Abstract Sequential recommendation (SR) models often capture user preferences based on the historically interacted item IDs, which usually obtain sub-optimal performance when the interaction history is limited. Content-based sequential recommendation has recently emerged as a promising direction that exploits items' textual and visual features to enhance preference learning. However, there are still three key challenges: (i) how to reduce the semantic gap between different content modality representations; (ii) how to jointly model user behavior preferences and content preferences; and (iii) how to design an effective training strategy to align ID representations and content representations. To address these challenges, we propose a novel model, self-supervised representation learning with ID-Content modality alignment, named SICSRec. Firstly, we propose a LLM-driven sample construction method and develop a supervised fine-tuning approach to align item-level modality representations. Secondly, we design a novel Transformer-based sequential model, where an ID-modality sequence encoder captures user behavior preferences, a content-modality sequence encoder learns user content preferences, and a mix-modality sequence decoder grasps the intrinsic relationship between these two types of preferences. Thirdly, we propose a two-step training strategy with a content-aware contrastive learning task to align modality representations and ID representations, which decouples the training process of content modality dependency and item collaborative dependency. Extensive experiments conducted on four public video streaming datasets demonstrate our SICSRec outperforms the state-of-the-art ID-modality sequential recommenders and content-modality sequential recommenders by 8.04% on NDCG@5 and 6.62% on NDCD@10 on average, respectively.

Keywords Sequential recommendation, supervised fine-tuning, representation alignment.

1 Introduction

Sequential recommendation aims to model user preferences based on their historical behaviors (i.e., click, purchase, etc.) and predict the next item for the user. Existing methods utilize a unique item ID to represent each item, and design different neural networks to model user-item interactions (e.g., RNN [1], Transformer [2, 3], MLP [4] and GNN [5, 6]). This ID modality-based sequential modeling paradigm essentially learns item-to-item collaborative dependency from a statistical perspective. Previous works have achieved great success in the last decade, but often lead to sub-optimal performance and poor generalization ability when user-item interactions are few [7].

Recently, content-modality based recommendation becomes a promising direction that exploits the item content modality information (e.g., text and image) to enhance user preference learning [8, 9]. These content modality data widely exist on online platforms such as YouTube, TikTok, and BiliBili, which contain rich semantic information about the items. Furthermore, modeling user content modality information gives a comprehensive view to understand user preferences [10, 11]. For example, in a fashion recommendation scenario, a user may be attracted by a stylish clothing image or a promotional description [12]. The decision of users

to click or not within an online website is usually impacted by the visual characteristics [13].

Existing content modality-based sequential recommendation methods model the semantic similarity between different items to enhance the item representations, and combine the users' content dependency and item collaborative dependency to improve the recommendation performance [14–16]. Despite the advanced progress, there are still challenges.

(1) *How to reduce the semantic gap between different content modality information and obtain a unified item content representations?* The item content modality representations come from a pre-trained language embedding space or a vision embedding space. Due to differences in training data and optimization objectives, there is an inherent semantic gap between the text modality representations and image modality representations, even when they represent the same item [17–19].

(2) *How to acquire accurate user preferences by jointly modeling content sequences and ID sequences?* In our methods, we mainly consider two preferences, i.e., behavior preferences and content preferences. User behavior preferences tend to reflect the item-to-item collaborative dependency by modeling ID sequences. However, the content representations of a corresponding item sequence lacks an understanding of the personal intents. It thus needs to explicitly learn user intents from both the content sequences and ID sequences for better user preference learning.

(3) *How to design an effective training strategy to align ID modality representations and content modality representations?* The item ID modality is the most commonly used modality, which serves as a fine-grained numerical feature. The item content modality comes from a pre-trained embedding space, and provides more coarse-grained semantic information. Due to the heterogeneity of these two modalities, end-to-end training of a recommender that combines content representations and ID representations may lead to unstable performance caused by representation interference.

To tackle these issues, we propose a novel self-supervised representation learning framework (i.e., SICSRec) with ID-content modality alignment for sequential recommendation. In this paper, we mainly consider two types of modality information, i.e., item ID, and item content (i.e., text and image).

For the first challenge, we propose a novel content modality semantic alignment module to reduce the semantic gap between different modalities. We first propose a novel LLM-driven sample construction method, which leverages LLM as a semantic discriminator to select the most similar item-content modality pairs from users' interaction sequences. Then, we design a supervised fine-tuning approach to jointly tune the text and image encoders, which facilitates item-level modality representation alignment and obtains representations that are better suitable for downstream recommendation tasks.

For the second challenge, we first adopt an L2 normalization to transfer the multi-content representations into a unified content representation space. Then we develop a novel Transformer-based encoder-decoder model, where an ID-modality sequence encoder captures user behavior preferences from the item-ID sequence, a content-modality sequence encoder learns user content preferences from the item-content sequence, and a mix-modality sequence decoder grasps the intrinsic relationship between these two types of preferences. We aggregate these three outputs as the final user preferences.

For the third challenge, we adopt a two-step training strategy to train our model, which decouples

the training process of content dependency and item collaborative dependency. Firstly, we pre-train an ID modality sequence encoder with a standard cross-entropy loss, and then fix its weights. Then, we propose content-aware contrastive learning as an auxiliary loss, which aligns user preferences with content-modality representations. We utilize a low-rank adaptation layer to extract user behavior preferences from the ID-modality encoder, and post-train other components.

In the experiments, we compare our SICSRec with eleven competitive baselines including ID modality-based sequential recommenders and content modality-based sequential recommenders on four public video streaming datasets. Our SICSRec achieves significant improvement in ranking-oriented evaluation metrics. Moreover, we conduct extensive ablation studies to validate our model's components. The contributions are summarized as follows:

(1) We propose a novel content-modality semantic alignment method, reducing the semantic gap between different content-modality representations. It reveals that using LLM for data construction and supervised fine-tuning for the content encoder is a promising way to enhance item-level content representations.

(2) We design a Transformer-based encoder-decoder architecture to model user behaviors, content preferences, and their inherent relationships. We further introduce a novel content-aware contrastive learning task and an effective two-step training strategy to combine content dependency and item collaborative dependency, facilitating efficient representation learning.

(3) Extensive experiments on four public datasets show that our SICSRec outperforms the state-of-the-art baselines. Additionally, the ablation study highlights the effectiveness of each key component and the robustness of performance.

2 Related Work

2.1 ID-based Sequential Recommendation

Classical sequential recommendation methods utilize unique item IDs to represent each item and design different neural networks to model a user's long-term and short-term preferences. Previous works have introduced RNN (e.g., GRU4Rec [1]), Transformer (e.g., SASRec [2], BERT4Rec [3], Transformer4Rec [20], RETR [21]), MLP (e.g., FMLP-Rec [22], MMMLP [23] and BMLP [4]) and GNN (e.g., SR-GNN [5] and BA-GNN [6]) to sequential modeling.

Further works leverage side information, such as item category or price as prior knowledge, and design different fusion networks to enhance item representations [24, 25]. For example, Cafe [26] explicitly learns coarse-grained user intents for item category sequences. DIF [27] proposes a decoupled fusion method to adaptively fuse side information and item representations. Furthermore, MSSR [28] designs a multi-sequence integrated attention layer and a user representation alignment module to optimize representation learning.

Self-supervised learning is utilized to grasp supervision signals from user interaction sequences or attribute sequences to enhance user preference learning [29, 30]. For example, SelfGNN [31] encodes short-term collaborative relationships via graph neural networks and captures stable user representations via self-augmented learning. S3Rec [32] proposes four auxiliary self-supervised objectives to learn the intrinsic data correlation and enhance user representation learning. DuoRec [30] focuses

on the representation degeneration issue and proposes a contrastive regularization term to alleviate it. DCRec [33] adopts a debiased contrastive learning paradigm to capture item-level and user-level dependencies. Furthermore, Poisoning-SSL [34] explores the feasibility of the poisoning attacks on self-supervised learning methods and the weaknesses of these methods.

The modeling paradigm of the above methods can be called item ID modality-based sequential recommendation, which represents each user or item with ID representations, and essentially learns the item-item collaborative relationship from a statistical perspective. However, despite advanced progress, the item ID-based sequential recommendation still struggles with data sparsity and cold-start issues, leading to sub-optimal performance and poor generalization ability when user interactions are few.

2.2 Content-based Sequential Recommendation

Content-based sequential recommendation aims to utilize item content information (e.g., item image and item text) to enhance item representations [10, 35, 36]. This modeling paradigm can be categorized into two main branches: content-centric SR methods and content-enhanced SR methods.

Inspired by the success of pre-trained models, content-centric SR methods focus on using text or image modality representations to replace the item-ID modality representations and directly conduct end-to-end recommendation. For example, MoRec trains the modality encoder and recommender jointly with end-to-end training, which achieves similar performance compared to the ID modality-based recommender in some scenarios [37,38]. Recformer [39] models item text features as language representations and trains Longformer to understand recommendation tasks. TASTE [40] uses T5 as the backbone, represents items and users with text, and predicts the next item for each user based on the relevance of the text representations. However, end-to-end training for content modality-based recommenders is costly, which may not meet real-time inference requirements in personalized recommendation.

The content-enhanced SR methods utilize a pre-trained modality model as a feature encoder to obtain the item-level content representations, and then combine them with the ID representations to enhance the recommendation performance. This modeling paradigm is effective and efficient to deploy in industry. For example, some works utilize pre-trained BERT to extract side features from reviews to enhance item representations [41, 42]. UniSRec [14] learns universal item representations from associated description text of items, and introduces two contrastive pre-training tasks to build transferable recommendation. VQ-Rec [15] maps item text embedding into multiple code embedding, and designs a differentiable permutation-based network for recommendation. MISSRec extends UniSRec to a multi-modal learning framework, which jointly models image and text preferences [16]. Moreover, MSRec [43] proposes a mixture-of-experts (MoE) fusion network for multi-modal information fusion. MML [44] designs a group of multimodal meta-learners, each learns the corresponding kind of modality information, and fuses them in the prediction layer. M5 [45] learns content graph embeddings from a metagraph, and combines the ID embeddings in multi-interest extraction layer.

These works design different deep networks, which combines the ID modality representations and content modality representations to enhance user preference learning. However, these works rarely consider the item-level semantic gap between differ-

ent content modality representations and the heterogeneity of content modality and ID modality, which may lead to unstable performance and difficulty in model convergence.

3 Methodology

3.1 Problem Definition

For each $u \in \mathcal{U}$, we define $S_{id} = \{i_1, i_2, .., i_n\}$ as the item-ID sequence, $S_{text} = \{t_1, t_2, ..., t_n\}$ as the item-text sequence, and $S_{ima} = \{g_1, g_2, ..., g_n\}$ as the item-image sequence, where n is the fixed length of a sequence. These three sequences are sorted in chronological order. For each item i_k at time step k, there is a corresponding content pair (t_i, g_i), where t_i is the item text and g_i is the item image. Our goal is to exploit these three sequences to predict the user's next preferred item i_{n+1}, which can be formulated as follows,

$$\arg\max(i_{n+1}|S_{id}, S_{text}, S_{ima}). \quad (1)$$

3.2 Overview of Our SICSRec

The overall framework of our SICSRec is illustrated in Figure 1, consisting of three parts, i.e., (i) content modality semantic alignment in Section 3.3, (ii) sequence preference learning in Section 3.4, and (iii) a two-step training strategy with content-aware contrastive learning task in Section 3.5. In the first part, we design an LLM-driven sample construction method and a supervised fine-tuning method of joint text encoder and image encoder to achieve item-level modality representation alignment. In the second part, we propose a Transformer-based encoder-decoder model for user preference learning. In the third part, we design a content-aware contrastive learning task to align content representations and ID representations and adopt a two-step training strategy to train our model. The important notations and their explanations are listed in Table 1.

Table 1 Notations and their explanations.

Notation	Description
U	The user set
I	The item set
S_{id}	The item-ID sequence
S_{text}	The item-text sequence
S_{ima}	The item-image sequence
$M_e \in \mathbb{R}^{\|I\|\times d}$	The input item-ID embedding matrix
$M_t \in \mathbb{R}^{\|I\|\times d}$	The input item-text embedding matrix
$M_g \in \mathbb{R}^{\|I\|\times d}$	The input item-image embedding matrix
$M_c \in \mathbb{R}^{\|I\|\times d}$	The input item-content embedding matrix
n	the maximum length of a sequence
d	the latent vector dimension

3.3 Content Modality Semantic Alignment

Directly using the pre-trained content representations in the recommendation model may lead to sub-optimal performance. There are two main reasons: first, the item content representations (i.e., text and image) come from a pre-trained language embedding space and a vision embedding space. Due to the inconsistency in training data and objectives, there is an item-level semantic gap between text and image representations for the same item. Secondly, the original content representations lack an understanding of the item-item collaborative information for a specific recommendation scenario because the content encoders do not exploit the user-item interaction history.

Some previous works [46, 47] show that fine-tuning a content encoder for a domain-specific data can effectively improve the content representations in downstream tasks. Large language model has powerful semantic discriminative capabilities, which have achieved significant success across a range of

tasks, such as text generation and classification. Inspired by the sample efficiency of LLM-enhanced recommender systems [48], in this paper, we propose an LLM-driven sample construction method, which uses an LLM to select some most similar content pairs from a user's historically interacted items. Then, we design an effective supervised fine-tuning method for item-level modality representation alignment, which jointly tunes the text encoder and the image encoder based on the LLM-selected data. After that, we utilize these tuned content encoders to obtain high-quality content embeddings, and combine them with ID embeddings in the downstream recommendation tasks.

3.3.1 LLM-driven Sample Construction

We select a powerful large language model as a semantic discriminator to select semantically similar item pairs. Specifically, we first construct task-specific prompts based on the user-item interaction history. The prompt contains three parts: instruction, input, and output guidance. The instruction part defines a specific task and instructs the LLM to engage in role-playing. The input part is the target item text and the candidate items. The output guidance requires the LLM to generate content in the correct format. In the input part, given a user-item interaction history, we choose the title of the final item in the user-item interaction sequence as the target item text, and the same user's previously interacted items' titles as the candidates.

We design this prompt to activate the reasoning ability and open-world knowledge of LLM. The LLM selects the semantically most similar item from the candidate ones according to the text semantics of the target item and outputs the corresponding item-ID pair. Notice that LLM evaluates the semantic similarity between items based on raw text analysis, instead of on embedding-based retrieval. The selected item pair can be considered as having both semantic similarity and a collaborative co-occurrence pattern.

The prompt template is as follows.

Semantic Sample Construction Prompt

<Instruction>: You are a video similarity evaluation assistant. I will provide you with a target video title and a list of candidate video titles. Please help me find the most similar video title from the candidate list to the target video.

<Input>: The target video is *<target_itemID>* - *<target_item_title>*, and the candidate videos are: *<candidate_item_descriptions>*.

<Output Guidance>: Please find the most similar video title from the candidates and output the corresponding item-ID pair in the following format: *<target_itemID>*-*<similar_itemID>*.

If there are no similar videos, output (-1,-1) directly. Please ensure the format is correct; any other format will be considered invalid.

In this template, the special token *<target_itemID>*-*<target_item_title>* is replaced by the target item ID and its title.

The special token *<candidate_item_descriptions>* consists of several tokens like *<candidate_itemID>*-*<candidate_item_title>*, which is constructed from the same user's historically interacted items. We feed the complete prompt to an LLM and obtain a semantically most similar item-ID pair.

3.3.2 Supervised Fine-tuning of Joint Text and Image Encoders

Supervised fine-tuning facilitates a content encoder to generate better semantic representations for similar items in a downstream task and also align different content representations for a same item. Therefore, we define three alignment tasks, including text-to-text (t2t) alignment, image-to-image (i2i) alignment, and text-to-image (t2i) alignment.

Text-to-text (t2t) alignment: Given a batch of text pairs, we consider each text pair as a positive sample and target items' texts from other text pairs in the batch as negative samples. We pull text representations of positive samples and push apart the text representations of negative samples. We adopt the classical InfoNCE [49] loss,

$$\begin{aligned}\mathcal{L}_{t2t} &= -\frac{1}{N}\sum_{i=1}^{N}\log\left(\frac{\exp\left(B_{ii}^{t2t}\right)}{\exp\left(B_{ii}^{t2t}\right)+\sum_{j\neq i}\exp\left(B_{ij}^{t2t}\right)}\right),\\ B_{ii}^{t2t} &= sim(e_{i_p}^{text}, e_{i_q}^{text})/\tau,\\ B_{ij}^{t2t} &= sim(e_{i_p}^{text}, e_{j_q}^{text})/\tau,\end{aligned} \tag{2}$$

where N is the batch size, sim is the vector inner product operation, and $\tau \in [0, 1]$ is the temperature parameter. Note that $e_{i_p}^{text} \in \mathbb{R}^{1\times d}$ and $e_{i_q}^{text} \in \mathbb{R}^{1\times d}$ represent the i-th positive text representation pair. B_{ii}^{t2t} is the representation similarity of the positive text pair and $\sum_{i\neq j} B_{ij}^{t2t}$ is the sum of negative-instance similarities. Similarly, we have a loss for the image-to-image alignment $\mathcal{L}_{i2i}$.

Text-to-image (t2i) alignment: Inspired by the CLIP loss [50], we design a text-to-image (t2i) alignment task. We use the text and image representations of a same item as a positive sample pair, and those of different items as negative sample pairs. We again use the InfoNCE [49] loss,

$$\begin{aligned}\mathcal{L}_{t2i} &= -\frac{1}{N}\sum_{i=1}^{N}\log\left(\frac{\exp\left(B_{ii}^{t2i}\right)}{\exp\left(B_{ii}^{t2i}\right)+\sum_{jeqi}\exp\left(B_{ij}^{t2i}\right)}\right),\\ B_{ii}^{t2i} &= sim(e_{i}^{text}, e_{i}^{ima})/\tau,\\ B_{ij}^{t2i} &= sim(e_{i}^{text}, e_{j}^{ima}), /\tau\end{aligned} \tag{3}$$

where B_{ii}^{t2i} is the representation similarity of the text representations and image representations of a same item i, and $\sum_{i\neq j} B_{ij}^{t2i}$ is the sum of similarities of negative sample pairs.

Finally, we obtain the final supervised fine-tuning loss,

$$\mathcal{L}_{SFT} = \mathcal{L}_{t2t} + \mathcal{L}_{i2i} + \mathcal{L}_{t2i}. \tag{4}$$

We use $\mathcal{L}_{SFT}$ to jointly fine-tune the text encoder and image encoder for each downstream task, which achieves item-level modality representation alignment. Then, we use the tuned content encoder to encode each item's text and item image as semantic representations. These semantic representations would engage in sequence preference learning.

3.4 Sequence Preference Learning

User dynamic preferences can be captured by their previous interaction history, as well as by the content modality (e.g., title or image) of the items. Therefore, we consider two types of preferences in our model, i.e., behavior preferences on item IDs and content preferences.

Firstly, we construct an item-ID embedding matrix and an item-content embedding matrix via an embedding layer. Secondly, we unified the image representations and text representations as unified content representations via L2-normalization. Then, we develop a Transformer-based encoder-decoder sequence model, where an ID-modality sequence encoder captures user behavior preferences, a content-modality sequence encoder learns user content pref-

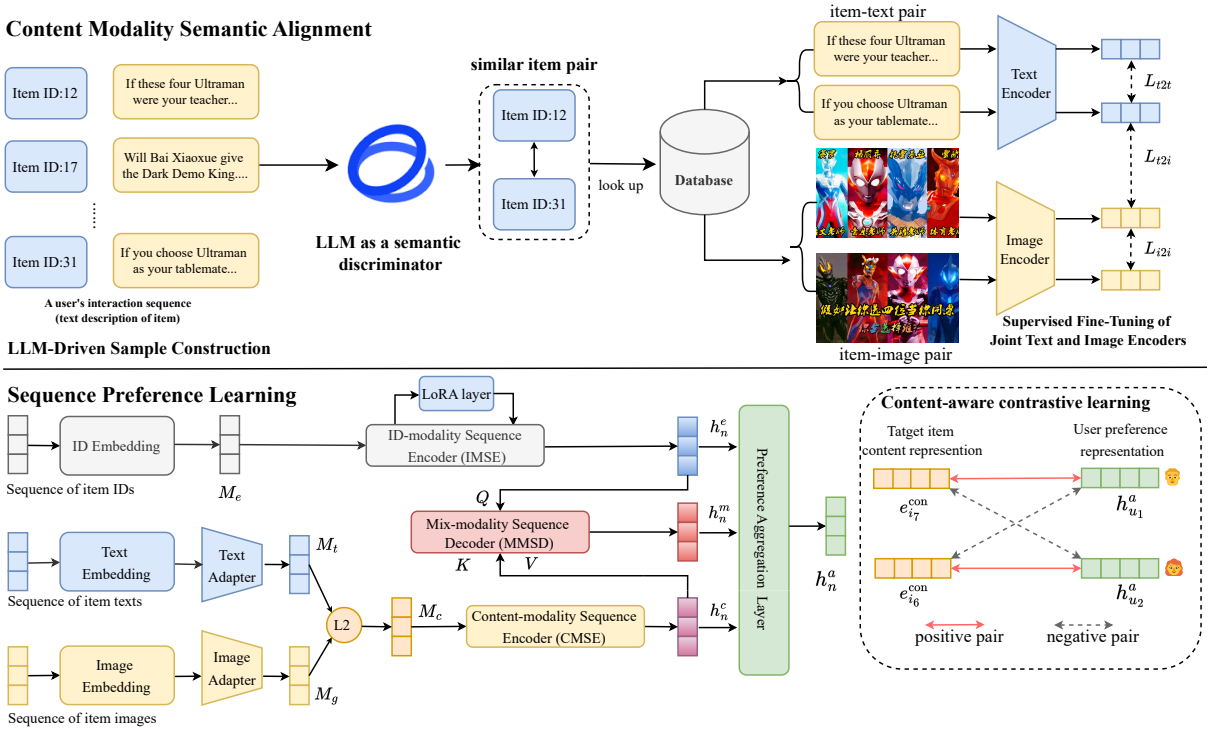

Fig. 1 The architecture of our SICSRec, including (i) content modality semantic alignment in Section 3.3 and (ii) sequence preference learning in Section 3.4. We propose a two-step training strategy with a content-aware contrastive learning task, which is described in Section 3.5.

erences, and a mix-modality sequence decoder grasps the intrinsic relationship between these two types of preferences. Finally, we aggregate these three outputs as the fine-grained user preferences.

3.4.1 Embedding Layer

We construct an item-ID embedding matrix $E_{id} \in \mathbb{R}^{|I|\times d}$, where d is the latent dimension of the vector. Given a user behavior sequence $S_{id} = \{i_1, i_2, ..., i_n\}$, we look up the item-ID embedding matrix and add a learnable position embedding matrix $P \in \mathbb{R}^{n\times d}$ to obtain an input item-ID embedding matrix,

$$\begin{aligned} M_e &= \text{Emb}(S_{id}) \\ &= \left\{e_1^{id} + p_1, \ldots, e_j^{id} + p_j, \ldots, e_n^{id} + p_n\right\}, \end{aligned} \tag{5}$$

where $e_j^{id} \in \mathbb{R}^{1\times d}$ and $p_j^{id} \in \mathbb{R}^{1\times d}$ are the corresponding j-th item-ID embedding vector and position embedding vector.

We then utilize the fine-tuned text encoder (i.e., BERT [51]) and image encoder (i.e., Swin-base [52]) in Section 3.3, to obtain an item-text embedding matrix $E_{text} \in \mathbb{R}^{|I|\times d_{text}}$, and an item-image embedding matrix $E_{ima} \in \mathbb{R}^{|I|\times d_{ima}}$, where d_{text} and d_{ima} are the output dimensions. Given an item-text sequence $S_{text} = \{t_1, t_2, ..., t_n\}$ and an item-image sequence $S_{ima} = \{g_1, g_2, ..., g_n\}$, we firstly encode them by looking up the corresponding embedding matrix and then transform their dimension to match the ID embedding via an adapter layer, which is implemented by a multi-layer perceptron layer. Secondly, we add the shared learnable position embedding matrix $P \in \mathbb{R}^{n\times d}$ for content input embedding

matrix to learn temporal dependency,

$$\begin{aligned} M_t &= Adapter(Emb(S_{text})) \\ &= \{e_1^{text} + p_1, \ldots, e_n^{text} + p_n\}, \end{aligned} \tag{6}$$

$$\begin{aligned} M_g &= Adapter(Emb(S_{ima})) \\ &= \{e_1^{ima} + p_1, \ldots, e_n^{ima} + p_n\}, \end{aligned} \tag{7}$$

where $e_j^{text} \in \mathbb{R}^{1\times d}$ and $e_j^{image} \in \mathbb{R}^{1\times d}$ are the corresponding j-th text embedding vector and image embedding vector.

3.4.2 ID-modality Sequence Encoder

The user behavior interaction sequences contain important user preference signals. We apply the directional Transformer model SASRec [2] as an ID sequence encoder, which can be replaced by some other ID-based sequential models. Given the input item-ID embedding matrix $M_e \in \mathbb{R}^{n\times d}$, we first transform it into query, key, and value by linear projections, and then feed them to a self-attention module [53],

$$\begin{aligned} \hat{H}_e &= Attention(M_e, M_e, M_e) \\ &= Softmax\left(\frac{(M_e W_e^Q)(M_e W_e^K)^T}{\sqrt{d}}\right)(M_e W_e^V), \end{aligned} \tag{8}$$

where $W_e^Q, W_e^K, W_e^V \in \mathbb{R}^{d\times d}$ are learnable linear projection matrices. After that, we conduct a point-wise feed-forward network to get the hidden behavior matrix,

$$\begin{aligned} H_e &= FFN(\hat{H}_e) \\ &= ReLU(\hat{H}_e W_1^e + b_1^e)W_2^e + b_2^e, \end{aligned} \tag{9}$$

where $W_1^e, W_2^e \in \mathbb{R}^{d\times d}$ are learnable weight matrices, and $b_1^e, b_2^e \in \mathbb{R}^{1\times d}$ are learnable bias vectors. $H_e = \{h_1^e, h_2^e, ..., h_n^e\} \in \mathbb{R}^{n\times d}$ is the output hidden behavior preference sequence. We use the last-step output h_n^e as the user behavior preferences.

3.4.3 Content-modality Sequence Encoder

The item content information like text and image contains rich semantic signals. We aim to capture a user's content-based preferences from the item-content sequence. Given an item-text embedding matrix $M_t \in \mathbb{R}^{n\times d}$ and an item-image embedding matrix $M_g \in \mathbb{R}^{n\times d}$, we use an L2 normalization to combine the item-text and item-image representations in each position as a unified content one [54],

$$e_i^{cont} = \frac{e_i^{text} + e_i^{ima}}{\|e_i^{text} + e_i^{ima}\|}. \tag{10}$$

Therefore, we obtain a unified content embedding sequence $M_c = \{e_1^{cont}, e_2^{cont}, ..., e_n^{cont}\} \in \mathbb{R}^{n\times d}$. The L2 normalization term has several advantages. Firstly, the unified content representations of the text modality and image modality improve rich semantic understanding and address the limitations inherent in uni-modal representations. For example, when the image modality lacks contextual information, the text modality effectively supplements it with semantic content. Secondly, it only needs a content encoder to model unified content representations rather than two individual encoders to model two types of content representations, thereby reducing the complexity of the preference model.

Then, similar to the ID-modality sequence encoder, we utilize a directional Transformer model to learn the content preferences. Firstly, we feed the content input matrix M_c into a self-attention module,

$$\begin{aligned} \hat{H}_c &= Attention(M_c, M_c, M_c) \\ &= Softmax\left(\frac{(M_c W_c^Q)(M_c W_c^K)^T}{\sqrt{d}}\right)(M_c W_c^V), \end{aligned} \tag{11}$$

where $W_c^Q, W_c^K, W_c^V \in \mathbb{R}^{d\times d}$ are learnable linear projection matrices. Then we feed $\hat{H}_c$ into the feed-

forward network,

$$\begin{aligned} H_c &= FFN(\hat{H}_c) \\ &= ReLU(\hat{H}_c W_1^c + b_1^c)W_2^c + b_2^c, \end{aligned} \tag{12}$$

where $W_1^c, W_2^c \in \mathbb{R}^{d\times d}$ are learnable weight matrices, and $b_1^c, b_2^c \in \mathbb{R}^{1\times d}$ are learnable bias vectors. $H_c = \{h_1^c, h_2^c, ..., h_n^c\} \in \mathbb{R}^{n\times d}$ is the output hidden content preference sequence. We use the last-step output h_n^c as the user content preferences.

3.4.4 Mix-modality Sequence Decoder

Previous encoders often independently model the ID sequences and content sequences and do not consider their inherent correlations. Therefore, we introduce a mix-modality sequence decoder to grasp the ID modality and content modality relations for learning better sequence representations.

Given a behavior preference sequence $H_e \in \mathbb{R}^{n\times d}$ from Section 3.4.2 and a content preference sequence $H_c \in \mathbb{R}^{n\times d}$ from Section 3.4.3, we adopt H_e as the query, and H_c as the key and value in the cross-attention mechanism. The attention map in cross-attention captures the relationship between the behavior preferences H_e and the content preferences H_c. Then, this attention map would be combined with the original content preferences H_c to generate the output $\hat{H}_m$. The formulation is as follows,

$$\begin{aligned} \hat{H}_m &= Attention(H_e, H_c, H_c) \\ &= Softmax\left(\frac{(H_e W_m^Q)(H_c W_m^K)^T}{\sqrt{d}}\right)(H_c W_m^V), \end{aligned} \tag{13}$$

where $W_m^Q, W_m^K, W_m^V \in \mathbb{R}^{d\times d}$ are learnable linear projection matrices. Similarly, we feed $\hat{H}_m$ into a feed-forward layer to obtain the mix-modality preferences,

$$\begin{aligned} H_m &= FFN(\hat{H}_m) \\ &= ReLU(\hat{H}_c W_1^m + b_1^m)W_2^m + b_2^m, \end{aligned} \tag{14}$$

where $W_1^m, W_2^m \in \mathbb{R}^{d\times d}$ are learnable weight matrices, and $b_1^m, b_2^m \in \mathbb{R}^{1\times d}$ are learnable bias vectors. $H_m = \{h_1^m, h_2^m, ..., h_n^m\} \in \mathbb{R}^{n\times d}$ is the output hidden preference sequence. We use the last-step output h_n^m as the mix-modality preferences.

3.4.5 Preference Aggregation

We aggregate the behavior preferences, content preferences, and mix-modality preferences, and obtain the final preference representations by a linear projection,

$$h_n^a = Projection(Concat(h_n^e, h_n^c, h_n^m)), \tag{15}$$

where $h_n^a \in \mathbb{R}^{1\times d}$ is the user's final sequential preferences.

3.5 A Two-step Training Strategy

Training a sequence model from scratch is often inefficient and costly, because the content representations and ID representations may interfere with each other, leading to difficulty in model convergence and unstable performance. Therefore, we propose a two-step training strategy, i.e., (i) pre-train the ID modality sequence encoder with a standard cross-entropy loss and fix its weight; and (ii) post-train the content sequence encoder and mix-modality sequence decoder. We propose a content-aware contrastive learning to align the content modality representations and the ID modality representations, and utilize a low-rank adaptation layer to extract the user behavior preferences. In this setting, we decouple the training process of the content-modality dependency and the item-collaborative dependency.

3.5.1 Next-item Prediction Task

In the first step, we utilize the next-item prediction task to train an ID-modality encoder and fix

its weights in the second step.

The prediction score for item i as the next preferred item can be estimated as,

$$\hat{y}_i = h_n^a (e_i^{id})^T, \tag{16}$$

where $\hat{y}_i$ is the predicted score of item i and e_i^{id} is the i-th item-ID embedding vector. We adopt the cross-entropy loss function to measure the difference between the prediction $\hat{y}$ and the ground truth y,

$$\mathcal{L}_{CE} = -\sum_{i=1}^{|I|} y_i \log(\widehat{y_i}). \tag{17}$$

3.5.2 Content-aware Contrastive Learning Task

We introduce a content-aware contrastive learning task to align user preferences with content modality representations for sequence-level modality representation alignment. Given a batch of B training instances $\{< h_1^a, e_1^c >, < h_2^a, e_2^c >, ..., < h_B^a, e_B^c >\}$, where the j-th training instance is a pair of aggregated sequential preference representations h_j^a and content representations e_j^c. We define the contrastive learning loss as follows,

$$\mathcal{L}_{ConCL} = -\frac{1}{B}\sum_{i=1}^{B} \log\left(\frac{\exp(sim(h_i^a, e_i^c)/\tau)}{\sum_{i'=1}^{B}\exp(sim(h_i^a, e_{i'}^c)/\tau)}\right), \tag{18}$$

where in-batch negative instances $\{e_{i'}^c\}$ are the content embeddings of the positive items of another sequences and $\tau \in [0, 1]$ is the temperature parameter.

3.5.3 Low-rank Adaptation Layer

The final loss function can be formulated as,

$$\mathcal{L} = \mathcal{L}_{CE} + \alpha\mathcal{L}_{ConCL} + \lambda\|\mathbf{\Theta}\|_F, \tag{19}$$

where $\mathbf{\Theta} = \{E_{id}, E_{text}, E_{image}\}$ and $\|\cdot\|_F$ denotes the L2 normalization. Note that $\alpha \in [0, 1]$ is the balance parameter and λ is the regularization parameter.

Inspired by the low-rank adaptation method [55], we adopt a LoRA layer to fine-tune the ID-modality sequence encoder and post-train other components with the final loss $\mathcal{L}$ in end-to-end training.

Given the fixed parameter weight $W_0 \in R^{d\times k}$ of the ID-modality sequence encoder, we represent its updating process as follows,

$$W_0 + \Delta W = W_0 + BA, \tag{20}$$

where $\Delta W = BA$ is the updating weight, and $B \in R^{d\times r}, A \in R^{r\times k}$ are learnable lightweight matrices. Note that $r << min(d, k)$ is the rank of the matrix. We initialize matrix A with a zero-mean normal distribution and matrix B with zeros.

3.6 Analysis of Time Complexity

In content modality semantic alignment, we use LLM-driven semantic discriminator to select a sample data, and fine-tune the text encoder and image encoder jointly based on the sample data. Then, we encode the text embedding and the image embedding matrix via these tuned encoders, which can be cached in advance. Therefore, there is no need to invoke LLM and content encoders in sequence preference learning, which reduces the extra inference costs.

In sequence preference learning, we adopt a two-step training strategy to train our model. In the prediction stage, we remove the content-aware contrastive learning task and infer user preferences based on the encoder-decoder model.

We assume each LLM inference takes time t and there are k samples. We show the time complexity of our SICSRec in Table 2, where b, n, and d denote the batch size, input sequence length, and hidden dimension, respectively. Our SICSRec has the

same level of time complexity with Transformer-based sequential recommendation methods.

Table 2 Time complexity analysis of our SICSRec.

Component	Time Complexity
LLM inference	$O(t \cdot k)$
$\mathcal{L}_{SFT}$	$O(b^2 \cdot d)$
IMSE	$O(b \cdot (n^2 \cdot d + n \cdot d^2)$
CMSE	$O(b \cdot (n^2 \cdot d + n \cdot d^2)$
MMSE	$O(b \cdot (n^2 \cdot d + n \cdot d^2)$
$\mathcal{L}_{ConCL}$	$O(b^2 \cdot d)$
Training	$O(b \cdot (n^2 \cdot d + n \cdot d^2) + b^2 \cdot d)$
Inference	$O(b \cdot (n^2 \cdot d + n \cdot d^2))$

4 Experiments

In this section, we conduct extensive experiments to answer the following five research questions:

- **RQ1**: How does our SICSRec perform against the state-of-the-art ID-based and content-based SR methods? (see Sec. 4.4)
- **RQ2:** What is the impact of supervised fine-tuning of the content encoder on our SICSRec? (see Sec. 4.5)
- **RQ3:** What are the effects of different components in our SICSRec? (see Sec. 4.6)
- **RQ4:** Does the two-stage training strategy contribute to preference learning in our SICSRec? (see Sec. 4.7)
- **RQ5:** How do the hyper-parameters affect the performance of our SICSRec? (see Sec. 4.8)
- **RQ6:** How is the inference efficiency of our SICSRec? (see Sec. 4.9)
- **RQ7:** How can we visually demonstrate the impact of model modifications in our SICSRec? (see Sec. 4.10)

Table 3 Statistical details of the datasets.

Dataset	#Users	#Items	#Interactions	Sparsity
Cartoon	30,300	4,724	215,443	99.88%
Dance	10,715	2,307	83,392	99.66%
Food	6,549	1,579	39,740	99.62%
Movie	16,525	3,509	115,576	99.80%

4.1 Datasets and Evaluation Metrics

We use four datasets from NinRec [56], which is a large-scale multi-modality benchmark collected from an online video platform BiliBili with different scenarios. Each item in these datasets is associated with an item ID, a piece of descriptive text, and a high-resolution cover image. We show the statistical details of the datasets in Table 3. We adopt the leave-one-out strategy to split each dataset, which uses the last item for test, the penultimate one for validation, and the other items for training. We use the full-ranking strategy for a fair comparison [57]. Two commonly used metrics, i.e., Hit@K and NDCG@K are used for evaluation, where $K \in \{5, 10\}$.

4.2 Baselines

(1) Item ID-based sequential recommenders

- GRU4Rec [1]: A session-based recommendation method, which models user behavior sequences via a GRU network.
- SASRec [2]: A self-attentive sequential recommendation method, which models behavior sequences via a left-right unidirectional Transformer.
- BERT4Rec [3]: A bidirectional Transformer-based sequential recommender with a mask-item modeling task.

(2) Item content-based sequential recommenders

Table 4 Recommendation performance comparison of eleven baselines and our SICSRec on the four datasets. Bold scores represent the best performance, while underlined scores indicate the second-best performance. "*T*", "*V*" and "*ID*" stand for text, image, and ID. "Improved" denotes the relative improvement of our SICSRec compared with the second-best method.

Input Type & Model →		ID			T			T+ID			T+V+ID			Improved
Dataset	**Metric**	**GRU4Rec**	**SASRec**	**BERT4Rec**	**UniSRec**	**VQ-Rec**	**MISSRec**	**LLMESR**	**UniSRec**	**MISSRec**	**LLMESR**	**MISSRec**	**SICSRec**	
Cartoon	Hit@5	0.0678	0.0845	0.0354	0.0754	0.0470	0.0465	0.0713	0.0801	0.0856	0.0716	0.0730	**0.0875**	2.22%
	Hit@10	0.1074	0.1318	0.0628	0.1293	0.0909	0.0683	0.1120	0.1327	0.1296	0.1130	0.1166	**0.1344**	1.28%
	NDGC@5	0.0431	0.0474	0.0213	0.0432	0.0280	0.0241	0.0400	0.0464	0.0454	0.0415	0.0430	**0.0531**	12.03%
	NDCG@10	0.0558	0.0626	0.0300	0.0607	0.0421	0.0308	0.0531	0.0634	0.0589	0.0548	0.0564	**0.0682**	7.57%
Dance	Hit@5	0.1395	0.1489	0.0914	0.1387	0.1029	0.0914	0.1379	0.1420	0.1266	0.1392	0.1535	**0.1578**	2.80%
	Hit@10	0.2067	0.2251	0.1495	0.2155	0.1670	0.1409	0.2088	0.2150	0.1876	0.2102	0.2248	**0.2295**	1.95%
	NDGC@5	0.0928	0.0933	0.0583	0.0883	0.0660	0.0505	0.0877	0.0897	0.0696	0.0887	0.0965	**0.1074**	11.30%
	NDCG@10	0.1143	0.1179	0.0770	0.1131	0.0866	0.0657	0.1106	0.1133	0.0883	0.1115	0.1184	**0.1304**	10.14%
Food	Hit@5	0.0866	0.1356	0.0437	0.0556	0.0779	0.0739	0.1244	0.0999	0.1283	0.1266	**0.1448**	0.1367	-5.59%
	Hit@10	0.1376	0.1993	0.0765	0.1026	0.1460	0.1222	0.1849	0.1587	0.1855	0.1881	0.2023	**0.2037**	0.69%
	NDCG@5	0.0561	0.0758	0.0265	0.0332	0.0440	0.0388	0.0711	0.0582	0.0631	0.0719	0.0837	**0.0849**	1.43%
	NDCG@10	0.0724	0.0964	0.0370	0.0481	0.0658	0.0535	0.0906	0.0771	0.0807	0.0917	0.1013	**0.1064**	5.03%
Movie	Hit@5	0.0618	0.0784	0.0378	0.0761	0.0545	0.0415	0.0694	0.0765	0.0687	0.0734	0.0764	**0.0810**	3.32%
	Hit@10	0.0966	0.1247	0.0623	0.1229	0.0953	0.0631	0.1102	0.1248	0.1107	0.1105	0.1155	**0.1260**	0.96%
	NDCG@5	0.0404	0.0434	0.0226	0.0445	0.0332	0.0236	0.0383	0.0461	0.0369	0.0412	0.0446	**0.0495**	7.38%
	NDCG@10	0.0516	0.0583	0.0304	0.0595	0.0464	0.0301	0.0515	0.0617	0.0498	0.0532	0.0565	**0.0640**	3.73%

- UniSRec(T) [14]: A text modality-based sequential recommendation method that utilizes multi-domain text to learn universal item representations.
- VQ-Rec [15]: A vector-quantized sequential recommendation method that designs text representations as multiple code representations.
- MISSRec(T) [16]: A text-modality-based sequential recommendation method that utilizes text to model user intents.

(3) Item ID and content-based sequential recommenders

- UniSRec(T+ID) [14]: An improved version of UniSRec, which fine-tunes item embedding for downstream recommendation tasks.
- MISSRec(T+ID) [16]: An improved version of MISSRec, combining item embedding and text embedding for user preference modeling.
- LLMESR(T+ID) [58]: A dual-view sequential recommendation method that combines semantic embeddings from a large language model and behavior embeddings from user-item interactions.
- LLMESR++(T+V+ID) [58]: An improved version of LLMESR, which combines text embeddings and image embeddings as the input of the cross-attention mechanism.
- MISSRec(T+V+ID) [16]: An improved version of MISSRec, utilizing text, image, and item ID to model user preferences.

4.3 Implementation Details

For a fair comparison, we implement our SICSRec and all the baselines by RecBole [59]. The codes of GRU4Rec, SASRec, and BERT4Rec come from the RecBole platform. Moreover, we use the public codes of UniSRec[1)], VQ-Rec[2)], LLMESR[3)] and MISSRec[4)]. We publish the datasets and source code of our SICSRec[5)]. The latent dimension d is tuned from {64, 128, 256}, and d=256 yields the best performance. Following the setting of previous works [2, 3], the maximum sequence length is

[1)]https://github.com/RUCAIBox/UniSRec
[2)]https://github.com/RUCAIBox/VQ-Rec
[3)]https://github.com/liuqidong07/LLM-ESR
[4)]https://github.com/gimpong/MM23-MISSRec
[5)]https://github.com/donglinzhou/SICSRec

50. The temperature τ is 0.05 and the dropout rate is 0.5. The training batch size is 128 in Stage 1 and 1024 in Stage 2. We use early stopping with the patience of 10 epochs to prevent overfitting. The balance parameter α ranges from 0.1 to 1.0 with a step of 0.1. The regularization parameter λ is chosen from {0.001,0.0001}. The rank r is chosen from {4, 8, 12, 16}. We use the Adam optimizer with a learning rate of 1e-3. We search the hyperparameters of all the compared methods using validation data.

For sample construction, we adopt different LLMs as semantic discriminators and compare their performance, including GLM-4-9B-Chat [60], Hunyuan [61], Qwen1.5-14B-chat [62], and DeepSeek [63]. We use the text encoder from BERT [51], RoBERTa [64], and Sentence-T5 [65]. We use the image encoder from Swin-base [52], ViT [66], and Resnet50 [67]. The original text embedding size is 768 and the image embedding size is 1000. We conduct experiments on a Tesla V100-PCIe GPU with 32GB memory.

4.4 Overall Performance Comparison (RQ1)

We report the experimental results in Table 4 and have the following observations.

- Item ID-based sequential recommenders still achieve good performance on content-modality recommendation scenarios. For example, SASRec outperforms other baselines on Hit@5 on Movie and Hit@10 on Dance. GRU4Rec and BERT4Rec do not perform well. Previous works have also shown that the performance of BERT4Rec does not surpass SASRec under the full-ranking evaluation setting [27, 32, 68].
- Text modality-only sequential recommenders (i.e., VQ-Rec, UniSRec(T), and MISSRec(T)) do not surpass the strong item-ID-based sequential recommenders (i.e., SASRec). Importantly, combining the text modality and ID modality can achieve a similar performance compared with the ID modality-based methods, and surpass their text modality-only versions, showing that ID modality is still an important feature in recommender systems.

- Jointly considering the image and text modalities always improves the recommendation performance. MISSRec (T+V+ID) achieves the second-best performance on Dance and Food and UniSRec achieves the second-best performance on Cartoon and Movie, showing that using more content information enhances representation learning.

- Our SICSRec achieves the best performance on almost all the datasets, with an average improvement of 8.04% on NDCG@5 and 6.62% on NDCG@10 compared with the best performing baseline. On the Food dataset, our SICSRec achieves the second-best performance on Hit@5 and remains superior on other metrics, likely due to its small scale. As shown in Table 3, the Food dataset is relatively small, with only 6,549 users and 1,579 items. It is not difficult for a typical deep learning-based model to handle such a dataset. Therefore, the baseline models like MISSRec and SASRec achieve strong performance, while our SICSRec only shows marginal improvements. Moreover, using RoBERTa&ViT as the modality encoder further improves the performance of our SICSRec on Food, which can be observed from Table 6.

4.5 Supervised Fine-tuning Analyses (RQ2)

In this subsection, we examine the effectiveness of supervised fine-tuning and report the results in Figure 2.

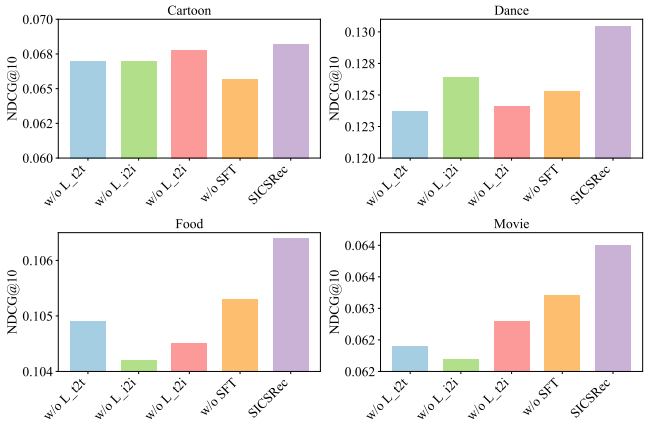

Fig. 2 Supervised fine-tuning analyses. "t2t", "i2i", and "t2i" stand for text-to-text alignment, image-to-image alignment, and text-to-image alignment, respectively. "SFT" refers to the combination of these three alignment tasks.

We have the following observations: (i) Supervised fine-tuning (SFT) of the text encoder and image encoder improves content representation learning and reduces the item-level semantic gaps. (ii) Removing any alignment task decreases the performance. (iii) The text-to-text alignment task plays a key role on Cartoon and Dance. Without the image-to-image alignment task, our SICSRec obtains the greatest drop in performance on Food and Movie.

Furthermore, we study using different LLMs as the semantic discriminator and compare their performance. The results are shown in Table 5. Hunyuan achieves the best Hit@10 performance on Dance while GLM achieves the best performance on Cartoon and Movie. Qwen gains the best performance on Food. DeepSeek also achieves comparable performance to that of GLM on most datasets. Different large language models will generate different samples, which may influence the fine-tuned content encoder, but their performance remains similar.

Table 5 Recommendation performance with different LLMs as semantic discriminators. H@10 and N@10 stand for Hit@10 and NDCG@10, respectively.

Combination	Cartoonn		Dance		Food		Movie	
	H@10	N@10	H@10	N@10	H@10	N@10	H@10	N@10
Hunyuan	0.1337	0.0676	**0.2324**	0.1303	0.2011	0.1058	0.1254	0.0638
Qwen	0.1343	0.0674	0.2316	0.1290	**0.2054**	**0.1077**	0.1253	0.0638
DeepSeek	0.1340	0.0679	0.2301	0.1284	0.2032	0.1070	0.1248	0.0638
GLM	**0.1344**	**0.0682**	0.2295	**0.1304**	0.2037	0.1064	**0.1260**	**0.0640**

Finally, we study how different combinations of a text encoder and an image encoder affect the recommendation performance. Table 6 shows that the combination of BERT and Swin outperforms others on most datasets. RoBERTa&ViT achieves the best NDCG@10 performance on Food and Movie, while Sentence-T5&Resnet50 gains the worst results, which indicates that the combination of content encoders has a significant impact on the recommendation performance. We fix the text encoder (i.e., BERT) and replace the visual encoders. The results show that Swin performs better Hit@10 on Cartoon and Food, while ViT achieves higher performance on Dance and Movie. Overall, Transformer-based visual encoders (e.g., Swin and ViT) exhibit superior representation ability compared with ResNet-based architectures. Swin introduces a shifted window mechanism, enabling hierarchical feature extraction beyond ViT, so it may perform well in some scenarios. We also fix the visual encoder (i.e., Swin), and replace the text encoders. We find that BERT and RoBERT achieve comparable performance on most datasets, while Sentence-T5 gains lower results. This may be because encoder-only text encoders (e.g., BERT and RoBERTa) are generally more suitable for semantic understanding tasks than encoder-decoder models (e.g., Sentence-T5).

Table 6 Recommendation performance with different combinations of text and image encoders. H@10 and N@10 stand for Hit@10 and NDCG@10, respectively.

Combination	Cartoonn		Dance		Food		Movie	
	H@10	N@10	H@10	N@10	H@10	N@10	H@10	N@10
BERT&Swin	**0.1344**	**0.0682**	0.2295	**0.1304**	0.2037	0.1064	**0.1260**	0.0640
RoBERTa&ViT	0.1324	0.0670	**0.2304**	0.1283	**0.2042**	**0.1076**	0.1256	**0.0643**
Sentence-T5&Resnet50	0.1322	0.0674	0.2285	0.1252	0.2014	0.1061	0.1245	0.0634
BERT&Swin	**0.1344**	**0.0682**	0.2295	**0.1304**	**0.2037**	0.1064	0.1260	**0.0640**
BERT&ViT	0.1336	0.0671	**0.2315**	0.1283	0.2019	**0.1071**	**0.1264**	0.0638
BERT&ResNet50	0.1317	0.0664	0.2249	0.1256	0.1987	0.1052	0.1241	0.0635
BERT&Swin	**0.1344**	**0.0682**	0.2295	**0.1304**	**0.2037**	0.1064	**0.1260**	**0.0640**
RoBERTa&Swin	0.1325	0.0674	0.2301	0.1283	0.2032	**0.1077**	0.1254	0.0638
Sentence-T5&Swin	0.1321	0.0679	**0.2305**	0.1280	0.2016	0.1071	0.1250	0.0638

4.6 Ablation Study (RQ3)

We study how each component affects the recommendation performance and report the ablation study results in Table 7. Specifically, we study the contributions of the ID-modality sequence encoder (IMSE), the content-modality sequence encoder (CMSE), the mix-modality sequence decoder (MMSD), content-aware contrastive learning ($\mathcal{L}_{ConCL}$), the LoRA layer, and the L2-norm for CMSE.

Table 7 Recommendation performance (NDCG@10) in the ablation study.

Variants	Cartoon	Dance	Food	Movie
w/o IMSE	0.0622	0.1192	0.0977	0.0601
w/o CMSE	0.0670	0.1260	0.1050	0.0625
w/o MMSD	0.0665	0.1255	0.1046	0.0627
w/o $\mathcal{L}_{ConCL}$	0.0658	0.1222	0.1051	0.0604
w/o LoRA layer	0.0667	0.1246	0.1062	0.0617
w/o L2-norm	0.0670	0.1252	0.1047	0.0634
Our SICSRec	**0.0682**	**0.1304**	**0.1064**	**0.0640**

We have the following observations:

- The ID-modality sequence encoder is the most important component in our SICSRec. Removing it achieves the lowest performance on all datasets.
- The content-modality sequence encoder and the mix-modality sequence decoder contribute to the performance of our SICSRec, because they learn the users' content preferences, and the inherent correlations between behavior preferences and content preferences.
- Removing content-aware contrastive learning ($\mathcal{L}_{ConCL}$) decreases the performance because it contributes to the sequence-level representation alignment between user preferences and content representations.
- Without the LoRA layer in the ID-modality sequence encoder, our SICSRec suffers from inadequate behavior preference learning.
- We find that removing the L2 normalization term for CMSE and using the two encoders to model the text sequence and the image sequence would decrease the performance because the content modality gap may interfere with representation learning.

Table 8 Recommendation performance (NDCG@10) with different modality information.

Input Type	Variant	Cartoon	Dance	Food	Movie
T	UniSRec	**0.0607**	0.1131	0.0481	**0.0595**
	MISSRec	0.0308	0.0657	0.0535	0.0301
	SICSRec	0.0591	**0.1167**	**0.0944**	0.0546
V	MISSRec	0.0378	0.0931	0.0815	0.0429
	SICSRec	**0.0621**	**0.1216**	**0.0979**	**0.0592**
T+V	MISSRec	0.0451	0.0983	0.0882	0.0466
	SICSRec	**0.0653**	**0.1298**	**0.1092**	**0.0605**
ID+T	UniSRec	0.0634	0.1133	0.0771	0.0617
	MISSRec	0.0589	0.0883	0.0807	0.0498
	SICSRec	**0.0662**	**0.1217**	**0.1010**	**0.0624**
ID+V	MISSRec	0.0625	0.1029	0.0828	0.0487
	SICSRec	**0.0665**	**0.1243**	**0.1056**	**0.0634**
ID+T+V	MISSRec	0.0564	0.1184	0.1013	0.0565
	SICSRec	**0.0682**	**0.1304**	**0.1064**	**0.0640**

Secondly, we investigate the capability of our SICSRec and other content-based sequential models in leveraging multi-modal information. The results are reported in Table 8.

We have the following observations: (i) Removing the ID modality and only modeling user content preferences results in the worst performance,

demonstrating that the ID modality plays a significant role. (ii) Our SICSRec has a large advantage over MISSRec when only the T or V modality is used. The performance gap narrows when both the T and V modalities are utilized, which indicates that considering more content modality information always enhances the recommendation performance. (iii) Our SICSRec outperforms other content-modality models in most cases, showing that our SICSRec is robust and competitive.

4.7 Training Strategy of SICSRec (RQ4)

We compare our two-step training strategy with different training strategies. Table 9 shows their performance and the number of epochs required for model convergence. Note that the number of epochs may vary due to different random seeds and hardware.

Table 9 Recommendation performance with different training strategies. N@10 and #epo stand for NDCG@10 and the number of epochs, respectively. "Fixed" means fixing the model weight of this component.

Strategy	Cartoonn		Dance		Food		Movie	
	N@10	#epo	N@10	#epo	N@0	#epo	N@10	#epo
Fixed IDEnc	**0.0682**	17	**0.1304**	51	**0.1064**	17	**0.0640**	26
Fixed IDEmb	0.0653	34	0.1263	24	0.1056	18	0.0618	13
Fixed IDEmb&Enc	0.0656	17	0.1254	22	0.1059	17	0.0608	17
Fixed ConEnc	0.0470	76	0.1075	60	0.0610	135	0.0485	86
Fixed ConEmb	0.0476	110	0.1164	104	0.0644	116	0.0457	99
Fixed ConEmb&Enc	0.0496	132	0.1084	64	0.0605	121	0.0513	118
End2end(not fixed)	0.0449	75	0.1102	51	0.0586	79	0.0473	97

The observations are as follows. (i) Training our SICSRec from scratch (end-to-end training) results in low efficiency and unstable performance, because the content representations and the ID representations may interfere with each other, leading to difficulty in model convergence. (ii) Pre-training content-modality-related components firstly requires more training epochs and achieves sub-optimal performance. (iii) Pre-training ID modality-related components and post-training other components (i.e., our two-step representation learning) achieve stable performance and require fewer training epochs for model convergence.

4.8 Parameter Analyses (RQ5)

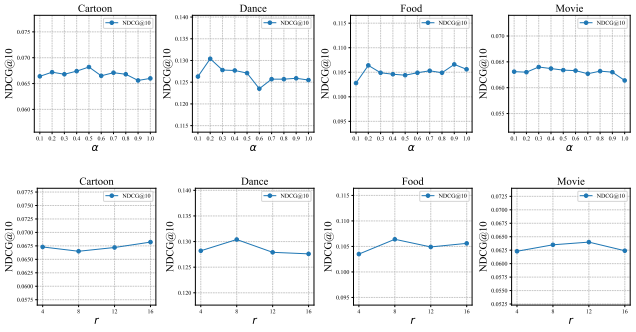

Fig. 3 Recommendation performance of our SICSRec with different values of the balance parameter α, and rank r.

In this experiment, we aim to investigate the effect of two hyper-parameters, i.e., the balance parameter α for $\mathcal{L}_{ConCL}$ and the LoRA rank r. Figure 3 shows that the contrastive learning ratio significantly impacts the performance of our SICSRec. Changing the value of r does not greatly increase the performance, and setting a small rank for tuning the ID-modality sequence encoder is enough.

4.9 Inference Efficiency (RQ6)

We study the inference efficiency of SASRec, MISSRec, and our SICSRec. We conduct experiments on a single NVIDIA Tesla V100-PCIe GPU with 32 GB memory. We load the pre-trained model weights into GPU memory and measure the forward pass latency, excluding data loading and preprocessing time. Figure 4 shows that SASRec achieves the fastest inference speed, and our SICSRec demonstrates competitive efficiency by outperforming MISSRec. The inference efficiency is acceptable for real-world recommendation.

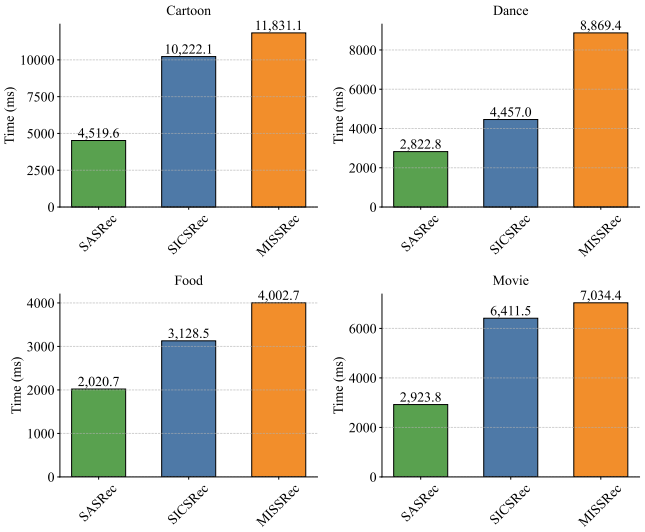

Fig. 4 Inference time of SASRec, MISSRec, and our SICSRec.

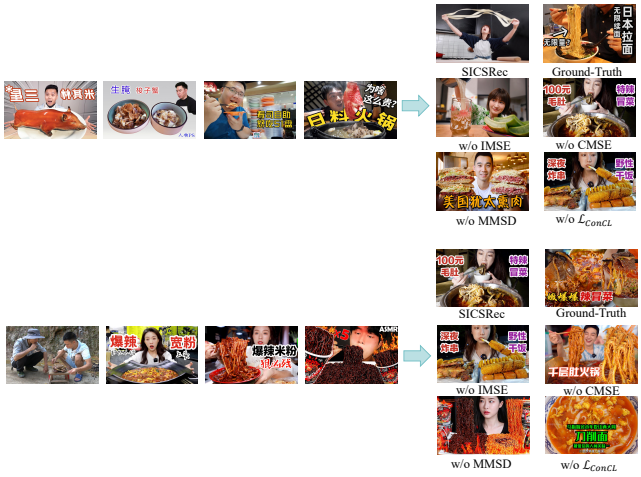

Fig. 5 Case studies of our SICSRec with different model modifications (left: historical preferred items; right: recommended items).

4.10 Case Study (RQ7)

We choose the Food scenario and conduct a case analysis on model modifications. We use the same user interaction sequence as input, load different pre-trained variant model weights to predict the next item, and compare the recommended item with the ground-truth.

From Figure 5, we can observe that our SICSRec recommends more accurate items than other variants by effectively combining visual, textual, and behavioral information. For example, in the first case, the user interaction history shows a stronger interest in Japanese food, and our SICSRec recommends a Japanese noodle dish, while other baseline models only recommend fast food.

5 Conclusions and Future Work

In this paper, we propose a novel self-supervised sequential representation learning method named SICSRec. Firstly, we propose a novel content modality semantic alignment module to reduce the item-level semantic gap between different modality representations. Then, we propose a novel Transformer-based encoder-decoder model to learn users' sequential preferences. Finally, we propose a two-step training strategy to train our model. We conduct extensive experiments and ablation studies to study the effectiveness of our SICSRec on four datasets and find that our model is very competitive compared with the state-of-the-art methods.

In the future, we are interested in exploring some self-adaptive modality fusion methods for sequential recommendation with rich side information. Moreover, we plan to extend our solution to handle the cold-start problem with new users and new items.

Acknowledgements We thank the support of Guangdong Basic and Applied Basic Research Foundation (No.2024A1515010122) and National Natural Science Foundation of China (Nos. 62461160311 and 62272315).

References

1. Hiedasi B, Karatzoglou A, Baltrunas L, Tikk D. Session-based recommendations with recurrent neural networks. arXiv preprint arXiv:1511.06939, 2016
2. Kang W C, McAuley J. Self-attentive sequential recommendation. In: Proceedings of the 18th IEEE International Conference on Data Mining. 2018, 197–206
3. Sun F, Liu J, Wu J, Pei C, Lin X, Ou W, Jiang P. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In:

Proceedings of the 28th ACM International Conference on Information and Knowledge Management. 2019, 1441–1450

4. Li W, Wu Y, Liu Y, Pan W, Ming Z. BMLP: behavior-aware MLP for heterogeneous sequential recommendation. Frontiers of Computer Science, 2024, 18(3): 183341
5. Wu S, Tang Y, Zhu Y, Wang L, Xie X, Tan T. Session-based recommendation with graph neural networks. In: Proceedings of the 33rd AAAI Conference on Artificial Intelligence. 2019, 346–353
6. Liang Y, Song Q, Zhao Z, Zhou H, Gong M. BA-GNN: Behavior-aware graph neural network for session-based recommendation. Frontiers of Computer Science, 2023, 17(6): 176613
7. Cheng M, Liu Q, Zhang W, Liu Z, Zhao H, Chen E. A general tail item representation enhancement framework for sequential recommendation. Frontiers of Computer Science, 2024, 18(6): 186333
8. Guo J, Wen L, Zhou Y, Song B, Chi Y, Yu F R. SPACE: Self-supervised dual preference enhancing network for multimodal recommendation. IEEE Transactions on Multimedia, 2024, 26: 8849–8859
9. Tao Z, Liu X, Xia Y, Wang X, Yang L, Huang X, Chua T S. Self-supervised learning for multimedia recommendation. IEEE Transactions on Multimedia, 2023, 25: 5107–5116
10. Zhang X, Xu B, Ren Z, Wang X, Lin H, Ma F. Disentangling id and modality effects for session-based recommendation. In: Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2024, 1883–1892
11. Yan A, He Z, Li J, Zhang T, McAuley J. Personalized showcases: Generating multi-modal explanations for recommendations. In: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2023, 2251–2255
12. Ding Y, Ma Y, Wong W K, Chua T S. Modeling instant user intent and content-level transition for sequential fashion recommendation. IEEE Transactions on Multimedia, 2022, 24: 2687–2700
13. Yang J Q, Dai C, Qu D, Li D, Huang J, Zhan D C, Zeng X, Yang Y. COURIER: contrastive user intention reconstruction for large-scale visual recommendation. Frontiers of Computer Science, 2025, 19(7): 197602
14. Hou Y, Mu S, Zhao W X, Li Y, Ding B, Wen J R. Towards universal sequence representation learning for recommender systems. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 2022, 585–593
15. Hou Y, He Z, McAuley J, Zhao W X. Learning vector-quantized item representation for transferable sequential recommenders. In: Proceedings of the ACM Web Conference 2023. 2023, 1162–1171
16. Wang J, Zeng Z, Wang Y, Wang Y, Lu X, Li T, Yuan J, Zhang R, Zheng H T, Xia S T. MISSRec: Pre-training and transferring multi-modal interest-aware sequence representation for recommendation. In: Proceedings of the 31st ACM International Conference on Multimedia. 2023, 6548–6557
17. Wei W, Huang C, Xia L, Zhang C. Multi-modal self-supervised learning for recommendation. In: Proceedings of the ACM Web Conference 2023. 2023, 790–800
18. Liu Y, Zhang K, Ren X, Huang Y, Jin J, Qin Y, Su R, Xu R, Yu Y, Zhang W. AlignRec: Aligning and training in multimodal recommendations. In: Proceedings of the 33rd ACM International Conference on Information and Knowledge Management. 2024, 1503–1512
19. Liang W, Zhang Y, Kwon Y, Yeung S, Zou J. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. In: Proceedings of the 36th International Conference on Neural Information Processing Systems. 2022, 17612–17625
20. Souza Pereira Moreira d G, Rabhi S, Lee J M, Ak R, Oldridge E. Transformers4Rec: Bridging the gap between nlp and sequential / session-based recommendation. In: Proceedings of the 15th ACM Conference on Recommender Systems. 2021, 143–153
21. Yao Z, Chen X, Wang S, Dai Q, Li Y, Zhu T, Long M. Recommender transformers with behavior pathways. In: Proceedings of the ACM Web Conference 2024. 2024, 3643–3654
22. Zhou K, Yu H, Zhao W X, Wen J R. Filter-enhanced MLP is all you need for sequential recommendation. In: Proceedings of the ACM Web Conference 2022. 2022, 2388–2399
23. Liang J, Zhao X, Li M, Zhang Z, Wang W, Liu H, Liu Z. MMMLP: Multi-modal multilayer perceptron for sequential recommendations. In: Proceedings of the ACM Web Conference 2023. 2023, 1109–1117
24. Liu C, Li X, Cai G, Dong Z, Zhu H, Shang L. Noninvasive self-attention for side information fusion in sequential recommendation. In: Proceedings of the 35th AAAI Conference on Artificial Intelligence. 2021, 4249–4256
25. Zhang X, Xu B, Ma F, Li C, Yang L, Lin H. Beyond co-occurrence: Multi-modal session-based recommendation. IEEE Transactions on Knowledge and Data Engineering, 2024, 36(4): 1450–1462
26. Li J, Zhao T, Li J, Chan J, Faloutsos C, Karypis G, Pantel S M, McAuley J. Coarse-to-fine sparse sequential recommendation. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Devel-

opment in Information Retrieval. 2022, 2082–2086
27. Xie Y, Zhou P, Kim S. Decoupled side information fusion for sequential recommendation. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2022, 1611–1621
28. Lin X, Luo J, Pan J, Pan W, Ming Z, Liu X, Huang S, Jiang J. Multi-sequence attentive user representation learning for side-information integrated sequential recommendation. In: Proceedings of the 17th ACM International Conference on Web Search and Data Mining. 2024, 414–423
29. Chen Y, Liu Z, Li J, McAuley J, Xiong C. Intent contrastive learning for sequential recommendation. In: Proceedings of the ACM Web Conference 2022. 2022, 2172–2182
30. Qiu R, Huang Z, Yin H, Wang Z. Contrastive learning for representation degeneration problem in sequential recommendation. In: Proceedings of the 15th ACM International Conference on Web Search and Data Mining. 2022, 813–823
31. Liu Y, Xia L, Huang C. Selfgnn: Self-supervised graph neural networks for sequential recommendation. In: Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2024, 1609–1618
32. Zhou K, Wang H, Zhao W X, Zhu Y, Wang S, Zhang F, Wang Z, Wen J R. S3-Rec: Self-supervised learning for sequential recommendation with mutual information maximization. In: Proceedings of the 29th ACM International Conference on Information and Knowledge Management. 2020, 1893–1902
33. Yang Y, Huang C, Xia L, Huang C, Luo D, Lin K. Debiased contrastive learning for sequential recommendation. In: Proceedings of the ACM Web Conference 2023. 2023, 1063–1073
34. Wang Y, Liu Y, Wang Q, Wang C, Li C. Poisoning self-supervised learning based sequential recommendations. In: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2023, 300–310
35. Hu H, Guo W, Liu Y, Kan M Y. Adaptive multi-modalities fusion in sequential recommendation systems. In: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. 2023, 843–853
36. Tan J, Xu S, Hua W, Ge Y, Li Z, Zhang Y. IDGenRec: LLM-RecSys alignment with textual id learning. In: Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2024, 355–364
37. Yuan Z, Yuan F, Song Y, Li Y, Fu J, Yang F, Pan Y, Ni Y. Where to go next for recommender systems? id- vs. modality-based recommender models revisited. In: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2023, 2639–2649
38. Li Y, Du H, Ni Y, Zhao P, Guo Q, Yuan F, Zhou X. Multi-modality is all you need for transferable recommender systems. In: Proceedings of the 40th IEEE International Conference on Data Engineering. 2024, 5008–5021
39. Li J, Wang M, Li J, Fu J, Shen X, Shang J, McAuley J. Text is all you need: Learning language representations for sequential recommendation. In: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 2023, 1258–1267
40. Liu Z, Mei S, Xiong C, Li X, Yu S, Liu Z, Gu Y, Yu G. Text matching improves sequential recommendation by reducing popularity biases. In: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. 2023, 1534–1544
41. Qiu Z, Wu X, Gao J, Fan W. U-BERT: Pre-training user representations for improved recommendation. In: Proceedings of the 35th AAAI Conference on Artificial Intelligence. 2021, 3421–3430
42. Huang B, Luo J, Du W, Pan W, Ming Z. Cascaded cross attention for review-based sequential recommendation. In: Proceedings of the 23rd IEEE International Conference on Data Mining. 2023, 170–179
43. Bian S, Pan X, Zhao W X, Wang J, Wang C, Wen J R. Multi-modal mixture of experts represetation learning for sequential recommendation. In: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. 2023, 110–119
44. Pan X, Chen Y, Tian C, Lin Z, Wang J, Hu H, Zhao W X. Multimodal meta-learning for cold-start sequential recommendation. In: Proceedings of the 31st ACM International Conference on Information & Knowledge Management. 2022, 3421–3430
45. Zhao P, Gao X, Xu C, Chen L. M5: Multi-modal multi-interest multi-scenario matching for over-the-top recommendation. In: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 2023, 5650–5659
46. Koukounas A, Mastrapas G, Günther M, Wang B, Martens S, Mohr I, Sturua S, Akram M K, Martínez J F, Ognawala S, Guzman S, Werk M, Wang N, Xiao H. Jina CLIP: Your CLIP model is also your text retriever. arXiv preprint arXiv:2405.20204, 2024
47. Sheng X R, Yang F, Gong L, Wang B, Chan Z, Zhang Y, Cheng Y, Zhu Y N, Ge T, Zhu H, Jiang Y, Xu J, Zheng

B. Enhancing taobao display advertising with multi-modal representations: Challenges, approaches and insights. In: Proceedings of the 33rd ACM International Conference on Information and Knowledge Management. 2024, 4858–4865

48. Lin J, Dai X, Shan R, Chen B, Tang R, Yu Y, Zhang W. Large language models make sample-efficient recommender systems. Frontiers of Computer Science, 2025, 19(4): 194328
49. Yao T, Yi X, Cheng D Z, Yu F, Chen T, Menon A, Hong L, Chi E H, Tjoa S, Kang J J, Ettinger E. Self-supervised learning for large-scale item recommendations. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management. 2021, 4321–4330
50. Radford A, Kim J W, Hallacy C, Ramesh A, Goh G, Agarwal S, Sastry G, Askell A, Mishkin P, Clark J, Krueger G, Sutskever I. Learning transferable visual models from natural language supervision. In: Proceedings of the 38th International Conference on Machine Learning. 2021, 8748–8763
51. Devlin J, Chang M W, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 57th Conference of the North American Chapter of the Association for Computational Linguistics. 2019, 4171–4186
52. Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Lin S, Guo B. Swin Transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the 38th IEEE/CVF International Conference on Computer Vision. 2021, 10012–10022
53. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A N, Kaiser L, Polosukhin I. Attention is all you need. In: Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017, 6000–6010
54. Kumar K, Arici T, Neiman T, Yang J, Sam S, Xu Y, Ferhatosmanoglu H, Tutar I. Unsupervised multi-modal representation learning for high quality retrieval of similar products at e-commerce scale. In: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. 2023, 4667–4673
55. Hu E J, Shen Y, Wallis P, Allen-Zhu Z, Li Y, Wang S, Wang L, Chen W. LoRA: Low-rank adaptation of large language models. In: Proceedings of the 35th International Conference on Learning Representations. 2022
56. Zhang J, Cheng Y, Ni Y, Pan Y, Yuan Z, Fu J, Li Y, Wang J, Yuan F. NineRec: A benchmark dataset suite for evaluating transferable recommendation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2025, 47(7): 5256–5267
57. Krichene W, Rendle S. On sampled metrics for item recommendation. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020, 1748–1757
58. Liu Q, Wu X, Zhao X, Wang Y, Zhang Z, Tian F, Zheng Y. Large language models enhanced sequential recommendation for long-tail user and item. In: Proceedings of the 38th International Conference on Neural Information Processing Systems. 2025, 26701–26727
59. Zhao W X, Mu S, Hou Y, Lin Z, Chen Y, Pan X, Li K, Lu Y, Wang H, Tian C, Min Y, Feng Z, Fan X, Chen X, Wang P, Ji W, Li Y, Wang X, Wen J R. RecBole: Towards a unified, comprehensive and efficient framework for recommendation algorithms. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management. 2021, 4653–4664
60. GLM T. ChatGLM: A family of large language models from glm-130b to glm-4 all tools. arXiv preprint arXiv:2406.12793, 2024
61. Team H. Hunyuan-large: An open-source moe model with 52 billion activated parameters by tencent. arXiv preprint arXiv:2411.02265, 2024
62. Team Q. Qwen technical report. arXiv preprint arXiv:2309.16609, 2023
63. DeepSeek-AI . Deepseek-v3 technical report. arXiv preprint arXiv:2412.19437, 2025
64. Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, Levy O, Lewis M, Zettlemoyer L, Stoyanov V. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692, 2019
65. Ni J, Hernandez Abrego G, Constant N, Ma J, Hall K, Cer D, Yang Y. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. In: Findings of the Association for Computational Linguistics: ACL 2022. 2022, 1864–1874
66. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S, Uszkoreit J, Houlsby N. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint 2010.11929, 2021
67. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition. 2016, 770–778
68. Li Y, Chen T, Zhang P F, Yin H. Lightweight self-attentive sequential recommendation. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management. 2021, 967–977