Optimizing Recall or Relevance? A Multi-Task Multi-Head Approach for Item-to-Item Retrieval in Recommendation

Jiang Zhang
Meta Platforms, Inc.
Menlo Park, CA, USA
jiangzhang2024@meta.com

Sumit Kumar
Meta Platforms, Inc.
Seattle, WA, USA
sumitkumar@meta.com

Wei Chang
Meta Platforms, Inc.
Menlo Park, CA, USA
mrweichang@meta.com

Yubo Wang
Meta Platforms, Inc.
Bellevue, WA, USA
yubowang@meta.com

Feng Zhang
Meta Platforms, Inc.
Austin, TX, USA
fengzhang1@meta.com

Weize Mao
Meta Platforms, Inc.
Menlo Park, CA, USA
wzmao@meta.com

Hanchao Yu
Meta Platforms, Inc.
Menlo Park, CA, USA
yhcece@gmail.com

Aashu Singh
Meta Platforms, Inc.
Menlo Park, CA, USA
aashusingh@meta.com

Min Li
Meta Platforms, Inc.
Menlo Park, CA, USA
minli@meta.com

Qifan Wang
Meta Platforms, Inc.
Menlo Park, CA, USA
wqfcr@fb.com

Abstract

The task of item-to-item (I2I) retrieval is to identify a set of relevant and highly engaging items based on a given item. I2I retrieval is a crucial component in modern recommendation systems, where users' previously engaged items serve as trigger items to retrieve relevant content for future engagement. However, existing I2I models in industry are primarily built on co-engagement data and optimized using the recall measure, which overly emphasizes co-engagement patterns while failing to capture semantic relevance. This often leads to overfitting short-term co-engagement trends at the expense of long-term benefits such as discovering novel interests and promoting content diversity. To address this challenge, we propose **MTMH**, a **M**ulti-**T**ask and **M**ulti-**H**ead I2I retrieval model that achieves both high recall and semantic relevance. Our model consists of two key components: 1) a multi-task learning loss for formally optimizing the trade-off between recall and relevance, and 2) a multi-head I2I retrieval architecture for retrieving both highly co-engaged and semantically relevant items. We evaluate MTMH using proprietary data from a commercial platform serving billions of users and demonstrate that it can improve recall by up to 14.4% and semantic relevance by up to 56.6% compared with prior state-of-the-art models. We also conduct live experiments to verify that MTMH can enhance both short-term consumption metrics and long-term user-experience-related metrics. Our work provides a principled approach for jointly optimizing I2I recall and semantic relevance, which has significant implications for improving the overall performance of recommendation systems.

CCS Concepts

• **Information systems → Retrieval models and ranking**.

Keywords

Semantic Relevance, Recommendation, Multi-head multi-task Learning, Item-to-item Retrieval

ACM Reference Format:
Jiang Zhang, Sumit Kumar, Wei Chang, Yubo Wang, Feng Zhang, Weize Mao, Hanchao Yu, Aashu Singh, Min Li, and Qifan Wang. 2025. Optimizing Recall or Relevance? A Multi-Task Multi-Head Approach for Item-to-Item Retrieval in 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, 11 pages. https://doi.org/10.1145/3711896.3737255

1 Introduction

Item-to-item (I2I) retrieval refers to the task of retrieving relevant and highly engaged items for a given item (also named trigger item), which is an important component in modern recommendation systems. In I2I retrieval, items from users' past engagement history are used as trigger items to retrieve new items for future engagement, providing users with personalized experience [9, 41, 58].

Most existing industry I2I models utilize co-engagement data to guide their training, operating under the assumption that items users engage with in quick succession are likely to be relevant. These I2I models often place excessive emphasis on co-engagement

KDD '25, Toronto, ON, Canada.

ACM ISBN 979-8-4007-1454-2/25/08
https://doi.org/10.1145/3711896.3737255

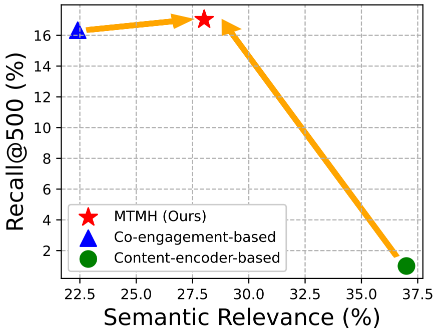


Figure 1: Recall vs. Relevance for I2I retrieval models. The co-engagement based model (blue triangle) refers to an I2I model trained exclusively on co-engagement data, while the content-encoder based model (green circle) represents an I2I model that directly utilizes embeddings generated by a pre-trained item content encoder.

patterns during retrieval because they are primarily trained to optimize recall metrics based on this data, without explicitly optimizing for semantic relevance [3, 12, 41, 58, 62]. This focus can lead to I2I retrieval models overfitting to short-term co-engagement data, potentially compromising long-term objectives such as user retention, content diversity, and the discovery of new interests [49].

Enhancing the semantic relevance of I2I models is crucial for improving the overall performance of recommendation systems and offers several significant benefits. First, it can enhance the recall of user interests in retrieved items, thereby providing a more personalized user experience [34]. Second, it facilitates the effective surfacing of fresh contents, particularly for new contents that lacks sufficient user engagement data [14]. Lastly, it helps mitigate short-term co-engagement bias, fostering a healthier and more valuable feedback loop that contributes to better long-term outcomes in recommendation systems [18]. However, optimizing I2I semantic relevance is challenging, primarily because quantifying the semantic relevance between two items is difficult, especially for multi-modal content such as short videos. This challenge is compounded by the lack of explicit supervision labels for optimizing semantic relevance. Fundamentally, there is a trade-off between recall and semantic relevance in I2I retrieval: *semantically relevant items may not always exhibit high co-engagement rates.*

To illustrate this trade-off, we compare the recall and relevance of two I2I retrieval models: (1) a co-engagement based model trained solely on co-engagement data, and (2) a content-encoder based model that utilizes a large pre-trained content encoder to generate item embeddings. As depicted in Figure 1, the co-engagement based model achieves high recall but exhibits poor semantic relevance. In contrast, the content-encoder based model is adept at retrieving items with high semantic relevance, yet its recall is less than 1%, significantly lower than that of the co-engagement based model. This trade-off underscores the challenge of balancing co-engagement rates and semantic relevance in I2I retrieval models. More experimental details are provided in Section 2.

To address the aforementioned challenges, we propose **MTMH**, a **M**ulti-**T**ask and **M**ulti-**H**ead I2I retrieval model that achieves an optimal balance between recall and semantic relevance, as illustrated in Figure 1. The design of MTMH incorporates two key components: (1) a multi-task learning loss for jointly optimizing recall and semantic relevance, and (2) a multi-head I2I model architecture for retrieving items that are both highly co-engaged and semantically relevant. Specifically, the co-engagement loss is crafted to maximize the co-engagement rate (or recall), while the semantic relevance loss is designed to preserve the semantic similarity between items by distilling item semantic knowledge from a pre-trained content encoder into the learned item embeddings. The multi-task loss is computed as a weighted sum of the engagement loss and relevance loss, providing a principled approach to jointly optimize co-engagement efficiency and semantic relevance during training. Moreover, we design a multi-head I2I retrieval model that includes an engagement head and a relevance head. The engagement head is trained solely on engagement loss to select highly co-engaged items, while the relevance head is trained using the multi-task learning loss to select items that are both highly relevant and co-engaged. By merging retrieved items from both heads, MTMH achieves improved recall and semantic relevance simultaneously during serving time.

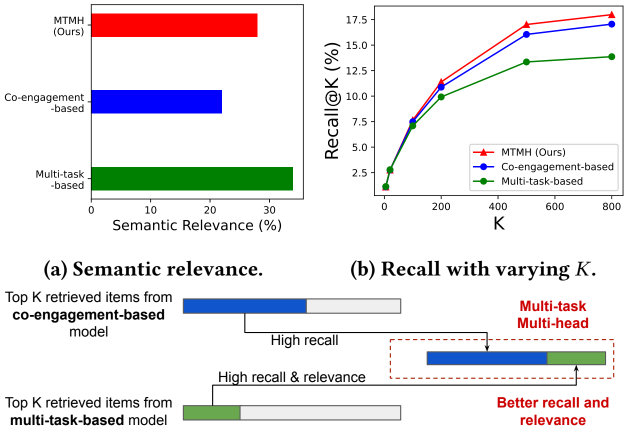
We evaluate the performance of MTMH using proprietary data from a commercial platform serving billions of users. Our results show that MTMH can improve I2I retrieval recall by up to 14.4% and semantic relevance by up to 56.6%, achieving the best trade-off between these two metrics compared with all baselines (see Section 4.2). To further validate the effectiveness of MTMH, we deploy MTMH on this commercial platform and conduct online A/B testing. Our online evaluation results demonstrate that MTMH not only successfully improves consumption metrics such as daily active users and time spent, but also significantly enhances user-experience-related metrics such as user interest recall, novel interest discovery rate, content diversity and freshness (see Section 4.6). We summarize our key contributions as follows:

- We systematically examine the fundamental trade-off between recall and semantic relevance in I2I retrieval, uncovering their interconnections and highlighting the challenges of balancing and optimizing these two metrics.
- We propose MTMH, a **M**ulti-**T**ask and **M**ulti-**H**ead I2I retrieval model, which provides a principled approach for jointly optimizing the trade-off between I2I co-engagement rate and semantic relevance.
- We evaluate MTMH on proprietary data from a commercial platform serving billions of users and demonstrate that it can improve recall by up to 14.4% and semantic relevance by up to 56.6% compared with prior SOTAs.
- We integrate MTMH into production to verify that MTMH can increase both topline consumption metrics and long-term user-experience-related metrics.

2 Preliminary Study

In this section, we conduct a preliminary study to investigate the trade-off between recall and semantic relevance in I2I retrieval, providing insights and motivating the MTMH's design.

Experimental setup. We train the aforementioned I2I retrieval models using user data from period T_1 and evaluate their performance using data from period T_2 after T_1. During evaluation, we first



(a) Semantic relevance.

(b) Recall with varying K.

(c) How to achieve both better recall and relevance.

Figure 2: Recall vs relevance for I2I retrieval models. The co-engagement based model (blue line) is trained via co-engagement loss only, while the multi-task-based model (green line) is trained on multi-task learning loss (co-engagement loss + relevance loss).

sample a set of users, and for each user we select 50 engaged items from their past user interaction history (UIH) as trigger items. Next, for each trigger item, we conduct Approximate Nearest Neighbor (ANN) search based on embedding cosine similarity to identify the top 2000 nearest candidate items and employ a preranker model to further select top 30 out of 2000 candidates for each trigger item. Finally, for each user, we sort candidate items retrieved by 50 trigger items based on their ranking scores and select top 500 candidate items to measure recall and semantic relevance. Note that the recall@500 is computed as the percentage of ground-truth engaged items from future UIH hit by these 500 retrieved items. Semantic relevance is measured by the average topic category (from human labels) match rates between each trigger item and each candidate item retrieved by this trigger.

Next, we present our key findings from this preliminary study. We start with the following question:

Q1: Can the embeddings generated by a large pre-trained content encoder be directly used for I2I retrieval? To answer this question, we compare the recall and semantic relevance performance of the following two baseline models: *1) Co-engagement based model*: This model is trained on co-engagement data to learn item embeddings. The training objective is to maximize the I2I co-engagement rate. *2) Content-encoder based model*: This model leverages a large pre-trained content encoder with superior content understanding capability to generate content embeddings for items, and directly uses them for I2I retrieval.

Results for Q1. Figure 1 presents the evaluation results of co-engagement based and content-encoder based models. We observe that the co-engagement based model achieves 22.4% semantic relevance, while the content-encoder based model achieves 37.5%. This indicates that item embeddings generated by the pre-trained content encoder exhibit significantly better semantic relevance compared to the co-engagement based model. However, the recall of the content-encoder based model is less than 1%, while the co-engagement based model achieves 16.3%. This stark contrast highlights a strong disconnection between semantic relevance and co-engagement rate: *high semantic relevance does not necessarily translate into a high recall in I2I retrieval.* Motivated by this, we introduce a multi-task learning loss that jointly optimizes co-engagement rate and semantic relevance.

Q2: Is it possible to simultaneously enhance recall and semantic relevance of I2I retrieval model? Although multi-task learning provides a structured approach to optimizing the trade-off between recall and semantic relevance, it does not fully resolve the fundamental challenge of balancing these two objectives: *maximizing recall and maximizing semantic relevance cannot be achieved simultaneously in multi-objective optimization.* We hypothesize that items co-engaged by a user are not always semantically relevant, meaning that retrieving only high semantic relevance items may overlook highly engaged but less relevant items (e.g., those popular items). To test this, we compare the recall and semantic relevance of a model trained via multi-task learning (denoted as the multi-task based model) with a co-engagement based model. The formal definition of the multi-task learning is provided in Section 3.2.

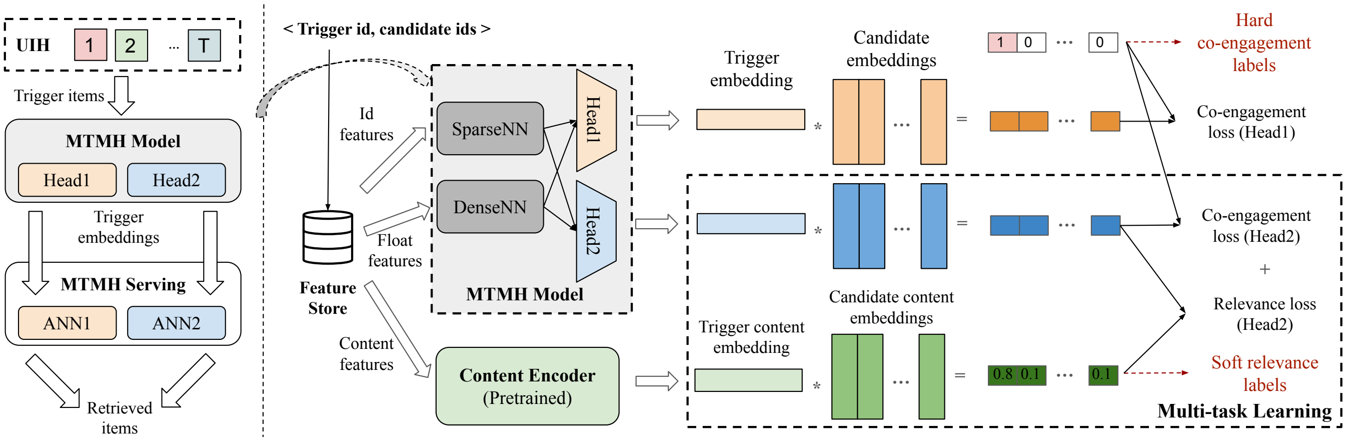
Results for Q2. As shown in Figure 2a, the multi-task based model achieves over a 50% increase in semantic relevance compared to the co-engagement based model, which is expected due to its multi-task relevance modeling. In contrast, as illustrated in Figure 2b, the recall of the multi-task based model is lower than that of the co-engagement-based model. Specifically, when K is smaller than 200, the multi-task based model achieves a recall comparable to that of the co-engagement based model, suggesting that *highly relevant candidates can also exhibit high engagement efficiency*. However, as K increases, the recall of the multi-task based model begins to plateau much earlier than that of the co-engagement based model. This finding indicates that *retrieving more semantically relevant items does not necessarily increase the overall recall.*

Motivated by these findings, we design a multi-task multi-head retrieval architecture. In this setup, one head is trained to retrieve highly engaged items by minimizing only the co-engagement loss, while the other head is trained to retrieve highly relevant items by minimizing the multi-task learning loss (see Section 3.5 for details). By merging candidates retrieved from both heads during serving (as illustrated in Figure 2c), MTMH is able to retrieve items that are extremely high in co-engagement (albeit with less semantic relevance) from the first head, and items that are highly semantically relevant from the second head, leading to both higher recall and enhanced semantic relevance.

3 Methodology

3.1 Overview of MTMH Approach

We first provide an overview of our MTMH model in figure 3. The goal of MTMH is to learn the item embeddings and use them to retrieve highly engaged and relevant items based on past user engagements. MTMH essentially consists of three key components: 1) a multi-task learning module for jointly optimizing recall and semantic relevance (Section 3.2); 2) a multi-head design for retrieving both highly co-engaged and semantically relevant items (Section 3.3) and 3) a pre-trained content encoder used to distill multi-modal content knowledge into the learned item embeddings (Section 3.4);



(a) MTMH workflow.

(b) MTMH model and training loss.

Figure 3: Overview of the proposed MTMH approach. 1) The multi-task learning consists of a co-engagement loss and a relevance loss, where the co-engagement loss has the format of InfoNCE [35], with the hard labels from the co-engagement data and the relevance loss is based on knowledge distillation from the pre-trained content encoder. 2) The multi-head design includes an engagement head (w/ only co-engagement loss) and a relevance head (w/ multi-task loss). 3) The pre-trained content encoder is used to generate relevance supervision for transferring the content semantic knowledge to the learned embeddings.

To train the MTMH model, we first collect a set of positive and negative <*trigger, candidate*> item pairs. Specifically, we construct positive pairs from user interaction history (UIH). Given a UIH containing T past engaged items in chronological order, we pair the T-th item with each of the previous T-1 items, forming T-1 positive pairs. To further enhance the relevance of these positive pairs, we leverage the Search-based Interest Model (SIM [36, 59]) to identify semantically relevant items within the user's UIH and assign higher weights. Additionally, we construct negative item pairs by randomly sampling items that the user has not engaged with and pairing them with the engaged items in each UIH.

3.2 Multi-task Learning

In this subsection, we present the multi-task learning module which optimizes both recall and semantic relevance in I2I retrieval model. At a high level, our multi-task learning loss is computed as the weighted sum of a co-engagement loss and a semantic relevance loss, as shown in Figure 3b. The co-engagement loss is designed to maximize the embedding similarity between positive item pairs (i.e. co-engaged items) while minimize the embedding similarity between negative item pairs. The semantic relevance loss is designed to minimize the contrastive semantic information loss between content embeddings generated by the content encoder and item embeddings learned by I2I retrieval model.

Co-engagement loss. We employ the widely used InfoNCE loss [35] as the co-engagement loss, which is formally defined as:

$$L_e = -\sum_{i=1}^{M}\sum_{j=1}^{N}\log p^+_{i,j},$$
$$p^+_{i,j} = \frac{e^{<E_i,E^+_{i,j}>}}{e^{<E_i,E^+_{i,j}>}+\sum_{k=1}^{k=L} e^{<E_i,E^-_{i,k}>}} \qquad (1)$$

where $p^+_{i,j}$ is the predicted probability for trigger item i to identify positive candidate item j from a set of items based on item embedding similarity. E_i is the i-th trigger item embedding. $E^+_{i,j}$ is the j-th item embedding which is positively paired with trigger item i, and $E^-_{i,k}$ is the k-th item embedding which is negatively paired with trigger item i. M is the total number of trigger items, whereas N is the number of candidate items which are positively paired with each trigger item. L is the number of items which are negatively paired with each trigger item, and $< x, y >$ represents the dot product between embedding x and y. By minimizing the co-engagement loss, the embeddings of positive item pairs are pulled closer, while the embeddings of negative item pairs are pushed apart.

Semantic relevance loss. Co-engagement modeling does not guarantee the preservation of semantic relevance in the learned item embeddings, often resulting in suboptimal item relevance. To address this, we introduce a semantic relevance loss that aligns the contrastive similarity between item embeddings generated by the I2I retrieval model with the contrastive similarity between item content embeddings produced by the content encoder. In this way, the semantic content knowledge is distilled into the learned embeddings. We use the content embeddings generated by the pre-trained content encoder to build soft semantic relevance labels between <trigger, candidate> pairs, and then use these soft relevance labels to guide the semantic relevance optimization (see Figure 3b for details). This process is also known as knowledge distillation [15]. Formally, the relevance loss is defined as:

$$L_r = \sum_{i=1}^{M}\sum_{j=1}^{N} D_{KL}(Q_{i,j}||P_{i,j}) = \sum_{i=1}^{M}\sum_{j=1}^{N}\left(q^+_{i,j}\log\frac{q^+_{i,j}}{p^+_{i,j}} + \sum_{k=1}^{L} q^-_{i,k}\log\frac{q^-_{i,k}}{p^-_{i,k}}\right) \qquad (2)$$

where:

$$q^+_{i,j} = \frac{e^{<F_i,F^+_{i,j}>}}{e^{<F_i,F^+_{i,j}>}+\sum_{k=1}^{k=L} e^{<F_i,F^-_{i,k}>}},\ q^-_{i,k} = \frac{e^{<F_i,F^-_{i,k}>}}{e^{<F_i,F^+_{i,j}>}+\sum_{k=1}^{k=L} e^{<F_i,F^-_{i,k}>}}$$

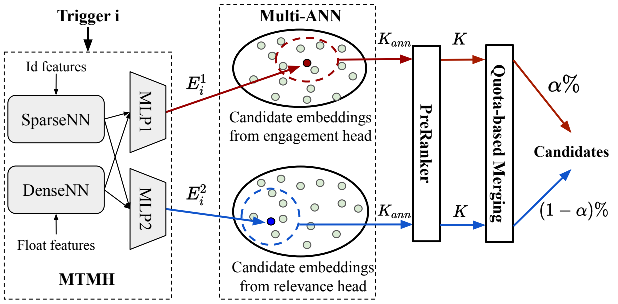


Figure 4: MTMH serving pipeline contains three modules: 1) a multi-ANN module to retrieve K_{ann} nearest candidates from each head; 2) a preranker module to select top K out of K_{ann} candidates from each head; 3) a quota-based merging module to merge candidates from the two heads.

$$p^-_{i,k} = \frac{e^{<E_i,E^-_{i,k}>}}{e^{<E_i,E^+_{i,j}>} + \sum_{k=1}^{k=L} e^{<E_i,E^-_{i,k}>}}$$

$Q_{i,j} = [q^+_{i,j}, q^-_{i,1}, ..., q^-_{i,L}]$, and $P_{i,j} = [p^+_{i,j}, p^-_{i,1}, ..., p^-_{i,L}]$. $D_{KL}(Q_{i,j}||P_{i,j})$ denotes the KL divergence between probability distributions $Q_{i,j}$ and $P_{i,j}$. F_i is the i-th trigger item content embedding, $F^+_{i,j}$ is the j-th item content embedding positively paired with trigger i, $F^-_{i,k}$ is the k-th item content embedding negatively paired with trigger i.

Note that $q^+_{i,j}$ and $q^-_{i,k}$ can be interpreted as the target probability for trigger item i to identify positively paired item j and negatively paired item k based on item content embedding similarity respectively. $p^+_{i,j}$ (defined in Eq. (1)) and $p^-_{i,k}$ can be interpreted as the predicted probability for trigger item i to identify positively paired item j and negatively paired item k based on item embedding similarity. By minimizing the KL divergence between $Q_{i,j}$ and $P_{i,j}$, the relative similarities between learned item embeddings are aligned with those of the item content embeddings generated by the content encoder. Hence, item embeddings generated by the I2I retrieval model can effectively preserve item semantic relevance.

Multi-task loss. The multi-task learning loss is defined as:

$$L_{mt} = L_e + w_r * L_r \tag{3}$$

where w_r is an hyper-parameter to control the weight of relevance loss. Increasing w_r is expected to improve the semantic relevance of the model but might decrease the co-engagement rate (or recall) at the same time (see Section 4.4 for details). Note that the default value of w_r is 0.5, unless otherwise specified.

3.3 Multi-head Architecture

As discussed in Section 2, although the I2I retrieval model trained via multi-task learning can retrieve items with high semantic relevance, it may miss some highly engaged but less relevant items. To overcome this challenge, we design a multi-head model with an engagement head and a relevance head. Specifically, the engagement head is trained to minimize co-engagement loss L_e (Eq. 1) only, focusing on retrieving highly co-engaged items. In contrast, the relevance head is trained to minimize the multi-task loss L_{mt} (Eq. 3) to retrieve items with high semantic relevance.

As demonstrated in Figure 4, at the bottom of the multi-head model, a Sparse Neural Network (SparseNN) is used to map each id feature of the input item into a unique embedding vector, and a Dense Neural Network (DenseNN) is used to map the float features of the input item into a latent vector. These vectors are then concatenated and fed into two MLP heads, each of which has the same input vectors but uses a dedicated MLP to map them into a different item embedding space. Notably, increasing the number of heads only slightly increases the total number of model parameters, since the majority of parameters come from SparseNN.

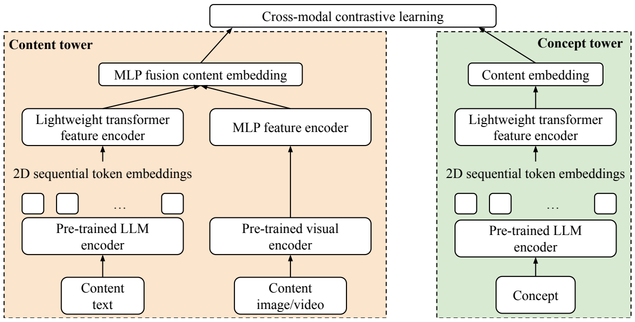

Cross-modal contrastive learning
Content tower
Concept tower
MLP fusion content embedding
Lightweight transformer feature encoder
MLP feature encoder
2D sequential token embeddings
...
Pre-trained LLM encoder
Pre-trained visual encoder
Content text
Content image/video
Content embedding
Lightweight transformer feature encoder
2D sequential token embeddings
...
Pre-trained LLM encoder
Concept

Figure 5: Model architecture of the content encoder. The content tower processes content inputs to generate multimodal embeddings, while the concept tower extracts semantic embeddings from text concepts. Contrastive loss optimizes relevance between content and concept representations.

3.4 Content Encoder

To effectively capture the item semantic representation, inspired by recent works LLM2VEC [1] and VLM2VEC [19], we propose a multimodal contrastive learning framework to learn the correspondence between content and concepts. Figure 5 shows the model architecture. The content tower is composed of a pre-trained LLM text encoder [44] and visual encoder [55] that extract multimodal representations from the content. These representations are then aggregated through a Multi-layer Perceptron (MLP) module to obtain a single embedding. Similarly, the concept tower generates concept embeddings from a pre-trained LLM encoder. Contrastive loss between the content and concept embeddings is used to train this model. The pairs of content and concepts are sourced from multiple places, including user-added hashtags, user search queries, and multimodal LLM-generated tags. The relevance knowledge in this content encoder is then distilled to the learned MTMH model. Note that we use VIT-H14-632M as the visual encoder model, and XLM-R-Large-550M as the text encoder model. Both the visual and text encoders have output dimension of 1024, and the final output content embedding output by fusion MLP has dimension 128 (see Table 3 for details).

3.5 MTMH Serving Strategy

Building on the top of MTMH architecture, we now describe the serving strategy of MTMH. As illustrated in Figure 4, the serving pipeline of MTMH consists of three key modules: 1) a multi-ANN module to conduct ANN search and retrieve top K_{ann} nearest candidates based on embeddings generated by each head in parallel; 2) a per-head preranker module to rank the candidates retrieved from each head and preserve top K candidates for each head; and 3) a quota-based candidate merging module which merges top K candidates from each head based on their percentage quota.

Multi-ANN module. For candidate item embeddings generated by each head of MTMH, we perform K-means clustering to divide items into different clusters offline. During online serving time, given a trigger item embedding generated by head i, we first select C nearest clusters based on the embedding similarity between trigger embedding and cluster centroid embeddings. Then, we find the top K_{ann} nearest candidates from items among these C nearest clusters. Since there is no computation dependency between the ANN search for each head, the multi-ANN module can run per-head ANN search in parallel efficiently with limited overhead. Note that in total, this module retrieves $2K_{ann}$ candidate items for each trigger item ($K_{ann} = O(1000)$ in MTMH).

Preranker module. The module uses a multi-task user-to-item (U2I) model as preranker, in order to select top K out of $2K_{ann}$ items retrieved by the multi-ANN module. These K items are expected to have the highest probability of being engaged by the user. Since preranker is trained to maximize the likelihood of selecting items engaged by the user, it may be biased towards highly engaged items by users and thus assign low ranking scores to semantically relevant items. To mitigate such bias, we rank the K_{ann} candidate items retrieved by each head separately, in order to guarantee that K candidate items from each head are preserved ($K = O(10)$).

Candidate merging module. This module takes the output items of the preranker module as input, which is designed to merge the top K candidate items retrieved from each head based on their percentage quota. Specifically, it selects the top $\alpha\%$ candidate items from the engagement head and the top $(100 - \alpha)\%$ candidate items from the relevance head. Note that we remove duplicated candidates from each head to guarantee that the total amount of candidate items after merging is K. A larger α increases the number of candidates retrieved from the engagement head, potentially increasing I2I recall while decreasing I2I semantic relevance; conversely, a smaller α has the opposite effect. It is worth noting that α is a hyperparameter that can be adjusted during serving time. This means that we can flexibly trade between recall and semantic relevance without retraining MTMH. α will be 50 in the remaining sections.

4 Experiments

To assess the effectiveness of MTMH, we conduct a comprehensive evaluation on a commercial recommendation platform serving billions of users. Our evaluation focuses on two key aspects: I2I co-engagement rate (i.e. recall) and I2I semantic relevance. We start with offline evaluation, where we train MTMH alongside other baseline models using user data from Period T_1 and then test these models on real user data from Period T_2 after T_1. The offline evaluation allowed us to compare the performance of MTMH with prior SOTAs in a controlled setting (see Section 4.2). To further validate our findings, we deploy MTMH on the commercial recommendation platform and compare its performance with the production model. The online evaluation provides valuable insights into how MTMH affects the real-world consumption metrics and user-experience-related metrics (see Section 4.6).

4.1 Baselines and Offline Metrics

During offline evaluation, we compare MTMH with four baseline I2I retrieval models. Note that all baseline models are trained to minimize the same co-engagement loss defined in Eq. (1). Additionally, we employ a propensity-score-based method to mitigate the selection bias toward popular items during training [4], and a mixed negative sampling strategy in [60] to reduce the selection bias of negative I2I pairs. The baselines differ only from the model architectures and input item features used to generate item embeddings. We describe them in detail below:

- ItemCF [20]: This model represents each item using an unique embedding vector in Euclidean space purely based on its id, which is originally proposed in [20].
- NeuCF [12]: This model uses a deep neural network (DNN) to learn item embeddings, which has been widely used in prior works. Note that the DNN in this model does not take any item content features as input.
- MoL [62]: Instead of only taking content-unrelated trigger/item features as input, this model also takes content features of items as DNN input.
- HLLM [3]: This model is modified from the most recent work [3], which takes content embedding generated by pre-trained content encoder models as augmented input of I2I retrieval model.
- HSTU* [63]: This model is one of the SOTA user-to-item retrieval models based on transformers [63]. We use the item embedding learnt by HSTU to perform item-to-item retrieval in evaluation.

We report both recall and the semantic relevance of MTMH and baselines. Specifically, for recall, we report recall@K on testing data with $K \in \{5, 20, 100, 500\}$. To measure the semantic relevance, we categorize items into distinct topic category groups based on their semantic relevance and report the topic match rate of <trigger,candidate> item pairs. It's worth noting that L1 topic categories group items into broader topic classes, while L2 topic categories group them into more fine-grained topic classes. We report both to evaluate semantic relevance at varying levels of granularity. Our model hyperparameters and training setups are detailed in Table 3.

4.2 Main Results

We first present offline evaluation results of MTMH and baselines. As shown in Table 1, MTMH achieves both the highest recall and semantic relevance compared with all baselines. Specifically, MTMH increases the recall@500 by at least 4.2% and up to 14.4%. In terms of L2 topic relevance, MTMH outperforms all baselines by at least 6.7% and up to 56.5%.

It is worth noting that both MoL and HLLM have better I2I semantic relevance but worse recall@500 compared with NeuCF. As mentioned in Section 4.1, MoL takes content features as input to generate item embeddings and HLLM uses content embeddings generated by LLM as input, which can greatly improve I2I semantic relevance. However, this usually comes with the expense of sacrificing I2I co-engagement rate. In contrast, MTMH is capable of preserving both highly co-engaged items and semantically relevant items due to its multi-head serving design (see Section 3.5), achieving both improved recall and semantic relevance.

4.3 Ablation Study

Next, we conduct the ablation study of MTMH by comparing it with the following baselines:

- MTMH-H1: It only uses the engagement head of MTMH.

Table 1: Offline evaluation results of MTMH and baselines. Note that we evaluate recall and semantic relevance of these models on testing user data. For recall, we report recall@K with $K \in \{5, 20, 100, 500\}$. For semantic relevance, we report the average L1/L2 topic cateogory match rate between <trigger,candidate> item pairs. L1 topic divides items into relatively coarse-grained semantic classes and L2 topic divides items into more fine-grained semantic classes. Note that both MoL and HLLM are modified versions of prior works for item-to-item retrieval; for HSTU, we utilize its item embeddings for item-to-item retrieval in evaluation.

Baseline	Recall				Semantic Relevance	
	Recall@5	Recall@20	Recall@100	Recall@500	L1 Topic	L2 Topic
ItemCF [20]	1.01%	2.50%	6.89%	14.88%	21.31%	17.90%
NeuCF [12]	1.10% (+8.9%)	2.73% (+9.2%)	7.51% (+9.0%)	16.33% (+9.7%)	25.77% (+20.9%)	22.40% (25.1%)
MoL [62]	1.10% (+8.9%)	2.75%(+10.0%)	7.49% (+8.7%)	16.05% (+7.9%)	27.70% (+30.0%)	24.74% (+38.2%)
HLLM [3]	1.08% (+6.9%)	2.67% (+6.8%)	7.35% (+6.7%)	16.07% (+8.0%)	28.63% (+34.4%)	26.25% (+46.6%)
HSTU* [63]	1.05% (+4.0%)	2.68% (+7.2%)	6.46% (-6.2%)	15.21% (+2.2%)	24.56% (+34.4%)	22.83% (+46.6%)
MTMH (Ours)	**1.10% (+8.9%)**	**2.76% (+10.4%)**	**7.65% (+11.0%)**	**17.02% (+14.4%)**	**30.17% (+41.6%)**	**28.02% (+56.5%)**

Table 2: Ablation study of MTMH. Note that MTMH-H1 and MTMH-H2 denote the engagement head and relevance head of MTMH respectively. STMH is a mult-head model where each head is trained on single-task learning loss, and MTSH represents a single-head model trained on multi-task learning loss.

Baseline	Recall				Semantic Relevance	
	Recall@5	Recall@20	Recall@100	Recall@500	L1 Topic	L2 Topic
MTMH	1.10%	2.76%	7.65%	17.02%	30.17%	28.02%
MTMH-H1	1.11% (+0.9%)	2.77% (+0.4%)	7.57% (-1.0%)	16.16% (-5.1%)	28.58% (-5.3%)	25.73% (-8.2%)
MTMH-H2	1.12% (+1.8%)	2.72% (-1.4%)	6.61% (-13.6%)	11.37% (-33.2%)	34.70% (+15.0%)	33.73% (+20.4%)
MTSH	1.14% (+3.6%)	2.79% (+1.1%)	7.09% (-7.3%)	13.35% (-21.6%)	34.15% (+13.2%)	33.04% (+17.9%)
STMH	1.11% (+0.9%)	2.75% (-0.4%)	7.64% (-1.3%)	14.31% (-15.9%)	28.97% (-4.0%)	27.84% (-0.6%)

Table 3: Hyperparameters and training details.

Name	Value
Number of GPUs	48 A100s
Batch size	2048
Learning rate	0.01
Optimizer	Adagrad
Training epoch	1
Item embedding dim	128
Content embedding dim	128
Visual encoder model	VIT-H14-632M
Text encoder model	XLM-R-Large-550M
Visual encoder output dim	1024
Text encoder output dim	1024
Fusion MLP output dim	128
Output dim of MLP in MTMH	128

- MTMH-H2: It only uses the relevance head of MTMH.
- MTSH (Multi-task single-head): This is a single-head model trained by minimizing multi-task learning loss (see Eq. (3)).
- STMH (Single-task multi-head): This is a multi-head model with one engagement head and one relevance head. Different from MTMH, the engagement head is trained to solely minimize the co-engagement loss (see Eq. (1)), while the relevance head is trained to solely minimize relevance loss (see Eq. (2)).

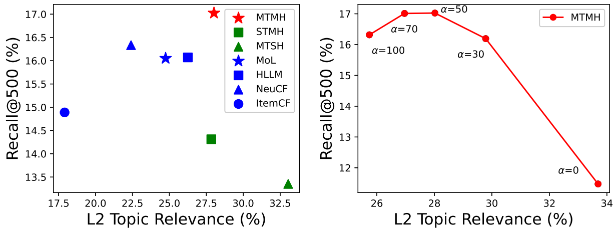
As shown in Table 2, MTMH-H1 (i.e. the engagement head) has the best recall compared with other baselines, and MTMH-H2 (i.e. the relevance head) has the highest I2I semantic relevance while the lowest recall@500 compared with other models. By merging candidates retrieved from both heads, the recall@500 of MTMH is further increased by 5.3% on top of MTMH-H1. At the same time, it improves I2I semantic relevance by 5.6% w.r.t. L1 and 8.9% w.r.t. L2 compared with MTMH-H1, since it preserves both co-engaged and semantically relevant candidates retrieved by MTMH-H2.

Moreover, we observe that both STMH and MTSH trade recall for better semantic relevance. For instance, compared with MTMH-H1 and baselines in Table 1, they exhibit significantly better semantic relevance but much worse recall. By contrast, MTMHis the only model which can improve I2I semantic relevance without trading recall, which demonstrates the effectiveness of multi-head modeling and serving strategy in Section 3.1.

4.4 Recall and Relevance Trade-off

In this subsection, we evaluate the recall and semantic relevance trade-off performance of baselines and MTMH with varying α and w_r. Note that α is a serving hyperparameter controlling the quota for the engagement head during serving time, while w_r is a training hyperparameter determining the weight of relevance loss (see Eq. 3). We use recall@500 and L2 topic relevance as our recall and semantic relevance metrics, and report the results in Figure 6.

Varying α. As shown in Figure 6a, MTMH is able to achieve both high recall and semantic relevance, outperforming all baselines except MTSH (multi-task single-head model) in terms of both metrics. While MTSH achieves the best L2 topic relevance, its recall@500 is



(a) MTMH vs baselines.

(b) MTMH with varying α.

Figure 6: Recall and semantic relevance trade-off evaluation results with varying α. Note that top right part of these figures represents both high recall and semantic relevance. α in MTMH controls the percentage of candidates from its engagement head. In the left figure, the default α value 50 is used.

significantly lower than that of the other models. Figure 6b demonstrates how varying serving parameter α can affect the recall and semantic relevance trade-off of MTMH. We observe that decreasing α improves the I2I semantic relevance, since more semantically relevant items retrieved by the relevance head of MTMH are preserved. By contrast, increasing α boosts the recall, by selecting more highly co-engaged items retrieved by the engagement head of MTMH. Moreover, we observe that when α is larger than 50, decreasing α can enhance I2I recall and semantic relevance simultaneously. This is expected since items with less co-engagement efficiency retrieved by the engagement head of MTMH are replaced by highly relevant and co-engaged items retrieved from the relevance head of MTMH . However, keeping reducing α leads to the drop of recall, specifically when $\alpha > 70$, since the relevance head may ignore highly co-engaged but less semantically relevant items (e.g. popular items). In summary, we observe that $\alpha = 50$ provides us with the optimal recall and semantic relevance trade-off.

Note that in practice, α can be flexibly adjusted to adapt to the specific requirements in production. For instance, for some applications where I2I semantic relevance is more important, smaller α can be used; while for applications focusing on more co-engagement rate, larger α should be used. By enabling α as a serving parameter, MTMH model can be deployed to serve different purposes without being retrained.

Varying w_r. Table 4 shows the additional results with varying values of w_r in the table below. It can be seen that larger w_r generally increases relevance but decreases recall. Overall, we observe that w_r=0.5 provides a decent trade-off between recall and relevance. Therefore, we use 0.5 as a default value for w_r during both offline and online experiments.

4.5 Embedding Convergence of Fresh Content

Most industry retrieval models, including I2I, rely heavily on user engagement data, leading to a strong bias toward older content. However, a key aspect of recommendation systems is the rapid delivery of fresh content, even with minimal user engagement, as it enhances the overall user experience. To achieve this, the I2I retrieval model must learn fresh content embeddings more quickly, ensuring their fast convergence during training. To assess whether MTMH accelerates fresh content delivery, we analyze the convergence speed of fresh content embeddings during MTMH training. Figure 7 compares the embedding convergence speed of fresh content between MTMH and our production model. The x-axis represents the number of updates to fresh content embeddings, while the y-axis denotes the embedding delta—defined as the L2 distance between embeddings at step t and those at step $T = 2000$. Our results show that MTMH achieves faster embedding convergence for fresh content compared to the production model. This finding is further validated by online experiments (Section 4.6), which confirm that MTMH successfully delivers more fresh content, aligning with expectations based on its improved embedding convergence. The reason is that our MTMH not only learns the item embeddings from the co-engagement data, but also incorporates the content semantic relevance through the multi-task learning, improving the model generalization on fresh content.

Table 4: Recall and semantic relevance trade-off evaluation results with varying w_r.

w_r	Recall	Semantic Relevance	
	ΔRecall@500	ΔL1 Topic	ΔL2 Topic
0	-2.99%	-13.57%	-19.87%
0.25	+0.96%	-3.81%	-5.54%
0.5 (Default)	–	–	–
1.0	-0.09%	+3.07%	+4.45%
5.0	-7.38%	+9.01%	+13.22%

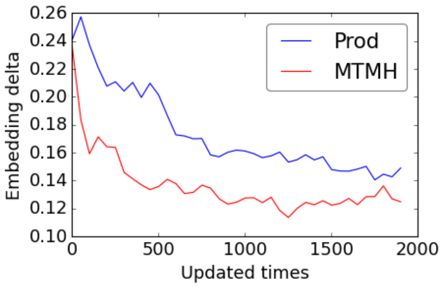


Figure 7: MTMH embedding convergence of fresh content.

4.6 Online A/B Testing

We deployed MTMH on a real-world platform serving billions of users and conducted a 7-day evaluation to assess its effectiveness. We observe significant gains on both topline consumption metrics and user-experience-related metrics. Table 5 reports the improvements in several key metrics, which are described below.

Daily active users (DAU). This metric measures the number of unique users who engage with the platform on a daily basis. We observe 0.05% increase in DAU, indicating that more users are using the platform each day.

Daily time Spent. This metric measures the amount of time users spent on the platform within a day. The deployment of MTMH leads to a 0.22% increase in users' time spent, suggesting that users are more engaged with the recommended items on the platform.

Distinct item views. This metric counts the number of unique items viewed by users on the platform in a day. An increase in distinct item views indicates that more diverse and unique items are retrieved and recommended to users. We report that MTMH brings 0.31% more distinct items into the platform.

Percentage of fresh content. This metric measures the percentage of fresh content with age less than 48 hours on the platform. MTMH improves this metric by 0.25%, indicating that users are seeing more fresh content.

Novel interest discovery rate. This metric tracks the number of new interests or topics that users discover on the platform. An increase in novel interest discovery rate indicates that users' new areas of interest can be discovered by the recommendation system faster. We observe that MTMH can also increase this metric by 0.33%.

User interest recall. This metric measures how well the platform is able to recommend content that aligns with a user's existing interests. An improvement in user interest recall suggests that our model is able to retrieve more semantically relevant content. As expected, MTMH successfully moves this metric up by 0.14%.

In summary, we conclude that MTMH improves both I2I co-engagement efficiency and semantic relevance during online A/B testing, consistent with what we observe during offline evaluation.

Table 5: Online A/B testing results for MTMH .

Metrics	Changes
Daily active users	+0.05%
Daily time spent	+0.22%
Distinct item views	+0.31%
Perentage of fresh content	+0.25%
Novel interest discovery rate	+0.33%
User interest recall	+0.14%

5 Related Work

I2I retrieval models. Early I2I retrieval methods include collaborative filtering [38, 39, 42], matrix factorization [16, 22, 23] and neighbor-based methods [30, 40]. They primarily focus on modeling interactions between raw item IDs without considering rich features like item attributes or content features [21, 28, 37, 54], which are insufficient for capturing complex patterns and relationships between items. Recent years, deep neural networks (DNNs) have been used in I2I retrieval to capture complex patterns and relationships between items. Among them, two-tower model architecture has emerged as a dominant paradigm, offering both effectiveness and efficiency [7, 17, 61]. Along this line of work, various techniques like adaptive mechanisms [26, 57, 61] and self-attention [24, 53] have been used to enhance two-tower models with richer input features while maintaining computational efficiency. However, these models are trained purely based on co-engagement data without considering I2I semantic relevance. To tackle the semantic understanding challenge, various approaches have been proposed to take content features as model input to improve I2I semantic relevance [13, 14, 24, 32–34, 43, 49]. Different from prior works, MTMH provides a principled approach for jointly optimizing co-engagement efficiency and semantic relevance via multi-task learning loss without taking content features as input.

Large foundation models for retrieval. With the emergence of large foundation models (e.g. large language models (LLMs)), there has been growing interest in leveraging their superior semantic understanding capabilities for recommendation tasks [2, 10, 25, 29, 31, 47, 51, 64, 65]. One line of work use LLMs for generative recommendations through prompt engineering [8, 11]. Another line of work integrate LLMs into retrieval systems [3, 8, 27, 46, 48] to better understand complex item relationships. However, these approaches face practical deployment challenges due to hugh runtime cost. More recent works have explored methods to inject LLM knowledge into recommendation models [6, 45], either by enhancing item representations through content feature extraction [50, 52, 66], or leveraging LLMs for data augmentation [5] and knowledge distillation [56]. In contrast to prior work, MTMH provides a principled approach for distilling LLMs' knowledge into retrieval models via multi-task learning without increasing model complexity. Moreover, the multi-head architecture design in MTMHenables us to flexibly trade between co-engagement efficiency and semantic relevance, which is unexplored in prior works.

6 Conclusion

This paper proposes MTMH, a multi-task and multi-head item-to-item (I2I) retrieval model that addresses the fundamental trade-off between recall and semantic relevance. MTMH provides a principled approach for jointly optimizing I2I co-engagement rate and semantic relevance, via a multi-task learning loss and a multi-head retrieval architecture. Our offline experimental results demonstrate that MTMH improves I2I retrieval recall by up to 14.4% and semantic relevance by up to 56.6%, outperforming all baselines. Our online A/B testing further verifies its effectiveness in enhancing both topline consumption metrics (e.g. daily active user and time spent) and user-experience-related metrics (e.g user interest recall, novel interest discovery rate, content diversity, and freshness). Overall, this work has the potential to significantly enhance the performance of recommendation systems in various applications.

References

[1] Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. 2024. Llm2vec: Large language models are secretly powerful text encoders. *arXiv preprint arXiv:2404.05961* (2024).

[2] Artun Boz, Wouter Zorgdrager, Zoe Kotti, Jesse Harte, Panagiotis Louridas, Dietmar Jannach, and Marios Fragkoulis. 2024. Improving Sequential Recommendations with LLMs. *ArXiv* abs/2402.01339 (2024). https://api.semanticscholar.org/CorpusID:267406555

[3] Junyi Chen, Lu Chi, Bingyue Peng, and Zehuan Yuan. 2024. Hllm: Enhancing sequential recommendations via hierarchical large language models for item and user modeling. *arXiv preprint arXiv:2409.12740* (2024).

[4] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2023. Bias and debias in recommender system: A survey and future directions. *ACM Transactions on Information Systems* 41, 3 (2023), 1–39.

[5] Lei Chen, Chen Gao, Xiaoyi Du, Hengliang Luo, Depeng Jin, Yong Li, and Meng Wang. 2024. Enhancing ID-based Recommendation with Large Language Models. *ArXiv* abs/2411.02041 (2024). https://api.semanticscholar.org/CorpusID:273812191

[6] Zhixuan Chu, Hongyan Hao, Ouyang Xin, Simeng Wang, Yan Wang, Yue Shen, Jinjie Gu, Qing Cui, Longfei Li, Siqiao Xue, James Y. Zhang, and Shenghe Li. 2023. Leveraging Large Language Models for Pre-trained Recommender Systems. *ArXiv* abs/2308.10837 (2023). https://api.semanticscholar.org/CorpusID:261049176

[7] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*. 191–198.

[8] Sunhao Dai, Ninglu Shao, Haiyuan Zhao, Weijie Yu, Zihua Si, Chen Xu, Zhongxiang Sun, Xiao Zhang, and Jun Xu. 2023. Uncovering chatgpt's capabilities in

recommender systems. In *Proceedings of the 17th ACM Conference on Recommender Systems*. 1126–1132.

[9] Mukund Deshpande and George Karypis. 2004. Item-based top-n recommendation algorithms. *ACM Transactions on Information Systems (TOIS)* 22, 1 (2004), 143–177.

[10] Wenqi Fan, Zihuai Zhao, Jiatong Li, Yunqing Liu, Xiaowei Mei, Yiqi Wang, Jiliang Tang, and Qing Li. 2023. Recommender Systems in the Era of Large Language Models (LLMs). *IEEE Transactions on Knowledge and Data Engineering* 36 (2023), 6889–6907. https://api.semanticscholar.org/CorpusID:259342486

[11] Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. 2022. Recommendation as Language Processing (RLP): A Unified Pretrain, Personalized Prompt & Predict Paradigm (P5). *Proceedings of the 16th ACM Conference on Recommender Systems* (2022). https://api.semanticscholar.org/CorpusID:247749019

[12] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*. 173–182.

[13] Balázs Hidasi and Alexandros Karatzoglou. 2017. Recurrent Neural Networks with Top-k Gains for Session-based Recommendations. *Proceedings of the 27th ACM International Conference on Information and Knowledge Management* (2017). https://api.semanticscholar.org/CorpusID:1159769

[14] Balázs Hidasi and Domonkos Tikk. 2013. Context-aware item-to-item recommendation within the factorization framework. In *Proceedings of the 3rd Workshop on Context-awareness in Retrieval and Recommendation*. 19–25.

[15] Geoffrey Hinton. 2015. Distilling the Knowledge in a Neural Network. *arXiv preprint arXiv:1503.02531* (2015).

[16] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE International Conference on Data Mining*. 263–272.

[17] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*. 2333–2338.

[18] Amir H Jadidinejad, Craig Macdonald, and Iadh Ounis. 2020. Using exploration to alleviate closed loop effects in recommender systems. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2025–2028.

[19] Ziyan Jiang, Rui Meng, Xinyi Yang, Semih Yavuz, Yingbo Zhou, and Wenhu Chen. 2024. Vlm2vec: Training vision-language models for massive multimodal embedding tasks. *arXiv preprint arXiv:2410.05160* (2024).

[20] Mohammad Khoshneshin and W Nick Street. 2010. Collaborative filtering via euclidean embedding. In *Proceedings of the fourth ACM conference on Recommender systems*. 87–94.

[21] Mohammad Khoshneshin and William Nick Street. 2010. Collaborative filtering via euclidean embedding. In *ACM Conference on Recommender Systems*. https://api.semanticscholar.org/CorpusID:7176266

[22] Yehuda Koren. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Knowledge Discovery and Data Mining*. https://api.semanticscholar.org/CorpusID:207168823

[23] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37.

[24] Chao Li, Zhiyuan Liu, Mengmeng Wu, Yuchi Xu, Huan Zhao, Pipei Huang, Guoliang Kang, Qiwei Chen, Wei Li, and Dik Lun Lee. 2019. Multi-interest network with dynamic routing for recommendation at Tmall. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2615–2623.

[25] Lei Li, Yongfeng Zhang, Dugang Liu, and L. Chen. 2023. Large Language Models for Generative Recommendation: A Survey and Visionary Discussions. In *International Conference on Language Resources and Evaluation*. https://api.semanticscholar.org/CorpusID:261531422

[26] Xiangyang Li, Bo Chen, Huifeng Guo, Jingjie Li, Chenxu Zhu, Xiang Long, Sujian Li, Yichao Wang, Wei Guo, Longxia Mao, Jinxing Liu, Zhenhua Dong, and Ruiming Tang. 2022. IntTower: The Next Generation of Two-Tower Model for Pre-Ranking System. *Proceedings of the 31st ACM International Conference on Information & Knowledge Management* (2022). https://api.semanticscholar.org/CorpusID:252904690

[27] Xinyi Li, Yongfeng Zhang, and Edward C Malthouse. 2023. Pbnr: Prompt-based news recommender system. *arXiv preprint arXiv:2304.07862* (2023).

[28] Dawen Liang, Laurent Charlin, James McInerney, and David M Blei. 2016. Modeling user exposure in recommendation. In *Proceedings of the 25th International Conference on World Wide Web*. 951–961.

[29] Jianghao Lin, Xinyi Dai, Yunjia Xi, Weiwen Liu, Bo Chen, Xiangyang Li, Chenxu Zhu, Huifeng Guo, Yong Yu, Ruiming Tang, and Weinan Zhang. 2023. How Can Recommender Systems Benefit from Large Language Models: A Survey. *ArXiv* abs/2306.05817 (2023). https://api.semanticscholar.org/CorpusID:259129651

[30] G. Linden, B. Smith, and J. York. 2003. Amazon.com recommendations: item-to-item collaborative filtering. *IEEE Internet Computing* 7, 1 (2003), 76–80. doi:10.1109/MIC.2003.1167344

[31] Qidong Liu, Xiangyu Zhao, Yuhao Wang, Yejing Wang, Zijian Zhang, Yuqi Sun, Xiang Li, Maolin Wang, Pengyue Jia, Chong Chen, Wei Huang, and Feng Tian. 2024. Large Language Model Enhanced Recommender Systems: Taxonomy, Trend, Application and Future. *ArXiv* abs/2412.13432 (2024). https://api.semanticscholar.org/CorpusID:274822665

[32] Zheng Liu, Jianxun Lian, Junhan Yang, Defu Lian, and Xing Xie. 2020. Octopus: Comprehensive and elastic user representation for the generation of recommendation candidates. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 289–298.

[33] Pasquale Lops, Marco Degemmis, and Giovanni Semeraro. 2011. Content-based Recommender Systems: State of the Art and Trends. In *Recommender Systems Handbook*. https://api.semanticscholar.org/CorpusID:6102334

[34] Junmei Lv, Bin Song, Jie Guo, Xiaojiang Du, and Mohsen Guizani. 2019. Interest-related item similarity model based on multimodal data for top-N recommendation. *IEEE access* 7 (2019), 12809–12821.

[35] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018).

[36] Qi Pi, Guorui Zhou, Yujing Zhang, Zhe Wang, Lejian Ren, Ying Fan, Xiaoqiang Zhu, and Kun Gai. 2020. Search-based user interest modeling with lifelong sequential behavior data for click-through rate prediction. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2685–2692.

[37] Steffen Rendle. 2010. Factorization machines. In *2010 IEEE International Conference on Data Mining*. 995–1000.

[38] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. BPR: Bayesian personalized ranking from implicit feedback. In *International Conference on Machine Learning*. 452–460.

[39] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2001. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International Conference on World Wide Web*. 285–295.

[40] Badrul Munir Sarwar, George Karypis, Joseph A. Konstan, and John Riedl. 2000. Analysis of recommendation algorithms for e-commerce. In *ACM Conference on Economics and Computation*. https://api.semanticscholar.org/CorpusID:12366165

[41] Tobias Schnabel and Paul N Bennett. 2020. Debiasing item-to-item recommendations with small annotated datasets. In *Proceedings of the 14th ACM Conference on Recommender Systems*. 73–81.

[42] Xiaoyuan Su and Taghi M Khoshgoftaar. 2009. A survey of collaborative filtering techniques. *Advances in Artificial Intelligence* 2009 (2009).

[43] Qiaoyu Tan, Jianwei Zhang, Jiangchao Yao, Ninghao Liu, Jingren Zhou, Hongxia Yang, and Xia Hu. 2021. Sparse-interest network for sequential recommendation. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 598–606.

[44] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971* (2023).

[45] Hangyu Wang, Jianghao Lin, Xiangyang Li, Bo Chen, Chenxu Zhu, Ruiming Tang, Weinan Zhang, and Yong Yu. 2023. FLIP: Fine-grained Alignment between ID-based Models and Pretrained Language Models for CTR Prediction. In *ACM Conference on Recommender Systems*. https://api.semanticscholar.org/CorpusID:264814457

[46] Hanbing Wang, Xiaorui Liu, Wenqi Fan, Xiangyu Zhao, Venkataramana B. Kini, Devendra Yadav, Fei Wang, Zhen Wen, Jiliang Tang, and Hui Liu. 2024. Rethinking Large Language Model Architectures for Sequential Recommendations. *ArXiv* abs/2402.09543 (2024). https://api.semanticscholar.org/CorpusID:267682048

[47] Qi Wang, Jindong Li, Shiqi Wang, Qianli Xing, Runliang Niu, He Kong, Rui Li, Guodong Long, Yi Chang, and Chengqi Zhang. 2024. Towards Next-Generation LLM-based Recommender Systems: A Survey and Beyond. *ArXiv* abs/2410.19744 (2024). https://api.semanticscholar.org/CorpusID:273653859

[48] Yancheng Wang, Ziyan Jiang, Zheng Chen, Fan Yang, Yingxue Zhou, Eunah Cho, Xing Fan, Xiaojiang Huang, Yanbin Lu, and Yingzhen Yang. 2023. Recmind: Large language model powered agent for recommendation. *arXiv preprint arXiv:2308.14296* (2023).

[49] Yuyan Wang, Mohit Sharma, Can Xu, Sriraj Badam, Qian Sun, Lee Richardson, Lisa Chung, Ed H Chi, and Minmin Chen. 2022. Surrogate for long-term user experience in recommender systems. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*. 4100–4109.

[50] Wei Wei, Xubin Ren, Jiabin Tang, Qinyong Wang, Lixin Su, Suqi Cheng, Junfeng Wang, Dawei Yin, and Chao Huang. 2023. LLMRec: Large Language Models with Graph Augmentation for Recommendation. *Proceedings of the 17th ACM International Conference on Web Search and Data Mining* (2023). https://api.semanticscholar.org/CorpusID:264832979

[51] Likang Wu, Zhilan Zheng, Zhaopeng Qiu, Hao Wang, Hongchao Gu, Tingjia Shen, Chuan Qin, Chen Zhu, Hengshu Zhu, Qi Liu, Hui Xiong, and Enhong Chen. 2023. A Survey on Large Language Models for Recommendation. *ArXiv* abs/2305.19860 (2023). https://api.semanticscholar.org/CorpusID:258987581

[52] Yunjia Xi, Weiwen Liu, Jianghao Lin, Jieming Zhu, Bo Chen, Ruiming Tang, Weinan Zhang, Rui Zhang, and Yong Yu. 2023. Towards Open-World Recommendation with Knowledge Augmentation from Large Language Models. *ArXiv* abs/2306.10933 (2023). https://api.semanticscholar.org/CorpusID:259202547

[53] Zhibo Xiao, Luwei Yang, Tao Zhang, Wen Jiang, Wei Ning, and Yujiu Yang. 2024. Deep Evolutional Instant Interest Network for CTR Prediction in Trigger-Induced Recommendation. *ArXiv* (2024).

[54] Xin Xin, Fajie Yuan, Xiangnan He, and Joemon M Jose. 2018. Batch is not heavy: Learning word representations from all samples. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*. 1853–1862.

[55] Hu Xu, Saining Xie, Xiaoqing Ellen Tan, Po-Yao Huang, Russell Howes, Vasu Sharma, Shang-Wen Li, Gargi Ghosh, Luke Zettlemoyer, and Christoph Feichtenhofer. 2023. Demystifying clip data. *arXiv preprint arXiv:2309.16671* (2023).

[56] Xiaohan Xu, Ming Li, Chongyang Tao, Tao Shen, Reynold Cheng, Jinyang Li, Can Xu, Dacheng Tao, and Tianyi Zhou. 2024. A Survey on Knowledge Distillation of Large Language Models. *ArXiv* abs/2402.13116 (2024). https://api.semanticscholar.org/CorpusID:267760021

[57] Zhenhui Xu, Meng Zhao, Liqun Liu, Lei Xiao, Xiaopeng Zhang, and Bifeng Zhang. 2022. Mixture of virtual-kernel experts for multi-objective user profile modeling. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 4257–4267.

[58] Feng Xue, Xiangnan He, Xiang Wang, Jiandong Xu, Kai Liu, and Richang Hong. 2019. Deep item-based collaborative filtering for top-n recommendation. *ACM Transactions on Information Systems (TOIS)* 37, 3 (2019), 1–25.

[59] Jing Yan, Liu Jiang, Jianfei Cui, Zhichen Zhao, Xingyan Bin, Feng Zhang, and Zuotao Liu. 2024. Trinity: Syncretizing Multi-/Long-Tail/Long-Term Interests All in One. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2024, Barcelona, Spain, August 25-29, 2024*, Ricardo Baeza-Yates and Francesco Bonchi (Eds.). ACM, 6095–6104. https://doi.org/10.1145/3637528.3671651

[60] Ji Yang, Xinyang Yi, Derek Zhiyuan Cheng, Lichan Hong, Yang Li, Simon Xiaoming Wang, Taibai Xu, and Ed H Chi. 2020. Mixed negative sampling for learning two-tower neural networks in recommendations. In *Companion proceedings of the web conference 2020*. 441–447.

[61] Yantao Yu, Weipeng Wang, Zhoutian Feng, and Daiyue Xue. 2021. A dual augmented two-tower model for online large-scale recommendation. In *DLP-KDD*.

[62] Jiaqi Zhai, Zhaojie Gong, Yueming Wang, Xiao Sun, Zheng Yan, Fu Li, and Xing Liu. 2023. Revisiting Neural Retrieval on Accelerators. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 5520–5531.

[63] Jiaqi Zhai, Lucy Liao, Xing Liu, Yueming Wang, Rui Li, Xuan Cao, Leon Gao, Zhaojie Gong, Fangda Gu, Michael He, et al. 2024. Actions speak louder than words: Trillion-parameter sequential transducers for generative recommendations. *arXiv preprint arXiv:2402.17152* (2024).

[64] Weizhi Zhang, Yuan-Qi Bei, Liangwei Yang, Henry Peng Zou, Peilin Zhou, Aiwei Liu, Yinghui Li, Hao Chen, Jianling Wang, Yu Wang, Feiran Huang, Sheng Zhou, Jiajun Bu, Allen Lin, James Caverlee, Fakhri Karray, Irwin King, and Philip S. Yu. 2025. Cold-Start Recommendation towards the Era of Large Language Models (LLMs): A Comprehensive Survey and Roadmap. https://api.semanticscholar.org/CorpusID:275323883

[65] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Z. Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jianyun Nie, and Ji rong Wen. 2023. A Survey of Large Language Models. *ArXiv* abs/2303.18223 (2023). https://api.semanticscholar.org/CorpusID:257900969

[66] Zhi Zheng, WenShuo Chao, Zhaopeng Qiu, Hengshu Zhu, and Hui Xiong. 2024. Harnessing Large Language Models for Text-Rich Sequential Recommendation. *Proceedings of the ACM on Web Conference 2024* (2024). https://api.semanticscholar.org/CorpusID:268536921