, \mathbf{e}_i$	(Final) embeddings of user u and item i
$\mathbf{e}'_u, \mathbf{e}'_i$	Views of user u and item i
d_m	Dimensionality of $\tilde{\mathbf{e}}_{i,m}$
d	Dimensionality of $\mathbf{e}^0_{u,m}, \mathbf{e}^0_{i,m}, \mathbf{e}^l_{u,m}$, and $\mathbf{e}^l_{i,m}$
$\mathcal{G}, \mathcal{G}_m$	Modality m's graph and refined graph
$\mathcal{E}, \mathcal{E}_m$	Sets of edges in $\mathcal{G}$ and $\mathcal{G}_m$
$s^m_{u,j}$	Cosine similarity between embeddings of the user u and non-interacted item j w.r.t. modality m
C_m	Candidate user–item non-interaction pairs w.r.t. modality m
$\pi^m_{u,i}$	Scores between user u and interacted item i for edge dropping w.r.t. modality m
$\pi^m_{u,j}$	Scores between user u and non-interacted item j for edge addition w.r.t. modality m
$\hat{y}_{u,j}$	Preference of user u for a non-interacted item j
k	The number of equal-width intervals
α, β	Weights for CL and regularization

4.2 Key Components

The overall flow of STARLINE is presented in Figure 2. STARLINE consists of the following three key components: encoder, predictor, and training. In the encoder, STARLINE generates user embeddings $\mathbf{e}_u \in \mathbb{R}^{2d}$ and user views $\mathbf{e}'_u \in \mathbb{R}^{2d}$ for $\forall u \in \mathcal{U}$, and item embeddings $\mathbf{e}_i \in \mathbb{R}^{2d}$ and item views $\mathbf{e}'_i \in \mathbb{R}^{2d}$ for $\forall i \in \mathcal{I}$, where d denotes the dimensionality of embedding and view for each modality, by utilizing all modality graphs $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = (\mathcal{U} \cup \mathcal{I})$ and $\mathcal{E} = \{(u, i) \mid u \in \mathcal{U}, i \in \mathcal{N}_u\}$ denote a set of nodes and a set of edges, respectively. Recall that all (original) modality graphs have the same graph structure. In the predictor, STARLINE predicts a user u's preference $\hat{y}_{u,j}$ for a non-interacted item j by using the user embedding $\mathbf{e}_u$ and the item embedding $\mathbf{e}_j$ generated by the

encoder for $\forall u \in \mathcal{U}$ and $\forall j \in \mathcal{I} \backslash \mathcal{N}_u$. In training, STARLINE learns all its trainable parameters, by utilizing the Bayesian Personalized Ranking (BPR) loss and the InfoNCE loss (for CL).

4.2.1 ***Encoder.*** STARLINE's encoder is divided into the following two parts: (Part 1) generates user and item embeddings $\mathbf{e}_u$ and $\mathbf{e}_i$, and (Part 2) generates user and item views $\mathbf{e}'_u$ and $\mathbf{e}'_i$.

(Part 1) User and Item Embeddings. Recently, some studies found that reflecting not only collaborative signals but also multimodal features in the user and item embeddings $\mathbf{e}_u$ and $\mathbf{e}_i$ helps improve accuracy [12, 13]. Therefore, STARLINE utilizes MeGCN [13], designed to reflect both collaborative signals and multimodal features in the user and item embeddings $\mathbf{e}_u$ and $\mathbf{e}_i$.

Specifically, for each modality graph $\mathcal{G}$, STARLINE first randomly initializes learnable user embeddings $\mathbf{e}^0_{u,m} \in \mathbb{R}^d$ for $\forall u \in \mathcal{U}$, and initializes learnable item embeddings $\mathbf{e}^0_{i,m} \in \mathbb{R}^d$ for $\forall i \in \mathcal{I}$ as the linear transformation of their modality m's features $\tilde{\mathbf{e}}_{i,m}$, *i.e.*, $\forall m \in \{v, t\}$, $\mathbf{e}^0_{i,m} = \mathbf{W}_m \tilde{\mathbf{e}}_{i,m} + \mathbf{b}_m$, where $\mathbf{W}_m \in \mathbb{R}^{d \times d_m}$ and $\mathbf{b}_m \in \mathbb{R}^d$ denote a trainable weight matrix and a bias vector, respectively [3, 5, 12, 13, 35, 36]. Then, STARLINE independently applies MeGCN to each modality graph $\mathcal{G}$. Formally, user embeddings $\mathbf{e}^l_{u,m}$ generated by the l-th MeGCN-layer ($1 \le l \le L$, where L is the number of layers) in modality graph $\mathcal{G}$ can be expressed as follows: $\forall m \in \{v, t\}$,

$$\mathbf{e}^l_{u,m} = \sum_{i \in \mathcal{N}_u} \frac{1}{\sqrt{|\mathcal{N}_u||\mathcal{N}_i|}} \mathbf{e}^{l-1}_{i,m} + \mathbf{e}^{l-1}_{u,m}. \tag{1}$$

All item embeddings $\mathbf{e}^l_{i,m}$ of all modality graphs $\mathcal{G}$ are obtained by the l-th MeGCN-layer, similarly to Eq. (1). Lastly, STARLINE generates the final user (*resp.* item) embeddings $\mathbf{e}_u$ (*resp.* $\mathbf{e}_i$), which are the concatenation of user (*resp.* item) embeddings obtained from the last L-th MeGCN-layers on all modality graphs $\mathcal{G}_m$, *i.e.*, $\mathbf{e}_u = concat(\mathbf{e}^L_{u,v}, \mathbf{e}^L_{u,t})$ (*resp.* $\mathbf{e}_i = concat(\mathbf{e}^L_{i,v}, \mathbf{e}^L_{i,t})$) [13].

(Part 2) User and Item Views. In Section 3, we have validated that in generating user and item views $\mathbf{e}'_u$ and $\mathbf{e}'_i$, refining the interactions and non-interactions is important for improving accuracy. Therefore, STARLINE refines modality graphs $\mathcal{G}$ into $\mathcal{G}_m = (\mathcal{V}, \mathcal{E}_m)$ for $\forall m \in \{v, t\}$ by carefully and flexibly dropping noisy edges from each modality graph $\mathcal{G}$ and adding necessary edges via the following *three steps* of LINE, and generates $\mathbf{e}'_u$ and $\mathbf{e}'_i$ from $\mathcal{G}_m$.

(Step 1) Candidate Filtering for Edge Addition. For each modality m, STARLINE obtains a set C_m of *candidate* user–item non-interaction pairs that have the potential to result in interactions in the future (*i.e.*, case (ii) of (Idea 2) in Section 1).

Specifically, for every user–item non-interaction pair (u, j), STARLINE calculates the cosine similarity $s^m_{u,j} = \frac{\mathbf{e}^{0\,\intercal}_{u,m} \mathbf{e}^0_{j,m}}{\|\mathbf{e}^0_{u,m}\| \|\mathbf{e}^0_{j,m}\|}$ between the initial user and item embeddings $\mathbf{e}^0_{u,m}$ and $\mathbf{e}^0_{j,m}$ being *learned* for each modality m. Subsequently, STARLINE partitions the range $[-1, 1]$ of $s^m_{u,j}$ into k ($k \ge 2$) equal-width intervals, *i.e.*, $[-1, -1 + 2/k]$, $[-1+2/k, -1+4/k]$, ..., $[1-2/k, 1]$, and defines C_m as follows: $\forall m \in \{v, t\}$,

$$C_m = \{(u, j) \mid s^m_{u,j} \in \text{the } k\text{-th interval}\}. \tag{2}$$

In other words, the candidate non-interaction pairs (u, j) in C_m are those with the *highest* $s^m_{u,j}$. Note that C_v may differ from C_t.

Sets C_m for $\forall m \in \{v, t\}$ are updated per iteration of training, based on $\mathbf{e}^0_{u,m}$ and $\mathbf{e}^0_{j,m}$ being learned. Instead of this real-time selection approach, we considered a pre-selection approach as an alternative, which uses fixed C_m for $\forall m \in \{v, t\}$ based on the similarity between (1) the average of the raw modality m's features of items which u interacted with [3, 26] and (2) the raw modality m's feature $\tilde{\mathbf{e}}_{j,m}$ of j. However, we ultimately adopted the real-time selection approach due to its higher accuracy (0.56%) and training time (1.2 times) comparable with the pre-selection approach.

(Step 2) Edge Dropping and Addition. For *each* modality m, STARLINE obtains the set $\mathcal{E}_m$ of edges in $\mathcal{G}_m$ by *dropping* the edges $(u, i) \in \mathcal{E}$ between user u and item i whose modality m's feature was *uninfluential* to u, and adding the *candidate* non-interaction pairs (u, j) in C_m between a user u and an item j whose modality m's feature is expected to be *influential* as edges.

To this end, for each modality m, STARLINE first calculates the vectors of *two scores* $\boldsymbol{\pi}^m_{u,i} \in \mathbb{R}^2$ for edge dropping and $\boldsymbol{\pi}^m_{u,j} \in \mathbb{R}^2$ for edge addition. Given each user u and modality m's feature of an item i that u has *interacted* with,[4] $\boldsymbol{\pi}^m_{u,i}$ denotes how influential and uninfluential that feature was to u. Given each user u and modality m's feature of an item j that u has *not* interacted with,[5] $\boldsymbol{\pi}^m_{u,j}$ denotes how likely and unlikely that feature is to be influential to u. Formally, $\forall m \in \{v, t\}$, $\forall u \in \mathcal{U}$, and $\forall i \in \mathcal{N}_u$ (*resp.* $\forall j \in \mathcal{I} \backslash \mathcal{N}_u$ such that $(u, j) \in C_m$),

$$\boldsymbol{\pi}^m_{u,i}(resp.\ \boldsymbol{\pi}^m_{u,j}) = concat(\mathbf{e}^0_{u,m}, \mathbf{e}^0_{i,m}(resp.\ \mathbf{e}^0_{j,m}))\mathbf{W}^s_m + \mathbf{b}^s_m, \tag{3}$$

where $\mathbf{W}^s_m \in \mathbb{R}^{2d \times 2}$ and $\mathbf{b}^s_m \in \mathbb{R}^2$ denote a trainable weight matrix and a bias vector, respectively.

Then, STARLINE applies a *Gumbel-softmax* [9] to $\boldsymbol{\pi}^m_{u,i}$ (*resp.* $\boldsymbol{\pi}^m_{u,j}$) to sample modality features that are uninfluential (*resp.* likely to be influential), while allowing for the flow of gradients; finally, STARLINE obtains $\mathcal{E}_m$ in $\mathcal{G}_m$ for $\forall m \in \{v, t\}$ by dropping the uninfluential edges (u, i) from each modality graph $\mathcal{G}$ and adding new edges (u, j) for the candidate pairs likely to be influential. In other words, $\forall m \in \{v, t\}$, $\forall i \in \mathcal{N}_u$, and $\forall j \in \mathcal{I} \backslash \mathcal{N}_u$ such that $(u, j) \in C_m$,

$$\mathcal{E}_m = \mathcal{E} \backslash \{(u, i) \mid g^m_{u,i,2} \ge 0.5\} \cup \{(u, j) \in C_m \mid g^m_{u,j,1} \ge 0.5\},$$
$$\text{where } g^m_{u,x,\gamma} = \frac{exp((\log \boldsymbol{\pi}^m_{u,x,\gamma} + z^m_{u,x,\gamma})/\tau_g)}{\sum^2_{\delta=1} exp((\log \boldsymbol{\pi}^m_{u,x,\delta} + z^m_{u,x,\delta})/\tau_g)}, \tag{4}$$

where $x \in \{i, j\}$, $\gamma \in \{1, 2\}$, $\boldsymbol{\pi}^m_{u,x,1}$ and $\boldsymbol{\pi}^m_{u,x,2}$ denote the scores for how influential and uninfluential, respectively, modality m's feature of item x is to u, $z^m_{u,x,\gamma}$ denotes the γ-th sample from the $Gumbel(0, 1)$ distribution for the user–item pair (u, x) and modality m, and τ_g is the temperature hyperparameter.

(Step 3) Edge Weight Assignment. STARLINE assigns weights to edges $\mathcal{E}_m$ in $\mathcal{G}_m$ as follows: $\forall m \in \{v, t\}$ and $\forall (u, i) \in \mathcal{E}_m$,

$$\boldsymbol{A}_m[u, i] = \begin{cases} 1, & \text{if } (u, i) \in \mathcal{E}, \\ s^m_{u,i}, & \text{otherwise.} \end{cases} \tag{5}$$

Significance. As demonstrated by finding (ii) of PQ2 in Section 3, incorrect edge additions can significantly degrade accuracy. Therefore, via (Step 1), STARLINE computes a set of candidate non-interactions by filtering out non-interactions that may negatively affect accuracy

[4] Although a user may have a different perception of each item in a modality, we assume that a user's perception of a modality is the same, following most multimedia recommender systems [3, 5, 12, 22, 26, 32, 35, 36].

(*i.e.*, those with low similarity). Then, via (Step 2), STARLINE carefully and flexibly drops interactions and adds the non-interactions selected among the candidates by utilizing the Gumbel-softmax, without any constraints and without relying solely on their similarities. Lastly, via (Step 3), STARLINE regards the original edges retained after refinement as fully reliable since they originate from actual interactions; the edges added via refinement are less reliable than those from actual interactions, but an edge with a larger weight represents a non-interaction more likely to become an interaction in the future. The effect of such design will be validated empirically via extensive experiments in Section 5.

Given the final refined modality graphs $\mathcal{G}_m$ for $\forall m \in \{v, t\}$, STARLINE initializes user and item embeddings in $\mathcal{G}_m$ as $\mathbf{e}^0_{u,m}$ and $\mathbf{e}^0_{i,m}$, respectively, and generates user and item views $\mathbf{e}'_u$ and $\mathbf{e}'_i$ via the same process as (Part 1).[5]

4.2.2 ***Predictor.*** STARLINE predicts a user u's preference $\hat{y}_{u,j}$ for a non-interacted item (*i.e.*, target item) j, by utilizing not only u's general interest (*i.e.*, the user embedding $\mathbf{e}_u$) in the multimodal features of the items that u interacted with, but also u's specific interest $\mathbf{e}^j_u \in \mathbb{R}^{2d}$ in the multimodal features of j, as this combination helps with accurate recommendations [13].

Formally, STARLINE predicts u's preference $\hat{y}_{u,j}$ for j as follows [13]: $\forall j \in \mathcal{I} \backslash \mathcal{N}_u$,

$$\hat{y}_{u,j} = (1-\lambda)\mathbf{e}_u^\intercal \mathbf{e}_j + \lambda \mathbf{e}_u^{j\,\intercal} \mathbf{e}_j,$$
$$\text{where } \mathbf{e}^j_u = \sum_{i \in \mathcal{N}_u} \frac{exp(\mathbf{e}_j^\intercal \mathbf{e}_i)}{\sum_{h \in \mathcal{N}_u} exp(\mathbf{e}_j^\intercal \mathbf{e}_h)} \mathbf{e}_i, \tag{6}$$

where hyperparameter $\lambda \in (0, 1)$ denotes a coefficient that controls the balance between the user u's general (*i.e.*, $\mathbf{e}_u^\intercal \mathbf{e}_j$) and target-specific (*i.e.*, $\mathbf{e}_u^{j\,\intercal} \mathbf{e}_j$) interests.[6]

4.2.3 ***Training.*** To learn all trainable parameters, STARLINE employs the BPR loss and the InfoNCE loss [3, 22, 26, 32, 36].

BPR Loss. The BPR loss ensures that a user u's preference $\hat{y}_{u,i}$ for an interacted item i is higher than u's preference $\hat{y}_{u,j}$ for a (randomly-selected) non-interacted item j as follows [21]:

$$\mathcal{L}_{BPR} = -\sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{N}_u} \sum_{j \in \mathcal{I} \backslash \mathcal{N}_u} \ln sigmoid(\hat{y}_{u,i} - \hat{y}_{u,j}). \tag{7}$$

InfoNCE Loss. Note that STARLINE's user and item embeddings $\mathbf{e}_u$ and $\mathbf{e}_i$ effectively represent the *characteristics* (*i.e.*, both collaborative signals and multimodal features) of u and i, respectively. Thus, STARLINE employs these user and item embeddings $\mathbf{e}_u$ and $\mathbf{e}_i$ in the InfoNCE loss (*i.e.*, treating them as other user and item views) as well as in the BPR loss. Ultimately, the InfoNCE loss pulls the views $\mathbf{e}_u$ (*resp.* $\mathbf{e}_i$) and $\mathbf{e}'_u$ (*resp.* $\mathbf{e}'_i$) of the same user u (*resp.* item i) close while pushing the views from different users (*resp.* items) apart, as follows [4]:

$$\mathcal{L}_{CL} = \mathcal{L}^{user}_{CL} + \mathcal{L}^{item}_{CL},$$
$$\text{where } \mathcal{L}^{user}_{CL} = \sum_{u \in \mathcal{U}}^{|\mathcal{U}|} -\log \frac{exp(sim(\mathbf{e}_u, \mathbf{e}'_u)/\tau_{CL})}{\sum_{w \in \mathcal{U}}^{|\mathcal{U}|} exp(sim(\mathbf{e}_u, \mathbf{e}'_w)/\tau_{CL})} \text{ and} \tag{8}$$
$$\mathcal{L}^{item}_{CL} = \sum_{i \in \mathcal{I}}^{|\mathcal{I}|} -\log \frac{exp(sim(\mathbf{e}_i, \mathbf{e}'_i)/\tau_{CL})}{\sum_{h \in \mathcal{I}}^{|\mathcal{I}|} exp(sim(\mathbf{e}_i, \mathbf{e}'_h)/\tau_{CL})},$$

where $sim(\cdot)$ measures the cosine similarity between two views and τ_{CL} denotes the temperature hyperparameter.

Final Loss. The final loss of STARLINE can be expressed as follows [3, 22, 26, 32, 36]:

$$\mathcal{L} = \mathcal{L}_{BPR} + \alpha \mathcal{L}_{CL} + \beta \|\theta\|_2^2, \tag{9}$$

where α and β denote the weights for CL and regularization, respectively, and θ denotes all trainable parameters of STARLINE.

Table 2: Dataset statistics

Dataset	# Users	# Items	# Inter.	Sparsity	Dim. of v / t
Baby	19,445	7,050	160,792	99.88%	4,096 / 1,024
Beauty	22,363	12,101	198,502	99.93%	4,096 / 1,024
Toys & Games	19,412	11,924	167,597	99.93%	4,096 / 1,024
Women Clothing	19,244	14,596	135,326	99.95%	4,096 / 1,024
Men Clothing	4,955	5,028	32,363	99.87%	4,096 / 1,024

4.3 Time Efficiency

Each step in LINE has the following time complexity:

- **(Step 1)**: $O((d+1)|\mathcal{U}||\mathcal{I}|)$
- **(Step 2)**: $O(d(2|\mathcal{E}| + |C_v| + |C_t|))$
- **(Step 3)**: $O(\mathcal{E}_v + \mathcal{E}_t)$

Although the time complexity of LINE does not significantly differ from that of the similarity-based graph refinement method used in the state-of-the-art recommender system SHaRe [10], STARLINE gains considerable benefits in training time per epoch compared to SHaRe (*e.g.*, about 2.3 times faster on the Baby dataset). This improvement is due to STARLINE employing *equal-width* partitioning, which does not require *sorting* the data, whereas SHaRe uses equal-depth partitioning that involves sorting, leading to increased computational overhead. It is also notable that STARLINE provides up to a 3.50% improvement in terms of recall@10 on Baby compared to SHaRe, which will be discussed in more detail in Section 5.

5 Evaluation

In this section, we validate the effectiveness and validity of STARLINE by answering the following key research questions (RQs):

- **RQ1**: Does STARLINE recommend more-accurate top-N items to users than state-of-the-art recommender systems?
- **RQ2**: Is STARLINE's core idea, LINE, effective in improving the accuracy?
- **RQ3**: Is STARLINE effective in alleviating the data sparsity problem?
- **RQ4**: How sensitive is the accuracy of STARLINE to hyperparameter k?

5.1 Experimental Settings

Datasets. We used the following five categories of the real-world Amazon dataset, which are commonly used in existing (CL-based)

[5] In this process, the numerator value (*i.e.*, 1) of the normalization term for the neighbor aggregation part in Eq. (1) is replaced by the weight of the corresponding edge.

[6] To comprehensively validate the design of STARLINE, we compared our current predictor with a variant that concatenates a user's (*resp.* an item's) view with her (*resp.* its) final embedding to predict the user's preference on the item based on these concatenated embeddings. This variant, however, showed a negative effect on accuracy.

Table 3: Accuracies of thirteen competitors and STARLINE on five datasets. The P@10 values of all methods are adjusted by multiplying by 10 for ease of comparison.

Datasets	Baby			Beauty			Toys & Games			Women Clothing			Men Clothing		
Metrics	P@10	R@10	NDCG@10	P@10	R@10	NDCG@10	P@10	R@10	NDCG@10	P@10	R@10	NDCG@10	P@10	R@10	NDCG@10
LightGCN (SIGIR'20)	0.0478	0.0451	0.0239	0.0973	0.0862	0.0471	0.0834	0.0781	0.0448	0.0434	0.0422	0.0237	0.0347	0.0339	0.0182
EGLN (SIGIR'21)	0.0474	0.0447	0.0241	0.1026	0.0877	0.0503	0.0872	0.0802	0.0466	0.0441	0.0437	0.0240	0.0347	0.0344	0.0179
SGL (SIGIR'21)	0.0508	0.0478	0.0256	0.1057	0.0930	0.0518	0.0900	0.0840	0.0478	0.0493	0.0487	0.0272	0.0383	0.0379	0.0213
RGCF (SIGIR'22)	0.0517	0.0502	0.0261	0.0945	0.0924	0.0519	0.0860	0.0860	0.0502	0.0508	0.0508	0.0283	0.0402	0.0402	0.0227
LightGCL (ICLR'23)	0.0551	0.0520	0.0274	0.1038	0.0916	0.0519	0.0926	0.0866	0.0499	0.0549	0.0538	0.0309	0.0409	0.0419	0.0231
GRCN (MM'19)	0.0532	0.0509	0.0278	0.0987	0.0893	0.0505	0.0919	0.0879	0.0504	0.0477	0.0486	0.0251	0.0353	0.0349	0.0179
LATTICE (MM'21)	0.0540	0.0515	0.0285	0.1044	0.0944	0.0543	0.0959	0.0917	0.0526	0.0513	0.0507	0.0282	0.0505	0.0507	0.0269
FREEDOM (MM'23)	0.0580	0.0552	0.0291	0.1060	0.0956	0.0522	0.0950	0.0899	0.0490	0.0580	0.0566	0.0313	0.0550	0.0539	0.0296
MONET (WSDM'24)	0.0614	0.0583	0.0330	0.1176	0.1055	0.0623	0.1123	0.1059	0.0623	0.0671	0.0669	0.0364	0.0581	0.0583	0.0317
MMGCL (SIGIR'22)	0.0530	0.0504	0.0266	0.1030	0.0902	0.0496	0.0870	0.0813	0.0449	0.0541	0.0535	0.0291	0.0421	0.0417	0.0224
MICRO (TKDE'22)	0.0580	0.0549	0.0309	0.1080	0.0982	0.0554	0.0972	0.0927	0.0528	0.0585	0.0579	0.0320	0.0525	0.0523	0.0283
MMSSL (WWW'23)	0.0615	0.0582	0.0312	0.1107	0.0972	0.0557	0.0964	0.0902	0.0516	0.0611	0.0603	0.0330	0.0534	0.0530	0.0289
LGMRec (AAAI'24)	0.0620	0.0594	0.0313	0.1150	0.1025	0.0562	0.1010	0.0953	0.0526	0.0620	0.0606	0.0336	0.0572	0.0568	0.0295
STARLINE	**0.0652**	**0.0621**	**0.0346**	**0.1258**	**0.1117**	**0.0666**	**0.1176**	**0.1108**	**0.0647**	**0.0694**	**0.0685**	**0.0377**	**0.0658**	**0.0655**	**0.0339**
Improvements (%)	**5.09**	**4.54**	**4.85**	**7.04**	**5.90**	**6.94**	**4.68**	**4.61**	**3.72**	**3.39**	**2.38**	**3.51**	**13.24**	**12.33**	**6.81**

multimedia recommender systems: Baby, Beauty, Toys & Games, Women Clothing, and Men Clothing; each category contains user–item interactions, where each of users and items has at least five interactions; every item of each category has the features of visual modality v and text modality t [3, 12, 13, 26, 35, 36]. Table 2 provides some statistics for each category.

Competitors. We compared STARLINE with thirteen state-of-the-art recommender systems. They can be divided into the following four groups based on the use of the multimodal features and the utilization of CL: (G1) Non-CL-based CF methods (LightGCN [6] and EGLN [31]), (G2) CL-based CF methods (SGL [29], RGCF [24], and LightGCL [1]), (G3) Non-CL-based multimedia recommender systems (GRCN [28], LATTICE [35], FREEDOM [37], and MONET [13]), and (G4) CL-based multimedia recommender systems (MMGCL [32], MICRO [36], MMSSL [26], and LGMRec [3]). For CF methods (*i.e.*, (G1) and (G2)), we only used interactions. For multimedia recommender systems (*i.e.*, (G3) and (G4)), we used not only interactions but also multimodal features. It is worth mentioning that we could not directly compare the state-of-the-art similarity-based graph refinement method SHaRe [10], as it is designed for social recommendation utilizing the social relations of users; instead, we considered the variants of STARLINE (*spec.*, (V2) $\text{LINE}_{\text{similarity}}$ and (V11) w/o $\text{CL}_{\text{similarity}}$) that are equipped with its graph refinement method in RQ2.

Evaluation Protocols. In each category, we split each user's interactions into a training set (80%), a validation set (10%), and a test set (10%) via random selection; subsequently, we recommended the top-10 items to each user by using each method and then evaluated the accuracy of each method in terms of precision, recall, and normalized discounted cumulative gain (NDCG) [3, 12, 13, 26, 35, 36]. In providing experimental results, we report the average of values obtained by performing five independent evaluations.

Hyperparameters. For a fair comparison, we set the dimensionality of user/item embeddings (and views) to 64 for all methods, following [3, 12, 13, 26, 35, 36]. For the remaining hyperparameters, we investigated their ranges by referring to the methods' original papers and carefully tuned the hyperparameters of all methods on the validation set. In particular, we set the hyperparameters of STARLINE as follows: $L = 2$; $k = 8$; $\tau_g = 0.1$ for Baby and Toys & Games, and 0.2 for Beauty, Women Clothing, and Men Clothing; $\lambda = 0.1$ for Baby, 0.35 for Beauty, 0.4 for Toys & Games, and 0.01 for Women Clothing and Men Clothing; $\tau_{CL} = 0.2$; $\alpha = 0.01$; $\beta = 0.00001$; learning rate = 0.0001.

5.2 Results and Analysis

For all RQs except RQ1, the results on Baby are included in this subsection, and those on Beauty, Toys & Games, and Women (*resp.* Men) Clothing are provided in the External Appendix[7], as the results on all the datasets generally showed similar tendencies. In all results, all improvements are *statistically significant* with a p-value ≤ 0.05. For simplicity, we represent precision@10 and recall@10 as P@10 and R@10, respectively, in the following tables and figures. For ease of comparison, we highlight the best and second-best results in each column (*i.e.*, metric) of the following tables in **bold** and underline, respectively; the P@10 values are adjusted by multiplying by 10.

RQ1: Comparison with Thirteen Competitors. As shown in Table 3, STARLINE *consistently* outperforms *all* competitors on *all* categories for *all* metrics, whereas the best performer among the competitors varies by category and metric. Specifically, on Baby, Beauty, Toys & Games, Women Clothing, and Men Clothing, STARLINE outperforms the best competitors LGMRec and MONET by up to 4.54%, 5.90%, 4.61%, 2.38%, and 12.33% in terms of R@10, respectively. Note that despite utilizing CL, MMGCL shows lower accuracy than not only all other CL-based but also most non-CL-based multimedia recommender systems on all categories for all metrics [3, 18, 26, 30]. This is because, in generating user/item views, MMGCL may drop the influential modality features of items for interactions by using an indiscriminate random strategy.

RQ2: Effectiveness of LINE. To generate *user/item views*, STARLINE *carefully* and *flexibly* refines each modality graph via its LINE, which consists of the following three steps: (Step 1) *Candidate Filtering for Edge Addition*, (Step 2) *Edge Dropping and Addition* (based on Gumbel-softmax), and (Step 3) *Edge Weight Assignment*. To validate the effectiveness and validity of the design choice of LINE, we compare STARLINE with its variants equipped with *modified versions* of LINE. In Table 4, we describe view generation approaches of these

[7] https://github.com/Bigdasgit/STARLINE.

Table 4: LINE and the view generation approaches of STARLINE's variants. (Step 1) is performed solely on non-interactions to derive candidate pairs. 'O' in the column of (Step 2) denotes that each interaction (*resp.* candidate pair) is carefully and flexibly refined via (Step 2). The parts modified in LINE are highlighted in orange.

Targets	Interactions (u, i)		Non-interactions (u, j)		
Steps	**2**	**3**	**1**	**2**	**3**
LINE	**O**	**1**	k-**th**	**O**	$s^m_{u,j}$
(V1) LINE$_{\text{random}}$ (MMGCL's Method)	Drop q Pairs Selected Randomly	1		—	
(V2) LINE$_{\text{similarity}}$ (SHaRe's Method)	Drop All Pairs with a Negative $s^m_{u,i}$ (q' Pairs, for short)	$s^m_{u,i}$	—	Add q' Pairs with the Highest $s^m_{u,j}$	$s^m_{u,j}$
(V3) LINE$_{\text{w/o +}}$	O	1		—	
(V4) LINE$_{\text{w/o −}}$	—		k-th	O	$s^m_{u,j}$
(V5) LINE$_{\text{(S2) GB w.r.t. +}}$	O	1	k-th	Add All from C_m	$s^m_{u,j}$
(V6) LINE$_{\text{(S2) GB w.r.t. −}}$	Drop All Pairs with a Negative $s^m_{u,i}$	1	k-th	O	$s^m_{u,j}$
(V7) LINE$_{\text{(S3) W. w.r.t. +}}$	O	1	k-th	O	1
(V8) LINE$_{\text{(S3) W. w.r.t. −}}$	O	$s^m_{u,i}$	k-th	O	$s^m_{u,j}$
(V9) LINE$_{\text{soft pruning}}$	—	$s^m_{u,i}$	k-th	O	$s^m_{u,j}$

variants in comparison to LINE. Note that the variants related to (Step 1) will be addressed in RQ4. Additionally, we consider the following four variants of STARLINE:

- **(V10) w/o CL$_{\text{LINE}}$** (*resp.* **(V11) w/o CL$_{\text{similarity}}$**) does *not* utilize CL and *replaces* user/item embeddings with the user/item views obtained from the modality graphs refined via LINE (*resp.* the similarity-based strategy of SHaRe) in STARLINE.
- **(V12) w/o** v (*resp.* **(V13) w/o** t) does not use visual v (*resp.* textual t) modality features of items in STARLINE.

Table 5 shows all results except for the (Step 1) variants, for which our findings are summarized as follows:

(i) STARLINE outperforms *all* variants for *all* metrics. In other words, STARLINE equipped with LINE is the most effective in improving accuracy, while supporting the validity of its design.

(ii) Note that STARLINE shows better accuracy than (V2) LINE$_{\text{similarity}}$ for all metrics. This result highlights the effectiveness of LINE, which refines the graph *more carefully* and *flexibly*, while validating that a similarity-based strategy *alone* is insufficient for accurate graph refinement.

(iii) Specifically, unlike (V2) LINE$_{\text{similarity}}$ which rigidly sets the *fixed* and *equal* numbers of interactions to drop and non-interactions to add, STARLINE flexibly drops interactions and adds non-interactions without such constraints thanks to the Gumbel-softmax mechanism. For instance, STARLINE added 1,768 non-interactions for the visual modality and 1,337 non-interactions for the textual modality on Baby (LINE's refinement is discussed in detail in Appendix A). This *flexible* refinement is crucial for improving model accuracy, as demonstrated by the superior performance of STARLINE.

(iv) The accuracy improves in the order of (V8) LINE$_{\text{(S3) W. w.r.t. −}}$, (V7) LINE$_{\text{(S3) W. w.r.t. +}}$, and STARLINE for all metrics. This shows that it is important in terms of accuracy to fully rely on (*i.e.*, edge weight = 1) the original edges retained after refinement since they originate from actual interactions. Paradoxically, this result also means that the similarity between embeddings of the user and the item connected by an original edge (*i.e.*, an *actual interaction*) can be significantly *lower* than 1; indeed, in Baby, these similarities are 0.58 for the visual modality and 0.51 for the textual modality on average. Therefore, it is necessary to judge *more carefully* whether to refine interactions based on Gumbel-softmax as well as their similarities.

Table 5: The effects of STARLINE's core idea, LINE. The P@10 values of all methods are adjusted by multiplying by 10 for ease of comparison.

Method	P@10	R@10	NDCG@10
STARLINE	**0.0652**	**0.0621**	**0.0346**
(V1) LINE$_{\text{random}}$	0.0617	0.0585	0.0330
(V2) LINE$_{\text{similarity}}$	0.0632	0.0600	0.0334
(V3) LINE$_{\text{w/o +}}$	0.0627	0.0597	0.0333
(V4) LINE$_{\text{w/o −}}$	0.0628	0.0599	0.0334
(V5) LINE$_{\text{(S2) GB w.r.t. +}}$	0.0633	0.0601	0.0334
(V6) LINE$_{\text{(S2) GB w.r.t. −}}$	0.0632	0.0599	0.0338
(V7) LINE$_{\text{(S3) W. w.r.t. +}}$	0.0643	0.0612	0.0339
(V8) LINE$_{\text{(S3) W. w.r.t. −}}$	0.0635	0.0604	0.0335
(V9) LINE$_{\text{soft pruning}}$	0.0630	0.0598	0.0333
(V10) w/o CL$_{\text{LINE}}$	0.0633	0.0603	0.0335
(V11) w/o CL$_{\text{similarity}}$	0.0619	0.0588	0.0320
(V12) w/o v	0.0603	0.0574	0.0310
(V13) w/o t	0.0550	0.0523	0.0287

(v) STARLINE (*resp.* (V2) LINE$_{\text{similarity}}$) shows better accuracy than (V10) w/o CL$_{\text{LINE}}$ (*resp.* (V11) w/o CL$_{\text{similarity}}$) for all metrics. In other words, CL utilizing both the refined modality graphs and the original ones is more effective in improving accuracy than using the original ones alone. This is speculated to be because, via CL, the original and refined modality graphs are used together to ensure that *crucial information* needed to capture the user's preferred features in the refined modality graphs is *retained/added*, while *noise* in the original ones is *reduced*.

(vi) STARLINE shows better accuracy than (V3) LINE$_{\text{w/o +}}$ *and/or* (V4) LINE$_{\text{w/o −}}$ for all metrics. Based on these results, we confirm that (1) despite increased data sparsity due to the effective dropping of original edges, interaction noise is reduced, which ultimately benefits in terms of accuracy, and (2) sparsity is mitigated by adding necessary edges, which positively impacts accuracy. As a result, the final STARLINE, which not only effectively drops original edges but also adds necessary ones, achieves the highest accuracy in multimedia recommendation.

(vii) (V9) LINE$_{\text{soft pruning}}$ shows lower accuracy than STARLINE for all metrics. This indicates that *excluding* uninfluential modality features of items in user–item interactions is more beneficial than assigning them lower weights, as interactions may arise from causes (ii) and (iii) of (Idea 1) in Section 1.

(viii) (V12) w/o v and (V13) w/o t show lower accuracies than STARLINE for all metrics. This is attributed to the fact that they cannot consider all modality features of the item that a user interacted with (*i.e.*, cause (i) of (Idea 1) in Section 1), and partially capture her potentially preferring modality features in her non-interactions (*i.e.*, case (ii) of (Idea 2) in Section 1).

(ix) (V13) w/o t shows lower accuracy than (V12) w/o v for all metrics. This result appears to be because, in Baby, the textual modality features of items have more influence on interactions than the

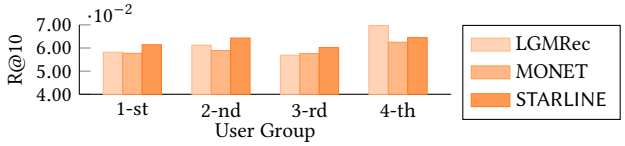


Figure 3: Accuracies of LGMRec, MONET, and STARLINE w.r.t. different user groups categorized by interaction sparsity.

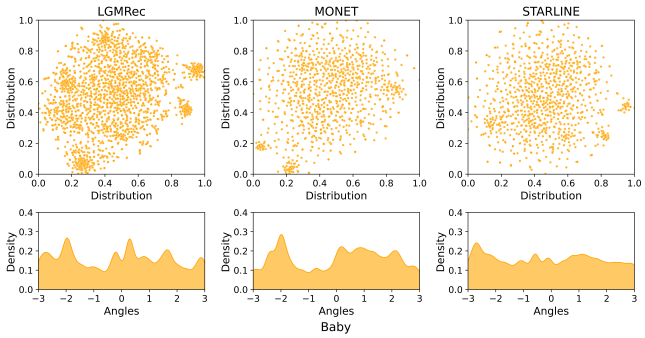


Figure 4: Distributions and density patterns of user embeddings from LGMRec, MONET, and STARLINE.

visual ones. Indeed, STARLINE refined (*i.e.*, dropped) more visual modality features than textual ones on interactions in Baby (*spec.*, 49,625 interactions for the visual modality and 42,321 interactions for the textual modality), as shown in Table 6.

RQ3: Effectiveness of STARLINE in Alleviating Data Sparsity. To verify that STARLINE effectively alleviates the data sparsity problem (*i.e.*, popularity bias) via graph augmentation, we compare it with its best competitors (CL-based) LGMRec and (non-CL-based) MONET within different user groups categorized by interaction sparsity [3, 26]. We first sorted the users in ascending order based on the number of their interactions. Next, we counted the number of users with the fewest interactions and used this number (say w) to partition all users (according to the number of interactions) into groups, each of which has w users (*i.e.*, equal-depth partitioning). As a result, on Baby, we obtained four user groups; specifically, each of the first three groups has about 6,100 users (*i.e.*, $w \approx 6{,}100$ in this case) and the last group has the remaining 1,100 users.

The accuracies of LGMRec, MONET, and STARLINE for each user group are shown in Figure 3. STARLINE outperforms LGMRec and MONET in the first three *sparse* groups, notably surpassing LGMRec by up to 5.34% in the sparsest first group. STARLINE performs worse than LGMRec in the densest group; however, the results in this group vary across datasets (see External Appendix). For instance, STARLINE is the most effective even in the densest group on Beauty. Note that in terms of R@10 on Baby, the accuracy of STARLINE on the densest group (0.0645) is slightly higher than its average accuracy across all user groups (0.0621); on the other hand, the accuracy of LGMRec on the densest group (0.0697) is significantly higher than its average accuracy across all user groups (0.0594) (see Table 3). In other words, while existing methods perform extremely well for dense groups but poorly for sparse groups, STARLINE effectively alleviates data sparsity by performing consistently well across both dense and sparse groups.

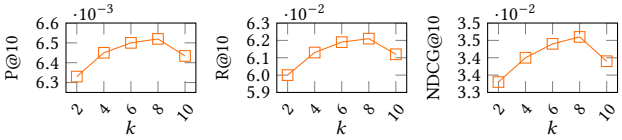


Figure 5: The effect of k on the accuracies of STARLINE.

Additionally, we analyzed STARLINE's effectiveness in achieving another goal of CL, *i.e.*, *feature alignment*. To this end, we again compare it with the two best competitors introduced earlier (*i.e.*, LGMRec and MONET), using user embeddings learned from each method. As in [33, 34], we randomly sample 1,000 users from each category, then reduce the dimensionality of the sampled user embeddings, and map them to a two-dimensional (2D) space via t-SNE [7]. To enable a more precise comparison, we plot the 2D feature distribution using Gaussian Kernel Density Estimation (KDE) [23]. The results of LGMRec, MONET, and STARLINE for the same users are shown in Figure 4. STARLINE shows more uniform feature distributions and more uniform patterns in KDE than LGMRec and MONET. This improved representation uniformity contributes to STARLINE's superior accuracy compared to LGMRec and MONET.

RQ4: Hyperparameter k Sensitivity. Recall that in (Step 1) of LINE, STARLINE partitions the range $[-1, 1]$ of cosine similarity $s^m_{u,j}$ between the user and item embeddings into k equal-width intervals and defines the non-interaction pairs in the k-th interval that has the highest $s^m_{u,j}$ as C_m. Figure 5 shows how the accuracy of STARLINE changes with different values of k. The accuracy of STARLINE increases steadily until $k = 8$ and then decreases. As the value of k increases from 2 to 8, C_m increasingly consists exclusively of non-interaction pairs likely to result in interactions. However, when the value of k is *too large* (*i.e.*, $k = 10$), C_m will include *only* such pairs of users and items that are *too similar* to those that the users interacted with. Unfortunately, users are unlikely to feel the need to interact with an item (*e.g.*, iPhone 16) too similar to an item (*e.g.*, iPhone 16 Pro) they have already purchased. Thus, providing *appropriate guidelines* based on *effective candidate pairs* filtered from the appropriate value of k is important for improving accuracy.

6 Conclusions

We proposed a novel CL-based multimedia recommendation framework, STARLINE, which effectively refines each modality graph. STARLINE is the *first* in multimedia recommendations to (1) *carefully* refine interactions, breaking away from the existing random strategy, and (2) include useful *non-interactions* in the refinement. Furthermore, STARLINE goes beyond the similarity-based strategies of state-of-the-art methods in other domains, conducting *more careful* and *flexible* refinement. Extensive experiments on five real-world datasets validated the effectiveness and validity of STARLINE.

Acknowledgments

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.RS-2022-00155586 and No.RS-2020-II201373).

References

[1] Xuheng Cai, Chao Huang, Lianghao Xia, and Xubin Ren. 2023. LightGCL: Simple Yet Effective Graph Contrastive Learning for Recommendation. In *The Eleventh International Conference on Learning Representations (ICLR)*.

[2] Dong-Kyu Chae, Jin-Soo Kang, Sang-Wook Kim, and Jaeho Choi. 2019. Rating augmentation with generative adversarial networks towards accurate collaborative filtering. In *Proceedings of the ACM on Web Conference 2019 (ACM WWW)*. 2616–2622.

[3] Zhiqiang Guo, Jianjun Li, Guohui Li, Chaoyang Wang, Si Shi, and Bin Ruan. 2024. LGMRec: Local and Global Graph Learning for Multimodal Recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. 8454–8462.

[4] Michael Gutmann and Aapo Hyvärinen. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. 297–304.

[5] Ruining He and Julian McAuley. 2016. VBPR: Visual Bayesian Personalized Ranking from Implicit Feedback. In *Proceedings of the AAAI conference on artificial intelligence (AAAI)*. 144–150.

[6] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval (ACM SIGIR)*. 639–648.

[7] G Hinton and L Van Der Maaten. 2008. Visualizing data using t-sne journal of machine learning research. *Journal of Machine Learning Research* 9 (2008), 2579–2605.

[8] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. In *Neural computation*. 1735–1780.

[9] Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical reparameterization with gumbel-softmax. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

[10] Wei Jiang, Xinyi Gao, Guandong Xu, Tong Chen, and Hongzhi Yin. 2024. Challenging Low Homophily in Social Recommendation. In *Proceedings of the ACM on Web Conference 2024 (ACM WWW)*. 3476–3484.

[11] Jiyeon Kim, Taeri Kim, and Sang-Wook Kim. 2023. Read between the interactions: Understanding non-interacted items for accurate multimedia recommendation. *Computer Science and Information Systems* 20, 3 (2023), 933–948.

[12] Taeri Kim, Yeon-Chang Lee, Kijung Shin, and Sang-Wook Kim. 2022. MARIO: Modality-Aware Attention and Modality-Preserving Decoders for Multimedia Recommendation. In *Proceedings of the 31st ACM International Conference on Information Knowledge Management (ACM CIKM)*. 993–1002.

[13] Yungi Kim, Taeri Kim, Won-Yong Shin, and Sang-Wook Kim. 2024. MONET: Modality-Embracing Graph Convolutional Network and Target-Aware Attention for Multimedia Recommendation. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining (ACM WSDM)*. 332–340.

[14] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Network. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

[15] Taeyong Kong, Taeri Kim, Jinsung Jeon, Jeongwhan Choi, Yeon-Chang Lee, Noseong Park, and Sang-Wook Kim. 2022. Linear, or Non-Linear, That is the Question!. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining (ACM WSDM)*. 517–525.

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS)*. 1106–1114.

[17] Hongjun Lim, Yeon-Chang Lee, Jin-Seo Lee, Sanggyu Han, Seunghyeon Kim, Yeongjong Jeong, Changbong Kim, Jaehun Kim, Sunghoon Han, Solbi Choi, et al. 2022. AiRS: A Large-Scale Recommender System at NAVER News. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. 3386–3398.

[18] Zhenghong Lin, Yanchao Tan, Yunfei Zhan, Weiming Liu, Fan Wang, Chaochao Chen, Shiping Wang, and Carl Yang. 2023. Contrastive Intra- and Inter-Modality Generation for Enhancing Incomplete Multimedia Recommendation. In *Proceedings of the 31st ACM International Conference on Multimedia (ACM MM)*. 6234–6242.

[19] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton van den Hengel. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval (ACM SIGIR)*. 43–52.

[20] Alec Radford, Luke Metz, and Soumith Chintala. 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. In *arXiv preprint arXiv:1511.06434*.

[21] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*. 452–461.

[22] Zhulin Tao, Xiaohao Liu, Yewei Xia, Xiang Wang, Lifang Yang, Xianglin Huang, and Tat-Seng Chua. 2022. Self-Supervised Learning for Multimedia Recommendation. In *IEEE Transactions on Multimedia*. 5107–5116.

[23] George R Terrell and David W Scott. 1992. Variable kernel density estimation. *The Annals of Statistics* (1992), 1236–1265.

[24] Changxin Tian, Yuexiang Xie, Yaliang Li, Nan Yang, and Wayne Xin Zhao. 2022. Learning to Denoise Unreliable Interactions for Graph Collaborative Filtering. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (ACM SIGIR)*. 122–132.

[25] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval (ACM SIGIR)*. 165–174.

[26] Wei Wei, Chao Huang, Lianghao Xia, and Chuxu Zhang. 2023. Multi-Modal Self-Supervised Learning for Recommendation. In *Proceedings of the ACM Web Conference 2023 (ACM WWW)*. 790–800.

[27] Yinwei Wei, Xiang Wang, Qi Li, Liqiang Nie, Yan Li, Xuanping Li, and Tat-Seng Chua. 2021. Contrastive learning for cold-start recommendation. In *Proceedings of the 29th ACM International Conference on Multimedia (ACM MM)*. 5382–5390.

[28] Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, and Tat-Seng Chua. 2020. Graph-Refined Convolutional Network for Multimedia Recommendation with Implicit Feedback. In *Proceedings of the 28th ACM international conference on multimedia (ACM MM)*. 3541–3549.

[29] Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. 2021. Self-supervised Graph Learning for Recommendation. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval (ACM SIGIR)*. 726–735.

[30] Wei Yang, Zhengru Fang, Tianle Zhang, Shiguang Wu, and Chi Lu. 2023. Modal-aware Bias Constrained Contrastive Learning for Multimodal Recommendation. In *Proceedings of the 31st ACM International Conference on Multimedia (ACM MM)*. 6369–6378.

[31] Yonghui Yang, Le Wu, Richang Hong, Kun Zhang, and Meng Wang. 2021. Enhanced Graph Learning for Collaborative Filtering via Mutual Information Maximization. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (ACM SIGIR)*. 71–80.

[32] Zixuan Yi, Xi Wang, Iadh Ounis, and Craig Macdonald. 2022. Multi-modal Graph Contrastive Learning for Micro-video Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (ACM SIGIR)*. 1807–1811.

[33] Junliang Yu, Hongzhi Yin, Xin Xia, Tong Chen, Lizhen Cui, and Quoc Viet Hung Nguyen. 2022. Are Graph Augmentations Necessary? Simple Graph Contrastive Learning for Recommendation. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval (ACM SIGIR)*. 1294–1303.

[34] Penghang Yu, Zhiyi Tan, Guanming Lu, and Bing-Kun Bao. 2023. Multi-View Graph Convolutional Network for Multimedia Recommendation. In *Proceedings of the 31st ACM International Conference on Multimedia (ACM MM)*. 6576–6585.

[35] Jinghao Zhang, Yanqiao Zhu, Qiang Liu, Shu Wu, Shuhui Wang, and Liang Wang. 2021. Mining Latent Structures for Multimedia Recommendation. In *Proceedings of the 29th ACM international conference on multimedia (ACM MM)*. 3872–3880.

[36] Jinghao Zhang, Yanqiao Zhu, Qiang Liu, Mengqi Zhang, Shu Wu, and Liang Wang. 2022. Latent Structure Mining with Contrastive Modality Fusion for Multimedia Recommendation. In *IEEE Transactions on Knowledge and Data Engineering (IEEE TKDE)*. 9154–9167.

[37] Xin Zhou and Zhiqi Shen. 2023. A Tale of Two Graphs: Freezing and Denoising Graph Structures for Multimodal Recommendation. In *Proceedings of the 31st ACM International Conference on Multimedia (ACM MM)*. 935–943.

Appendix A Analysis of Refinements

We investigate the *distribution* of the refined user–item interactions (*resp.* candidate non-interaction pairs in C_m) in each modality graph $\mathcal{G}$ via LINE, and the *characteristics* of modality features for the items in these interactions and non-interactions.

Distribution. Tables 6 and 7 show the number of dropped edges among all original edges in $\mathcal{G}$, and the number of added edges among all candidate pairs in C_m, respectively, for each modality (visual v and textual t) and each category. In Table 6, 'All_D' denotes the edges dropped for *all* modalities. In Table 7, 'All_C' denotes the user–item non-interactions selected as candidate pairs in *all* modalities (*i.e.*, $C_v \cap C_t$), and 'All_A' denotes the added edges for *all* modalities among the candidate pairs in 'All_C'.

In Table 6, the distribution of refined original edges varies per modality and category. We speculate that this result occurred because a *purchaser* and an *actual user* of an item may be different.

Table 6: The distribution of original edges and refined (*i.e.*, dropped) original edges in each modality graph for each category

Datasets		Baby	Beauty	Toys & Games	Women Clothing	Men Clothing
# Original Edges		118,551	147,320	123,612	95,629	22,296
# & Ratio (%) of Refined (*i.e.*, Dropped) Edges	v	49,625 (41.86)	22,769 (15.46)	18,122 (14.66)	27,064 (28.30)	10,582 (47.46)
	t	42,321 (35.70)	23,700 (16.09)	15,463 (12.51)	25,268 (26.42)	10,616 (47.61)
	$\mathbf{All_D}$	17,774 (14.99)	3,767 (2.56)	2,252 (1.82)	7,119 (7.44)	5,033 (22.57)

Table 7: The distribution of candidate pairs C_m and refined (*i.e.*, added) candidate pairs in each modality graph for each category. Ratio (%) denotes the proportion of refined edges among all candidate pairs in C_m.

Datasets		Baby	Beauty	Toys & Games	Women Clothing	Men Clothing
# Candidate Pairs C_m	v	2,201	2,267	280	487	23
	t	1,916	7,276	2,979	3,328	47
	$\mathbf{All_C}$	7	85	3	134	3
# & Ratio (%) of Refined (*i.e.*, Added) Edges	v	1,768 (80.33)	1,626 (71.72)	222 (79.29)	335 (68.79)	19 (82.61)
	t	1,337 (69.78)	5,848 (80.37)	2,528 (84.86)	2,658 (79.87)	41 (87.23)
	$\mathbf{All_A}$	6 (85.71)	64 (75.29)	1 (33.33)	57 (42.54)	2 (66.33)

For instance, items in Baby and Toys & Games are likely to be purchased by others (*e.g.*, parents) rather than the actual users (*i.e.*, babies and children) due to their low age range. Therefore, the specs (*i.e.*, textual t modality) of these items, such as materials and ingredients, are likely to have more influence on purchasers than their appearance (*i.e.*, visual v modality). As a result, more edges might have been dropped in the visual modality graph than in the text modality graph. On the other hand, items in Beauty are likely to be directly purchased by actual users (*i.e.*, typically adults). For items such as color cosmetics, their appearance is likely to convey their information more clearly than their specs, thereby having more influence on purchasers than the specs. Therefore, more edges might have been dropped in the textual modality graph than in the visual modality graph.

The ratio of dropped edges is most pronounced for Men Clothing, followed by Baby in every modality and '$\mathrm{All_D}$'. We focus on Baby as a representative case, since its purchasing behavior offers a clearer explanation. In Baby, purchasers (*e.g.*, parents) tend to buy items based on the needs of the users, *i.e.*, rapidly growing babies, *rather than their own preferences*. Therefore, many interactions might have been dropped as STARLINE judged that the multimodal features of the items in these interactions would not have influenced the purchasers. Indeed, we confirmed that (i) the average interaction cycle of users (in this context, purchasers) in Baby is faster than those in other categories (*e.g.*, 13.48% faster compared to Beauty and 29.45% faster compared to Toys & Games); (ii) the average cosine similarity between (raw) modality features, pre-obtained via deep-learning models *regardless of* user–item interactions, of items each user interacted with is lower in Baby than in other categories; for instance, the similarity between items that each user interacted with is 8.48% (*resp.* 4.76%) higher for the visual (*resp.* textual) modality than the similarity between all items in that modality on Baby, while being 11.15% (*resp.* 11.38%) higher on Beauty and 19.07% (*resp.* 13.44%) higher on Toys & Games.

Table 7 shows the distribution of refined candidate pairs per modality and category. Recall that every candidate non-interaction pair (u, j) in C_m is *likely to result in an interaction*, where a user u's preferred feature in modality m and the modality m's feature of item j are very similar. Because of this, the refinement ratio is significantly high compared to interactions. However, it is notable that at least 12% of candidate pairs are not considered as additional edges by STARLINE. This further supports the claim that a similarity-based strategy *alone* is insufficient for accurately refining non-interactions.

Characteristics. Given a user and multiple items she has interacted with, STARLINE refines her interactions in terms of each modality. Intuitively, the items retained after refinement are likely to have features aligned with her preferred feature in each modality; therefore, we can infer that the features of these items are likely to be similar to each other. Conversely, the items dropped after refinement are likely to have features quite different from her preferred feature in each modality; therefore, we can infer that despite these items being the ones she has interacted with, their features are unlikely to be similar to those of the items retained after refinement. Our hypotheses would also apply to the added or excluded candidate non-interactions in C_m in terms of each modality.

To confirm whether our hypotheses hold in reality, we analyze the characteristics of the refined items by using their (raw) visual and textual modality features obtained *regardless of* user–item interactions as in [13]. First, for *each user*, we calculate the following similarities:

- **R–R** (*resp.* **A–A**): The average cosine similarity between the modality features of items *retained* (*resp. added*) after refinement among the items that she interacted with (*resp.* items included in her candidate pairs C_m).
- **R–D** (*resp.* **A–E**): The average cosine similarity between the modality features of items *retained* (*resp. added*) after refinement and those of items *dropped* (*resp. excluded*) among the items that she interacted with (*resp.* items included in her candidate pairs in C_m).
- **I–I**: The average cosine similarity between the modality features of items that she *interacted with*.
- **I–N**: The average cosine similarity between the modality features of items that she *interacted with* and those of items that she did *not* interact with.

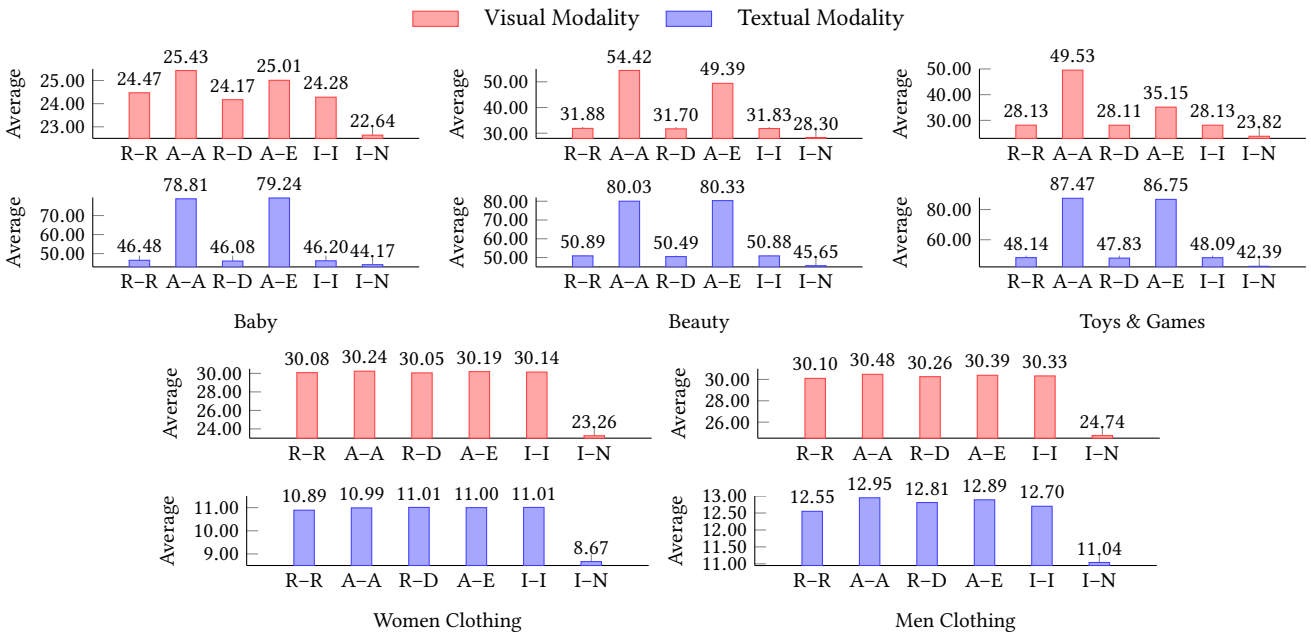


Figure 6: The average similarities between modality features of items retained (R–R) (*resp.* added (A–A)) after refinement; between modality features of items retained (*resp.* added) after refinement and those of items dropped (R–D) (*resp.* excluded (A–E)); between modality features of interacted items (I–I); between modality features of interacted items and those of non-interacted items (I–N) for each modality. All average similarities are adjusted by multiplying by 100 for ease of comparison.

Then, we *averaged* the R–R, A–A, R–D, A–E, I–I, and I–N across *all* users, which are provided per modality and category in Figure 6.

As shown in Figure 6, in terms of interaction refinements, the similarity is high in the order of R–R, I–I, R–D, and I–N in most modalities and categories; this overall pattern supports our hypotheses. Additionally, these results suggest that the user's preferred features can be captured from the modality features of the items that she interacted with (compare I–I with I–N) [13]; STARLINE can capture these preferred features *in greater detail* and *more accurately* (compare R–R with I–I and I–I with R–D), validated by STARLINE's highest accuracy.

On the other hand, in terms of candidate non-interaction refinements, the similarity is high in the order of A–A, A–E, I–I, and I–N in visual modality on all categories and A–E, A–A, I–I, and I–N (*resp.* A–A, A–E, I–I, and I–N) in textual modality on Baby and Beauty (*resp.* Toys & Games and Men Clothing). Contrary to our hypotheses, these results differ from the tendencies observed for interaction refinements. It is speculated that, because the items included in the candidate pairs have their embeddings similar to the user embeddings, which represent the users' preferred features, not only A–A but also A–E show higher similarity than I–I (even though these similarities are based on (raw) modality features *regardless of* user–item interactions). We highlight that STARLINE achieves the highest accuracy by *excluding* items from candidate pairs *if necessary* (see Table 7), despite showing that A–E is the highest in the textual modality on Baby and Beauty.

Appendix B Effect of Injected Noisy Interactions

To assess both the *refinement capability* and *robustness* of STARLINE, we added the same set of random noisy interactions (amounting to 10%, 30%, 50%, and 100% of the number of original interactions) to all modality graphs; we examined (1) how effectively STARLINE could filter them out, and (2) how STARLINE's accuracy compares to those of the two best competitors, LGMRec and MONET, under the same noisy conditions. As a result, STARLINE demonstrates the ability to remove noise, and its effectiveness increases as the noise level grows. Although not all noisy interactions are removed (likely because STARLINE does not directly filter noise but instead mitigates it indirectly through contrastive views) it still filters out a significant portion. Furthermore, under increasingly noisy graph conditions, STARLINE consistently outperforms LGMRec and MONET, exhibiting smaller performance degradation compared to these baselines as the noise level increases. These results highlight STARLINE's robustness to noisy interactions as well as its refinement ability. Unfortunately, due to space limitations, the results for this experiment are provided in the External Appendix.

Appendix C Supplementary Materials

The External Appendix also includes supplementary materials such as the pseudocode of STARLINE, additional ablation variants, and a case study.