Multimodal Pre-training Framework for Sequential Recommendation via Contrastive Learning

Lingzi Zhang, Xin Zhou, Zhiqi Shen

***Abstract*—Sequential recommendation systems utilize the sequential interactions of users with items as their main supervision signals in learning users' preferences. However, existing methods usually generate unsatisfactory results due to the sparsity of user behavior data. To address this issue, we propose a novel pre-training framework, named Multimodal Sequence Mixup for Sequential Recommendation (MSM4SR), which leverages both users' sequential behaviors and items' multimodal content (*i.e.*, text and images) for effectively recommendation. Specifically, MSM4SR tokenizes each item image into multiple textual keywords and uses the pre-trained BERT model to obtain initial textual and visual features of items, for eliminating the discrepancy between the text and image modalities. A novel backbone network, *i.e.*, Multimodal Mixup Sequence Encoder (M^2SE), is proposed to bridge the gap between the item multimodal content and the user behavior, using a complementary sequence mixup strategy. In addition, two contrastive learning tasks are developed to assist M^2SE in learning generalized multimodal representations of the user behavior sequence. Extensive experiments on real-world datasets demonstrate that MSM4SR outperforms state-of-the-art recommendation methods. Moreover, we further verify the effectiveness of MSM4SR on other challenging tasks including cold-start and cross-domain recommendation.**

***Index Terms*—Multimodal Recommendation, Sequential Recommendation, Contrastive Learning.**

I. Introduction

SEQUENTIAL recommendation systems are designed to capture the dynamic preferences of users based on their historical behaviors, with the goal of predicting the next item that they will be interested in [1]. The primary supervision signal utilized for learning the parameters of these models is typically derived from the sequential interactions of users with items. However, given the sparsity of user behavior data, sequential recommendation methods that rely solely on such data are susceptible to the problem of data sparsity, resulting in suboptimal performance.

In practice, there exists a significant amount of multimodal information associated with items (*e.g.*, images and text descriptions), which has been employed to alleviate the data sparsity problem in building conventional recommendation systems [2]–[5]. For example, [4], [5] leverage item multimodal content as a regularization factor and integrate it with collaborative filtering frameworks. Recent studies [6]–[8] utilize graph neural networks to uncover the hidden links between different modalities and establish an in-depth understanding of users' preferences. Although these methods have made considerable advancements, they are only applicable to non-sequential recommendation systems.

Motivated by the success of multimodal recommendation methods [4]–[8] and pre-trained models [9], we propose to apply the pre-training and fine-tuning paradigm with contrastive learning to effectively exploit the item multimodal content (*i.e.*, text and images) in order to enhance the sequential recommendation performance. However, there are two major challenges in integrating item multimodal content information with existing sequential recommendation frameworks. *Firstly*, there exists a representation discrepancy across different types of modality content. *Secondly*, there is a domain gap between users' sequential behaviors and items' multimodal content. Intuitively, users' behaviors are not only driven by the contents of items, but also evolve over time.

To this end, we propose a novel pre-training framework, named Multimodal Sequence Mixup for Sequential Recommendation (MSM4SR). To reduce the discrepancy between textual and visual modalities, MSM4SR first tokenizes the images of items into multiple textual keywords using a language-image pre-training model [10]. These keywords are then fed into the pre-trained Sentence-BERT model [11] to obtain items' initial visual embeddings. Analogously, initial textual embeddings of items are derived from their text descriptions using the same Sentence-BERT model. Compared to previous methods [6]–[8], which directly utilize a pre-trained image encoder to extract image features, the use of word tokens generated from images has several advantages. Firstly, it helps to bridge the semantic gap between multimodal features, eliminating the need for additional encoders to map them into a common semantic space. Secondly, it enables the use of existing language models to process multiple images and extract meaningful information while discarding redundant information. Lastly, merchants can use word tokens to describe their products in a more comprehensive manner, allowing them to differentiate themselves from competitors and increase customer engagement.

To address the domain gap between user sequential behaviors and item multimodal content, MSM4SR utilizes a pre-training strategy that optimizes a backbone network, the Multimodal Mixup Sequence Encoder (M^2SE), through the use of contrastive learning losses. M^2SE fuses item multimodal content with user behavior sequences by implementing a complementary sequence mixup strategy to generate mix-modality representations of a user's behavior sequence. Two contrastive learning losses, *i.e.*, 1) modality-specific next item prediction loss, and 2) cross-modality contrastive learning loss, are developed to help M^2SE learn more generalized sequence

Lingzi Zhang, Xin Zhou, and Zhiqi Shen are with the School of Computer Science and Engineering, Nanyang Technological University, Singapore, and also with the Alibaba-NTU Singapore Joint Research Institute (e-mail: lingzi001@e.ntu.edu.sg; xin.zhou@ntu.edu.sg; zqshen@ntu.edu.sg).

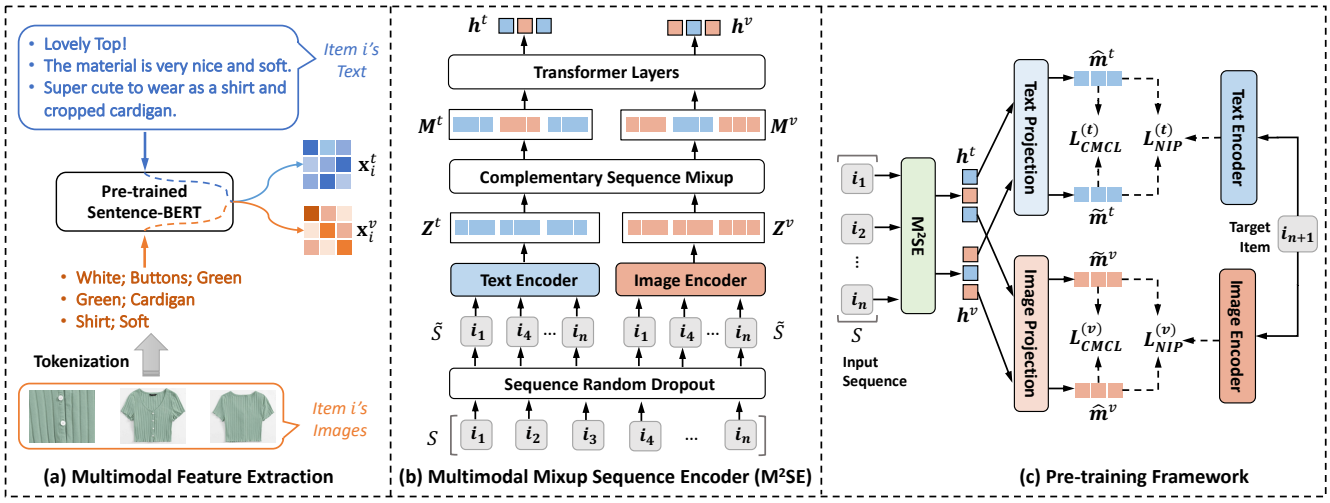


Fig. 1. (a) The multimodal feature extraction module used to obtain initial multimodal features of items. (b) The structure of the proposed multimodal mixup sequence encoder that fuses items' multimodal content with users' behavior sequence. (c) The workflow of the proposed pre-training framework, where $\mathcal{S}$ is the input sequence and i_{n+1} is the target item.

representations across modalities. The modality-specific next item prediction loss captures the correlation between a mix-modality sequence and the next item in each modality space, while the cross-modality contrastive learning loss is employed to calibrate discrepancies in representations from different modality spaces.

We perform extensive experiments on three real-world datasets to evaluate the effectiveness of MSM4SR. Experimental results show that MSM4SR can outperform state-of-the-art approaches for sequential recommendation and multimodal recommendation tasks. Moreover, we explore the promising applications of MSM4SR in other two challenging tasks: cold-start recommendation and cross-domain recommendation. The results suggest that MSM4SR has the potential to benefit the sequential recommendation task in both cold-start and cross-domain settings.

II. Related Work

A. Sequential Recommendation

Earlier works [12] on sequential recommendation adopt Markov Chain (MC) to capture the transitions over user-item interaction sequences. However, these methods are not well-suited to handle complex sequence patterns. More recently, deep learning techniques such as Convolutional Neural Networks (CNNs) [13], [14], Recurrent Neural Networks (RNNs) [15], [16], and Graph Neural Networks (GNNs) [17], [18] are applied to model users' sequential behaviors. Moreover, transformer architecture-based methods [19]–[22] have shown strong performance in capturing long-range dependencies in a sequence.

To improve the performance of sequential recommendation, some recent studies integrate various auxiliary information into the sequential recommendation framework. For example, in FDSA [2], different item features are first aggregated using a vanilla attention layer, followed by a feature-based self-attention block to learn how features transit among items in a sequence. The S^3-Rec model [21] adopts a pre-training strategy to predict the correlation between an item and its attributes. Moreover, in DIF-SR [23], the auxiliary information is moved from the input to the attention layer. The attention calculation of auxiliary information and item representation is decoupled to improve the modeling capability of item representations. Despite the success of these sequential recommendation methods, they ignore the multimodal information of items.

B. Multimodal Recommendation

Multimodal recommendation methods exploit the multimodal content of items to improve recommendation performance. Previous works [4], [5], [24] incorporate the visual features of item images into the matrix factorization based recommendation framework. In other works [3], [25], [26], attention networks are used to combine multimodal features to enhance the representation learning of users and items. For example, UVCAN [25] learns multi-modal information of users and items using stacked attention networks. Moreover, several recent studies [6]–[8], [27], [28] leverage GNNs to exploit the item multimodal information. For example, LATTICE [8] embeds a modality-aware graph structure learning layer, which identifies item-item graph structures using multimodal features. DualGNN [27] explicitly models the user's attention over different modalities and inductively learns her multi-modal preference. In MVGAE [28], a multi-modal variational graph auto-encoder is proposed to fuse modality-specific node embeddings according to the product-of-experts principle.

III. Methodology

Let $\mathcal{I}$ denote the set of items, and $\mathcal{S} = \{i_1, i_2, \cdots, i_n\}$ denote a user behavior sequence, where n items are sorted in a chronological order based on the interaction timestamp. In this work, we consider the text and image content of items to build the model. For each item i, it is associated

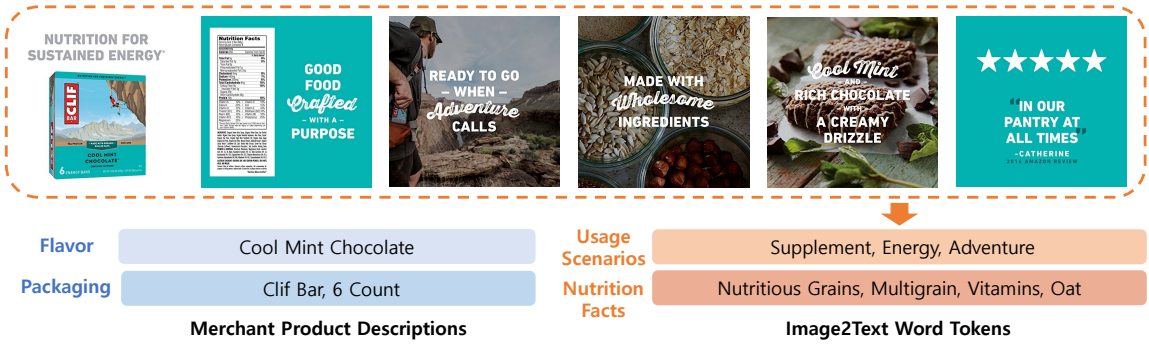


Fig. 2. An example of converting images of an item into text tokens. Images are retrieved from Amazon Pantry dataset. Text tokens are generated using CLIP [10]. Best viewed in color.

with a chunk of text descriptions that is split into sentences as $\mathcal{T}_i = \{t_1^i, t_2^i, \cdots, t_{|\mathcal{T}_i|}^i\}$, and a set of images $\mathcal{V}_i = \{v_1^i, v_2^i, \cdots, v_{|\mathcal{V}_i|}^i\}$, where $|\mathcal{T}_i|$ and $|\mathcal{V}_i|$ denote the number of sentences and images, respectively. Next, we introduce the main components of MSM4SR, including multimodal feature extraction, backbone network M^{2}SE, pre-training objectives, and fine-tuning objectives.

A. Multimodal Feature Extraction

Figure 1(a) shows the workflow using pre-trained models to obtain the initial text and image features of items.

1) Text Feature Extraction: For each sentence in $\mathcal{T}_i$, we feed it into the pre-trained Sentence-BERT [11] to obtain its latent representation. The initial text feature $\mathbf{x}_i^t$ of item i is obtained by stacking representations of all the sentences in $\mathcal{T}_i$ as follows,

$$\mathbf{x}_i^t = stack\left[\text{BERT}(t_1^i), \text{BERT}(t_2^i), \cdots, \text{BERT}(t_{|\mathcal{T}_i|}^i)\right], \quad (1)$$

where $\mathbf{x}_i^t \in \mathbb{R}^{|\mathcal{T}_i| \times d}$, $stack[,]$ denotes stacking multiple vectors into a matrix, and d is the embedding dimension.

2) Image Feature Extraction: Inspired by [29], we use a pre-trained language-image model, *i.e.*, CLIP [10], to describe each image by text tokens. This process can help eliminate the gap between text and image modality representations. To capture the key visual information of an image, N most relevant text tokens are retained based on their similarities to the image. Then, we obtain the initial feature $\mathbf{v}_\ell^i$ for an item image $v_\ell^i \in \mathcal{V}_i$, by concatenating these text tokens as a sentence and feeding it into the same pre-trained Sentence-BERT model. The initial image feature $\mathbf{x}_i^v$ of item i can be obtained by stacking the features of all images in $\mathcal{V}_i$ as follows,

$$\begin{aligned} f(w) &= \text{sim}(\text{CLIP}(v_\ell^i), \text{CLIP}(w)) \quad \forall w \in \mathcal{D}, \\ \mathbf{v}_\ell^i &= \text{BERT}(concat(\text{TopN}(\{f(w_1), \cdots, f(w_{|\mathcal{D}|})\}, N))), \\ \mathbf{x}_i^v &= stack[\mathbf{v}_1^i, \mathbf{v}_2^i, \cdots, \mathbf{v}_{|\mathcal{V}_i|}^i], \end{aligned} \quad (2)$$

where $\mathbf{x}_i^v \in \mathbb{R}^{|\mathcal{V}_i| \times d}$, w is a text token in the word dictionary $\mathcal{D}$, and $|\mathcal{D}|$ denotes the size of the dictionary. $\text{sim}(\cdot)$ is to compute the cosine similarity between the embedding of an image v_ℓ^i and a word w obtained by the CLIP model. $\text{TopN}(\cdot)$ function selects N words that have the highest similarities with the image. $concat(\cdot)$ is the operation of concatenating N words into one sentence. Note that $\mathbf{x}_i^t$ and $\mathbf{x}_i^v$ are derived during data pre-processing stage.

In order to verify text tokens retrieved from images using the pre-trained language-image model, we select one representative item from the Amazon Pantry dataset for analysis. The details of the item are depicted in Figure 2. The item is an energy bar that aims to provide athletes with carbohydrates and protein. While the seller of the item provides a brief summary, it is not comprehensive enough to fully convey the item to customers. By generating text tokens from images, key information such as nutritional facts (*e.g.*,, grains, vitamins, oats) and usage modes (*e.g.*,, adventure, supplement, energy) are obtained. Compared to the image feature extracted from pre-trained image encoders (*e.g.*, ResNet [6], [8]), word tokens can discard irrelevant information to achieve better recommendation performance. Additionally, word tokens converted from images can provide merchants with a new perspective to enhance item descriptions and market items more effectively.

B. Multimodal Mixup Sequence Encoder

Next, we propose a backbone network, *i.e.*, M^{2}SE, to encode user sequences with multimodal features extracted from items. The structure of M^{2}SE is shown in Figure 1(b). Observe that M^{2}SE includes four main components: sequence random dropout, text and image encoders, complementary sequence mixup, and transformer layers. Next, we introduce the details of each component.

1) Sequence Random Dropout: For a user behavior sequence $\mathcal{S}$, M^{2}SE randomly drops a portion of items from $\mathcal{S}$ with a drop ratio ρ, to help the model achieve better generalization performance. The obtained sub-sequence after the random dropout operation is denoted by $\widetilde{\mathcal{S}}$.

2) Text and Image Encoders: These two encoders are used to adapt the initial modality features of items obtained from the pre-trained language model to learn users' sequential behaviors. Both encoders share the same structure, including an attention layer and a Mixture-of-Expert (MoE) architecture [30].

In the text encoder, each item $i \in \widetilde{\mathcal{S}}$ is represented by its initial textual feature $\mathbf{x}_i^t$. The attention layer is composed of two

linear transformations to fuse i's sentence-level embeddings as follows,

$$\alpha^t = \text{softmax}\big((\mathbf{x}_i^t\mathbf{W}_1^t + \mathbf{b}_1^t)\mathbf{W}_2^t + b_2^t\big),$$
$$\mathbf{e}_i^t = \sum_{j=1}^{|\mathcal{T}_i|} \alpha_j^t \mathbf{x}_i^t[j,:], \tag{3}$$

where $\mathbf{W}_1^t \in \mathbb{R}^{d\times d_a}$, $\mathbf{W}_2^t \in \mathbb{R}^{d_a}$, $\mathbf{b}_1^t \in \mathbb{R}^{d_a}$, and $b_2^t \in \mathbb{R}$ are learnable parameters. d_a is the attention dimension size. α_j^t is the j-th element of α^t, and $\mathbf{x}_i^t[j,:]$ denotes the j-th row of feature matrix $\mathbf{x}_i^t$. Then, MoE is used to increase the model's capacity for adapting the fused modality representation $\mathbf{e}_i^t$. Each expert in MoE consists of a linear transformation, followed with a dropout layer and a normalization layer. Let $E_k(\mathbf{e}_i^t) \in \mathbb{R}^{d_0}$ denote the output of the k-th expert network, and $\mathbf{g}^t \in \mathbb{R}^O$ is the output of the gating network as follows,

$$E_k(\mathbf{e}_i^t) = \text{LayerNorm}(\text{Dropout}(\mathbf{e}_i^t\mathbf{W}_k^t)),$$
$$\mathbf{g}^t = \text{softmax}(\mathbf{e}_i^t\mathbf{W}_3^t), \tag{4}$$

where $\mathbf{W}_3^t \in \mathbb{R}^{d\times O}$ and $\mathbf{W}_k^t \in \mathbb{R}^{d\times d_0}$ are learnable parameters, O is the number of experts, and d_0 is the dimension of the hidden embedding. Then, the output of MoE for item i is formulated as follows,

$$\mathbf{z}_i^t = \sum_{k=1}^{O} g_k^t E_k(\mathbf{e}_i^t), \tag{5}$$

where $\mathbf{z}_i^t \in \mathbb{R}^{d_0}$, and g_k^t is the weight derived from k-th gating router. Here, we omit bias terms in the equation for simplicity. The outputs of MoE network for all items in $\widetilde{\mathcal{S}}$ are stacked to form the output of the text encoder, which is denoted by $\mathbf{Z}^t = stack[\mathbf{z}_1^t, \mathbf{z}_2^t, \cdots, \mathbf{z}_{|\widetilde{\mathcal{S}}|}^t]$.

Similarly, in the image encoder, each item in $\widetilde{\mathcal{S}}$ is represented by its image feature $\mathbf{x}_i^v$. The output of the image encoder is denoted by $\mathbf{Z}^v = stack[\mathbf{z}_1^v, \mathbf{z}_2^v, \cdots, \mathbf{z}_{|\widetilde{\mathcal{S}}|}^v]$, where $\mathbf{z}_i^v$ is the output of the MoE network for the i-th item in $\widetilde{\mathcal{S}}$.

3) Complementary Sequence Mixup: To alleviate the representation discrepancy between two different modality sequences, we propose a complementary sequence mixup method, which mixes up text representations and image representations in a complementary manner. Specifically, we define a mixup ratio p between 0 to 0.5, which is randomly generated during model training. For each item in $\widetilde{\mathcal{S}}$, we swap its embedding in $\mathbf{Z}^t$ and $\mathbf{Z}^v$ with probability p and generate two mix-modality sequence embeddings $\mathbf{M}^t$ and $\mathbf{M}^v$. The definition of $p \leq 0.5$ ensures the generated mix-modality sequence embedding dominates with information from the same modality. In this case, $\mathbf{M}^t$ and $\mathbf{M}^v$ complement each other in terms of the modality choice for each item in the sequence.

4) Transformer Layers: The Transformer structure [31] is used to further encode $\mathbf{M}^t$ and $\mathbf{M}^v$. We first add positional encodings to $\mathbf{M}^t$ and $\mathbf{M}^v$, and then feed the summed embeddings into L Transformer layers. Note that each Transformer layer consists of a multi-head self-attention sub-layer and a point-wise feed-forward network. Let $\mathbf{H}_L^t$ and $\mathbf{H}_L^v$ denote the output of the L-th Transformer layer based on $\mathbf{M}^t$ and $\mathbf{M}^v$, respectively. Following [21], we use last rows in $\mathbf{H}_L^t$ and $\mathbf{H}_L^v$ as two mix-modality representations of the input sequence, which are denoted by $\mathbf{h}^t$ and $\mathbf{h}^v$.

C. Pre-training Objectives

To pre-train the backbone model, we propose two optimization objectives, *i.e.*, modality-specific next item prediction, and cross modality contrastive learning, based on mix-modality sequence representations. These objectives enable the backbone model to better capture the connections between representations across different modalities. The workflow in the pre-training phase is shown in Figure 1(c).

Let $\mathcal{B} = \{(\mathcal{S}_j, i_j)\}_{j=1}^{|\mathcal{B}|}$ denote a batch of pre-training data, where $\mathcal{S}_j$ denotes a user's behavior sequence and i_j is her next interaction item after $\mathcal{S}_j$. With M^{2}SE, we can obtain two mix-modality sequence representations $\mathbf{h}_j^t$ and $\mathbf{h}_j^v$ for $\mathcal{S}_j$. As $\mathbf{h}_j^t$ and $\mathbf{h}_j^v$ are obtained by mixing up modalities, we first use two linear transformations to map them into the text feature space and image feature space respectively,

$$\widehat{\mathbf{m}}_j^t = \mathbf{h}_j^t\mathbf{W}_t + \mathbf{b}_t, \qquad \widetilde{\mathbf{m}}_j^t = \mathbf{h}_j^v\mathbf{W}_t + \mathbf{b}_t,$$
$$\widehat{\mathbf{m}}_j^v = \mathbf{h}_j^v\mathbf{W}_v + \mathbf{b}_v, \qquad \widetilde{\mathbf{m}}_j^v = \mathbf{h}_j^t\mathbf{W}_v + \mathbf{b}_v, \tag{6}$$

where $\mathbf{W}_t, \mathbf{W}_v \in \mathbb{R}^{d_0\times d_0}$ and $\mathbf{b}_t, \mathbf{b}_v \in \mathbb{R}^{d_0}$ are learnable parameters. Motivated by the success of contrastive learning in model pre-training, we define the pre-training objective functions in a contrastive manner.

1) Modality-specific Next Item Prediction: Modality-specific Next Item Prediction (NIP) aims to predict the next item based on the mix-modality sequence representations. For each $(\mathcal{S}_j, i_j)$ pair, $\mathcal{S}_j$ is the input sequence and i_j is the target item. Thus, in the text feature space, we pair $\widehat{\mathbf{m}}_j^t$ and $\widetilde{\mathbf{m}}_j^t$ with the i_j's text embedding $\mathbf{z}_j^t$ obtained by the text encoder as a positive sample, and pair $\widehat{\mathbf{m}}_j^t$ and $\widetilde{\mathbf{m}}_j^t$ with the text embeddings of other items $\{i_{j'}|j' \neq j, 1 \leq j' \leq |\mathcal{B}|\}$ from $\mathcal{B}$ as negative samples. The next item prediction loss defined in the text feature space is as follows,

$$\mathcal{L}_{\text{NIP}}^{(t)} = -\sum_{j=1}^{|\mathcal{B}|} \log \frac{f(\widehat{\mathbf{m}}_j^t, \mathbf{z}_j^t) + f(\widetilde{\mathbf{m}}_j^t, \mathbf{z}_j^t)}{\sum_{j'=1}^{|\mathcal{B}|}[f(\widehat{\mathbf{m}}_j^t, \mathbf{z}_{j'}^t) + f(\widetilde{\mathbf{m}}_j^t, \mathbf{z}_{j'}^t)]}, \tag{7}$$

where $f(\mathbf{s}, \mathbf{z}) = \exp(\text{sim}(\mathbf{s}, \mathbf{z})/\tau)$, and τ is a temperature hyper-parameter. Similarly, we can define the next item prediction loss $\mathcal{L}_{\text{NIP}}^{(v)}$ in the image feature space as follows,

$$\mathcal{L}_{\text{NIP}}^{(v)} = -\sum_{j=1}^{|\mathcal{B}|} \log \frac{f(\widehat{\mathbf{m}}_j^v, \mathbf{z}_j^v) + f(\widetilde{\mathbf{m}}_j^v, \mathbf{z}_j^v)}{\sum_{j'=1}^{|\mathcal{B}|}[f(\widehat{\mathbf{m}}_j^v, \mathbf{z}_{j'}^v) + f(\widetilde{\mathbf{m}}_j^v, \mathbf{z}_{j'}^v)]}. \tag{8}$$

2) Cross Modality Contrastive Learning: To capture the semantic relationship between different modalities, we develop a Cross Modality Contrastive Loss (CMCL). Specifically, the complementary mix-modality sequence representations mapped to the same feature space, *e.g.*, ($\widehat{\mathbf{m}}_j^t$, $\widetilde{\mathbf{m}}_j^t$) and ($\widehat{\mathbf{m}}_j^v$, $\widetilde{\mathbf{m}}_j^v$), are paired as positive samples, while randomly-selected samples in the training batch are paired as negative samples.

Following [32], CMCL for the text space is defined in a symmetric contrastive way as follows,

$$\ell(\widehat{\mathbf{m}}_j^t, \widetilde{\mathbf{m}}_j^t) = \log \frac{f(\widehat{\mathbf{m}}_j^t, \widetilde{\mathbf{m}}_j^t)}{\sum_{j'=1}^{|\mathcal{B}|} f(\widehat{\mathbf{m}}_j^t, \widetilde{\mathbf{m}}_{j'}^t) + \sum_{j' \neq j}^{|\mathcal{B}|} f(\widehat{\mathbf{m}}_j^t, \widehat{\mathbf{m}}_{j'}^t)},$$

$$\mathcal{L}_{\text{CMCL}}^{(t)} = -\frac{1}{2} \sum_{j=1}^{|\mathcal{B}|} (\ell(\widehat{\mathbf{m}}_j^t, \widetilde{\mathbf{m}}_j^t) + \ell(\widetilde{\mathbf{m}}_j^t, \widehat{\mathbf{m}}_j^t)). \tag{9}$$

Similarly, CMCL in the image space is defined as follows,

$$\mathcal{L}_{\text{CMCL}}^{(v)} = -\frac{1}{2} \sum_{j=1}^{\mathcal{B}} (\ell(\widehat{\mathbf{m}}_j^v, \widetilde{\mathbf{m}}_j^v) + \ell(\widetilde{\mathbf{m}}_j^v, \widehat{\mathbf{m}}_j^v)). \tag{10}$$

The overall loss function for model pre-training is formulated as follows,

$$\mathcal{L}_{\text{pre-train}} = \mathcal{L}_{\text{NIP}}^{(t)} + \mathcal{L}_{\text{NIP}}^{(v)} + \lambda(\mathcal{L}_{\text{CMCL}}^{(t)} + \mathcal{L}_{\text{CMCL}}^{(v)}), \tag{11}$$

where λ is a hyper-parameter used to balance these two groups of losses.

D. Fine-tuning for Sequential Recommendation

Let $\widetilde{\mathcal{B}}$ denote a batch of fine-tuning data. For each $(\mathcal{S}, i) \in \widetilde{\mathcal{B}}$, i is the user's next interaction item after her interaction sequence $\mathcal{S}$. From the classification perspective, i can be treated as the target label of the input sequence $\mathcal{S}$. Following [19], [21], we treat sequential recommendation as a supervised classification problem and use cross-entropy loss for model fine-tuning.

In the fine-tuning stage, we disable the sequence random dropout and complementary sequence mixup operations in the pre-trained M^{2}SE network, by setting the dropout ratio ρ and sequence mix up ratio p to 0. Moreover, we also incorporate the ID embeddings $\mathbf{E}_{\mathcal{S}}$ of items in the sequence $\mathcal{S}$ with its text and image representations $\mathbf{M}^t$ and $\mathbf{M}^v$ using element-wise summation, and then feed the summed embeddings into the Transformer layers to obtain the sequence embeddings $\mathbf{h}^t$ and $\mathbf{h}^v$. Then, for the sequence $\mathcal{S}$, the predicted probability distribution $\widehat{\mathbf{y}}^{(\mathcal{S})}$ of its potential target labels (*i.e.*, items) is defined as follows,

$$\widehat{\mathbf{y}}^{(\mathcal{S})} = \text{softmax}\big(\mathbf{h}^t(\mathbf{F}^t + \mathbf{E})^\top + \mathbf{h}^v(\mathbf{F}^v + \mathbf{E})^\top\big), \tag{12}$$

where $\widehat{\mathbf{y}}^{(\mathcal{S})} \in \mathbb{R}^{|\mathcal{I}|}$, $\mathbf{E} \in \mathbb{R}^{|\mathcal{I}| \times d_0}$ denotes the ID embedding matrix of all items. $\mathbf{F}^t, \mathbf{F}^v \in \mathbb{R}^{|\mathcal{I}| \times d_0}$ denote the text and image modality embedding matrices of all items, which are obtained by the text encoder and image encoder in M^{2}SE. The loss function for model fine-tuning is defined as follows,

$$\mathcal{L}_{\text{finetune}} = -\sum_{(\mathcal{S}, i) \in \widetilde{\mathcal{B}}} \log\big(\widehat{\mathbf{y}}^{(\mathcal{S})}(i)\big), \tag{13}$$

where $\widehat{\mathbf{y}}^{(\mathcal{S})}(i)$ denotes the predicted probability for the ground-truth label i of the sequence $\mathcal{S}$. By minimizing Eq. (13), we fine-tune the parameters of the pre-trained M^{2}SE network, as well as the ID embedding of items.

Alternatively, MSM4SR can be applied when only one modality is available. A comparison between unimodal and multimodal settings is discussed in the Section IV-D.

TABLE I
STATISTICS OF THE EXPERIMENTAL DATASETS. "AVG. N" DENOTES THE AVERAGE LENGTH OF INTERACTION SEQUENCES.

Datasets	#Users	#Items	#Inter.	Avg. n
Pantry	13,614	7,670	131,311	9.65
Arts	32,216	52,557	264,465	8.21
Office	68,224	59,705	527,209	7.73

IV. EXPERIMENTS

A. Experimental Settings

1) Datasets: The experiments are conducted on the Amazon review dataset [33], which provides the multimodal information of items. We use three "5-core" subsets for experimental evaluation, *i.e.*, *Pantry*, *Arts*, and *Office*. Following [18], [21], we convert each rating into an implicit feedback record. On each dataset, we group interactions by users and construct the interaction sequence for each user by sorting her interactions in a chronological order. The statistics of the pre-processed experimental datasets are summarized in Table I.

2) Evaluation Settings: Following [20], [21], we apply the *leave-one-out* strategy to evaluate the performance of recommendation models in both pre-training and fine-tuning stages. Specifically, for each user, the last item of her interaction sequence is used for testing, the second last item is used for validation, and the remaining items are used for model training. The performance of a recommendation model is evaluated by two widely used metrics, *i.e.*, Recall@K and Normalized Discounted Cumulative Gain@K (respectively denoted by R@K and N@K). K is empirically set to 5, 10, and 20. All evaluation metrics are computed on the whole candidate item set without negative sampling.

3) Baseline Methods: We compare the proposed model with three groups of baseline methods:

- *General Recommendation Model:* **LightGCN** [34] is one of the most representative GNN-based recommendation methods. It simplifies the design of GCNs for collaborative filtering by discarding feature transformation and non-linear activation.
- *Multimodal Recommendation Models:* 1) **GRCN** [7] is a graph-based multimodal recommendation model that refines the user-item interaction graph by identifying false-positive feedback and pruning noisy edges; 2) **DualGNN** [27] incorporates a multimodal representation learning module to explicitly model the user's attentions over different modalities.
- *Sequential Recommendation Models:* 1) **MV-RNN** [35] is a multimodal sequential recommendation model that combines multimodal features at its input and applies a recurrent structure to dynamically capture users' interests; 2) **SASRec** [19] is a directional self-attention method for next item prediction; 3) **S^3-Rec** [21] devises four self-supervised learning objectives for sequential recommendation based on the mutual information maximization principle; 4) **SINE** [36] introduces a sparse-interest module that adaptively infers a sparse set of concepts and outputs multiple embeddings for each user; 5) **DIF-**

TABLE II
THE OVERALL PERFORMANCE ACHIEVED BY DIFFERENT METHODS. THE BEST RESULTS ARE IN **BOLDFACE**, AND THE SECOND BEST RESULTS ARE UNDERLINED. %IMP INDICATES THE RELATIVE IMPROVEMENT PERCENTAGE OF MSM4SR OVER THE BEST BASELINE METHOD.

Dataset	Metric	LightGCN	GRCN	DualGNN	MV-RNN	SASRec	S^3-Rec	SINE	DIF-SR	SASRecM	MSM4SR	%Imp
Pantry	R@5	0.0270	0.0365	0.0321	0.0157	0.0277	0.0315	0.0297	0.0300	0.0369	**0.0405**	9.76%
	R@10	0.0460	0.0552	0.0485	0.0276	0.0457	0.0535	0.0534	0.0473	0.0600	**0.0673**	12.17%
	R@20	0.0774	0.0856	0.0739	0.0467	0.0722	0.0845	0.0873	0.0736	0.0934	**0.1040**	11.35%
	N@5	0.0176	0.0229	0.0202	0.0101	0.0147	0.0187	0.0167	0.0163	0.0223	**0.0235**	2.62%
	N@10	0.0236	0.0289	0.0254	0.0134	0.0204	0.0257	0.0243	0.0219	0.0298	**0.0321**	7.72%
	N@20	0.0315	0.0366	0.0318	0.0184	0.0271	0.0335	0.0329	0.0284	0.0382	**0.0414**	8.38%
Arts	R@5	0.0543	0.0546	0.0596	0.0299	0.0704	0.0715	0.0667	0.0712	0.0816	**0.0854**	4.66%
	R@10	0.0726	0.0741	0.0788	0.0446	0.0910	0.0961	0.0935	0.0899	0.1099	**0.1184**	7.73%
	R@20	0.0967	0.0999	0.1033	0.0661	0.1125	0.1250	0.1237	0.1126	0.1430	**0.1570**	9.79%
	N@5	0.0381	0.0386	0.0433	0.0184	0.0442	0.0467	0.0404	0.0449	0.0525	**0.0531**	1.14%
	N@10	0.0440	0.0448	0.0495	0.0232	0.0509	0.0546	0.0491	0.0510	0.0616	**0.0637**	3.41%
	N@20	0.0501	0.0513	0.0557	0.0283	0.0563	0.0619	0.0567	0.0567	0.0699	**0.0735**	5.15%
Office	R@5	0.0325	0.0556	0.0518	0.0259	0.0841	0.0823	0.0837	0.0857	0.0850	**0.0968**	12.95%
	R@10	0.0518	0.0714	0.0661	0.0416	0.1025	0.1027	0.1059	0.1039	0.1060	**0.1206**	13.77%
	R@20	0.0752	0.0911	0.0843	0.0641	0.1222	0.1254	0.1305	0.1241	0.1316	**0.1480**	12.46%
	N@5	0.0219	0.0408	0.0385	0.0159	0.0558	0.0575	0.0546	0.0561	0.0584	**0.0721**	23.46%
	N@10	0.0281	0.0460	0.0431	0.0210	0.0617	0.0641	0.0618	0.0620	0.0652	**0.0797**	22.23%
	N@20	0.0339	0.0509	0.0477	0.0266	0.0667	0.0698	0.0680	0.0671	0.0716	**0.0866**	20.95%

SR [23] moves various attribute information from the input to the attention layer, and decouples attribute information and item representation from the calculation of attention; 6) **SASRecM** integrates multimodal information of items with the SASRec architecture. Specifically, the model encodes modality features using the same text/image encoder as MSM4SR, sums them with item ID embeddings, and feeds the output into SASRec to generate recommendations.

4) Implementation Details: The proposed method is implemented by Pytorch [37] and an open-source recommendation framework RecBole [38]. The Adam optimizer [39] is used to learn model parameters. Following [21], we set the maximum sequence length to 50. Our training phase consists of two stages: pre-training and fine-tuning. The learned parameters from the pre-training stage are used to initialize the M^2SE network in the fine-tuning stage. Both the pre-training and fine-tuning are performed on the same dataset to obtain the final recommendation results.

In the multimodal feature extraction stage of MSM4SR, the pre-trained Sentence-BERT model maps every sentence of text descriptions or a group of word tokens extracted from an image into a 768-dimensional dense vector, *i.e.*, $d = 768$. For each item, we consider up to 10 sentences and 10 images.

For pre-training MSM4SR, we set the learning rate to 0.001, batch size to 1024, and the number of experts O to 8, on all datasets. In addition, we set ρ, τ, and λ to 0.2, 0.07, and 0.01, respectively. The attention dimension d_a and embedding dimension d_0 are fixed to 64. The proposed model is pre-trained for 300 epochs.

For fine-tuning MSM4SR and training baseline methods, we apply grid-search to identify the best hyper-parameter settings based on the validation data for each method. The search space is as follows: learning rate in $\{0.0001, 0.0005, 0.001\}$, batch size in $\{256, 512, 1024\}$, and weight decay in $\{0.0001, 0.0005, 0.001\}$. For fair comparison, the hyper-parameters of Transformer layers are kept identical for MSM4SR and transformer-based baselines (*i.e.*, SASRec, S^3-Rec, and DIF-SR). Specifically, the number of attention heads and number of self-attention blocks are set to 2. The remaining hyper-parameters for baseline methods follow the original papers. Additionally, we adopt an early stopping strategy, *i.e.*, we apply a premature stopping if R@20 on the validation data does not increase for 10 epochs.

B. Performance Comparison

We summarize the overall performance comparison results in Table II, from which we have the following observations. *Firstly*, the multimodal recommendation methods (*i.e.*, GRCN and DualGNN) consistently outperform the general recommendation model (*i.e.*, LightGCN). This suggests that leveraging the multimodal information of items can effectively enhance the recommendation performance. *Secondly*, sequential recommendation models generally perform better than non-sequential recommendation models, by capturing users' sequential behavior patterns. However, an exception to this is observed on the Pantry dataset, where the non-sequential model GRCN, which utilizes multimodal data, achieves better performance than sequential baseline methods. One potential explanation is that the multimodal features of items are more informative for this dataset, compared with other two datasets. Meanwhile, S^3-Rec usually achieves better performance than other sequential recommendation baselines, highlighting the effectiveness of using self-supervised signals and side information (*i.e.*, item attributes) for pre-training sequential recommendation models. *Thirdly*, MSM4SR and SASRecM both outperform baseline methods on most evaluation metrics by leveraging multimodal data and simultaneously capturing sequential behavior patterns of users to improve recommendation accuracy. *Lastly*, the proposed MSM4SR model consistently outperforms all baseline methods by a significant margin. This is attributed to the utilization of two pre-training objectives to

TABLE III
THE RECOMMENDATION PERFORMANCE OF MSM4SR AND ITS VARIANTS ON PANTRY AND OFFICE DATASETS.

Methods	Pantry						Office					
	R@5	R@10	R@20	N@5	N@10	N@20	R@5	R@10	R@20	N@5	N@10	N@20
MSM4SR	**0.0405**	**0.0673**	**0.1040**	**0.0235**	**0.0321**	**0.0414**	**0.0968**	**0.1206**	**0.1480**	**0.0721**	**0.0797**	**0.0866**
$\text{MSM4SR}_{\text{ResNet}}$	0.0387	0.0647	0.1007	0.0234	0.0317	0.0408	0.0924	0.1159	0.1435	0.0673	0.0749	0.0818
$\text{MSM4SR}_{\text{w/o NIP}}$	0.0292	0.0501	0.0816	0.0179	0.0245	0.0324	0.0863	0.1062	0.1302	0.0573	0.0637	0.0697
$\text{MSM4SR}_{\text{w/o CMCL}}$	0.0390	0.0662	0.1030	0.0223	0.0310	0.0403	0.0875	0.1095	0.1349	0.0578	0.0649	0.0713
$\text{MSM4SR}_{\text{w/o Mixup}}$	0.0380	0.0649	0.1014	0.0220	0.0307	0.0399	0.0957	0.1192	0.1480	0.0692	0.0768	0.0841
$\text{MSM4SR}_{\text{w/o Pre-train}}$	0.0359	0.0595	0.0920	0.0203	0.0286	0.0369	0.0883	0.1094	0.1335	0.0597	0.0665	0.0726
$\text{MSM4SR}_{\text{E2E}}$	0.0355	0.0605	0.0937	0.0209	0.0290	0.0373	0.0821	0.1013	0.1243	0.0527	0.0589	0.0647

capture the correlation of multimodal data with user behaviors, thereby improving the generalization capabilities of sequential recommendation models and resulting in the best overall performance.

C. Ablation Study

To study the contribution of each component of MSM4SR to sequential recommendation, we consider the following variants of MSM4SR for evaluation: 1) $\textbf{MSM4SR}_{\text{ResNet}}$: we use ResNet to extract features of item images, instead of converting an item image into keywords; 2) $\textbf{MSM4SR}_{\text{w/o NIP}}$: we remove the modality-wise next item prediction losses in the pre-training stage; 3) $\textbf{MSM4SR}_{\text{w/o CMCL}}$: we remove the cross-modality contrastive losses in the pre-training stage; 4) $\textbf{MSM4SR}_{\text{w/o Mixup}}$: we remove the complementary sequence mixup module in M^2SE; 5) $\textbf{MSM4SR}_{\text{w/o Pre-train}}$: we remove the pre-training tasks and train the proposed model from scratch based on the multimodal fine-tuning setting. 6) $\textbf{MSM4SR}_{\text{E2E}}$: we optimize the proposed model in an end-to-end manner by summing up the pre-training loss $\mathcal{L}_{\text{pre-train}}$ and the finetuning loss $\mathcal{L}_{\text{finetune}}$.

Table III presents the performance of MSM4SR and its variants on Pantry and Office datasets. It shows that each proposed component of MSM4SR consistently improves recommendation performance. Moreover, the modality-wise next item prediction losses are particularly important for pre-training MSM4SR for sequential recommendation, as removing them leads to a significant decline in performance. This is likely due to the fact that the model optimizes next item prediction as its main objective during both pre-training and fine-tuning stages. Furthermore, cross modality contrastive losses are more effective in improving recommendation performance on Office dataset compared to the Pantry dataset, indicating that items in the Office dataset provide more training signals to align various modalities. Additionally, we note that the recommendation performance decreases significantly when pre-training tasks are removed, which further validates the effectiveness of applying pre-training for multimodal sequential recommendation. Lastly, if the model is trained end-to-end by combining the $\mathcal{L}_{\text{pre-train}}$ and $\mathcal{L}_{\text{finetune}}$, the performance becomes worse as pre-train losses aim to learn the interactions across different modalities, while finetuning losses emphasize recommendation tasks using cross entropy losses. If they are optimized together, the model is unable to converge to the optimal solution for the recommendation task.

TABLE IV
THE RECOMMENDATION PERFORMANCE ACHIEVED BY THE BEST BASELINE METHOD AND MSM4SR UNDER UNIMODAL AND MULTIMODAL-BASED SETTINGS ON EACH DATASET.

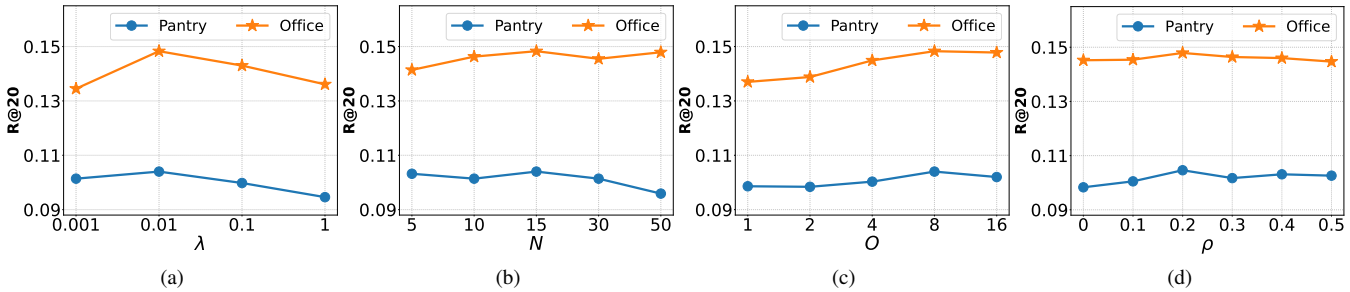
Dataset	Model	R@10	R@20	N@10	N@20
Pantry	GRCN	0.0552	0.0856	0.0289	0.0366
	MSM4SR-V	0.0596	0.0928	0.0290	0.0373
	MSM4SR-T	0.0649	0.1001	0.0318	0.0407
	MSM4SR	**0.0673**	**0.1040**	**0.0321**	**0.0414**
Arts	S^3-Rec	0.0961	0.1250	0.0546	0.0619
	MSM4SR-V	0.1018	0.1345	0.0581	0.0663
	MSM4SR-T	0.1144	0.1523	**0.0652**	**0.0748**
	MSM4SR	**0.1184**	**0.1570**	0.0637	0.0735
Office	S^3-Rec	0.1027	0.1254	0.0641	0.0698
	MSM4SR-V	0.1153	0.1415	0.0764	0.0830
	MSM4SR-T	0.1186	0.1464	0.0772	0.0841
	MSM4SR	**0.1206**	**0.1480**	**0.0797**	**0.0866**

D. Unimodal vs Multimodal Performance

To study the effectiveness of MSM4SR in exploiting different modality information, we consider the following two variants of MSM4SR for evaluation, *i.e.*, 1) **MSM4SR-V**: the text modality information is not used for model fine-tuning (*i.e.*, removing $\mathbf{h}^t(\mathbf{F}^t + \mathbf{E})^\top$ from Eq. (12)); 2) **MSM4SR-T**: the image modality information is not used for model fine-tuning (*i.e.*, removing $\mathbf{h}^v(\mathbf{F}^v + \mathbf{E})^\top$ from Eq. (12)). In MSM4SR-V, MSM4SR-T, and MSM4SR, the model is pre-trained with both text and image modalities. Table IV presents the recommendation performance of MSM4SR-V, MSM4SR-T, and MSM4SR, as well as the best baseline method on each dataset. We can note that MSM4SR-T outperforms MSM4SR-V on all datasets, indicating that text information of items contributes more to performance gain than item images. Leveraging both text and image modality information leads to the best recommendation performance for most datasets. This illustrates the effectiveness of exploiting items' multimodal information for sequential recommendation.

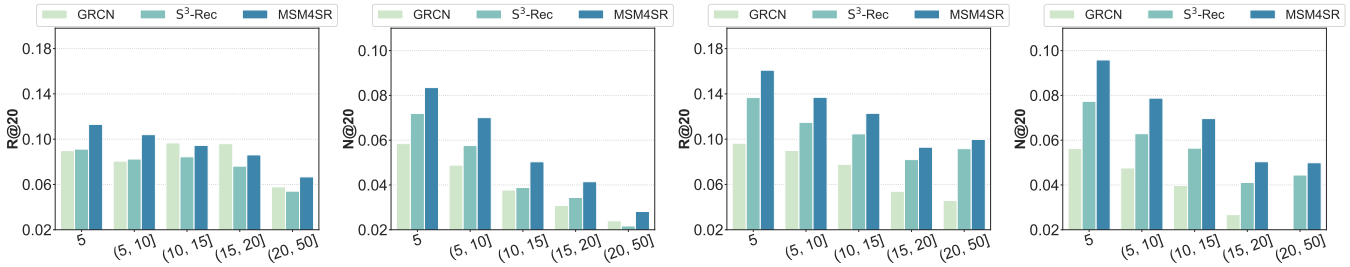
E. Parameter Sensitivity Study

In this experiment, we study the impact of three hyper-parameters, including the λ to balance between modality-wise next item prediction loss and cross-modality contrastive loss, the number of tokens retrieved for each image N, the number of experts used in the MoE architecture O, and the random dropout probability of a sequence ρ. We conduct experiments on Pantry and Office, and report R@20 for comparison.



(a) (b) (c) (d)

Fig. 3. The performance trends of MSM4SR with respect to different settings of λ, N, O, ρ on Pantry and Office datasets.



(a) Pantry-Recall (b) Pantry-NDCG (c) Office-Recall (d) Office-NDCG

Fig. 4. The performance on different user groups achieved by GRCN, S^3-Rec, and MSM4SR on Pantry and Office datasets.

1) Impact of λ: The performance comparison using different values of λ is shown in Figure 3(a). We vary λ in $\{0.001, 0.01, 0.1, 1.0\}$. We can note that the best performance is achieved when λ is set to 0.01. Recommendation performance is compromised with either a large or a small λ.

2) Impact of N: The number of tokens retrieved for each image N is varied in $\{5, 10, 15, 30, 50\}$. As shown in Figure 3(b), 15 word tokens per image appears to be the optimal setting for both datasets. Insufficient information from images is captured when fewer tokens are used, while excessive token usage usually introduces noise.

3) Impact of O: The number of experts used in the MoE architecture O is chosen from $\{1, 2, 4, 8, 16\}$. From Figure 3(c), we can notice the results with respect to O are consistent on both datasets. The best performance is achieved when O is set to 8. However, the further increase of O does not help with the performance.

4) Impact of ρ: As shown in Figure 3(d), we examine the model performance with different dropout probabilities of the user sequence ρ, which ranges from 0 to 0.5 with a step size of 0.1. We can observe that the model performance is relatively stable with the change of the random dropout probability.

F. Recommendation Performance on Different User Groups

Based on the overall performance comparison results shown in Table II, we investigate the influence of data sparsity on users. Specifically, we split all users into five groups according to the length of their interaction sequences, and models are evaluated on each group of users. Figure 4 shows the performance comparison on two datasets. We have the following observations. *Firstly*, the proposed MSM4SR model performs better than GRCN and S^3-Rec on all user groups for Office dataset. For Pantry dataset, both GRCN and MSM4SR outperform S^3-Rec when the user sequences are longer, indicating the usefulness of multimodal features. *Secondly*, compared with GRCN and S^3-Rec, a decrease in the sequence length of user behaviors leads to greater improvements for MSM4SR. This demonstrates that MSM4SR can perform better in more sparse scenarios.

TABLE V
THE RECOMMENDATION PERFORMANCE OF COLD-ITEMS ACHIEVED BY CLCREC, MASR, MSM4SR$_{\text{W/O PRE-TRAIN}}$, AND MSM4SR.

Dataset	Model	R@10	R@20	N@10	N@20
Pantry	MASR	0.0111	0.0119	0.0064	0.0066
	CLCRec	0.0166	0.0251	0.0081	0.0103
	MSM4SR$_{\text{w/o Pre-train}}$	0.0257	0.0337	0.0119	0.0139
	MSM4SR	**0.0360**	**0.0491**	**0.0176**	**0.0208**
Arts	MASR	0.0136	0.0165	0.0080	0.0090
	CLCRec	0.0178	0.0239	0.0101	0.0116
	MSM4SR$_{\text{w/o Pre-train}}$	0.0314	0.0455	0.0145	0.0181
	MSM4SR	**0.0403**	**0.0552**	**0.0191**	**0.0229**
Office	MASR	0.0079	0.0094	0.0050	0.0054
	CLCRec	0.0094	0.0120	0.0049	0.0056
	MSM4SR$_{\text{w/o Pre-train}}$	0.0128	0.0183	0.0061	0.0074
	MSM4SR	**0.0235**	**0.0312**	**0.0117**	**0.0136**

G. Exploration on Two Downstream Tasks

1) Cold-start Item Recommendation Performance: To validate the effectiveness of our model for cold-start recommendation, we consider the following methods for evaluation in addition to **MSM4SR**$_{\text{w/o Pre-train}}$ and **MSM4SR**: 1) **CLCRec** [40]: this method explores the mutual dependency between item multimodal features and collaborative representations to alleviate the cold-start item problem. 2) **MASR** [41]: authors

TABLE VI
THE RECOMMENDATION PERFORMANCE ACHIEVED BY RECGURU, UNISREC, MSM4SR$_{\text{w/o Pre-train}}$, AND MSM4SR UNDER CROSS-DOMAIN SETTING ON PANTRY AND ARTS DATASETS.

Dataset	Model	R@10	R@20	N@10	N@20
Pantry	RecGURU	0.0308	0.0537	0.0152	0.0210
	UniSRec	0.0582	0.0932	0.0265	0.0353
	MSM4SR$_{\text{w/o Pre-train}}$	0.0595	0.0920	0.0286	0.0369
	MSM4SR$_{\text{Cross}}$	**0.0622**	**0.0944**	**0.0294**	**0.0375**
Arts	RecGURU	0.0890	0.1174	0.0569	0.0641
	UniSRec	0.0995	0.1300	0.0565	0.0642
	MSM4SR$_{\text{w/o Pre-train}}$	0.1030	**0.1374**	0.0558	0.0644
	MSM4SR$_{\text{Cross}}$	**0.1041**	0.1367	**0.0573**	**0.0657**

construct two memory banks to store historical user sequences and a retriever-copy network to search similar sequences to enhance the recommendation performance for cold-start items.

In the experiments, counting all items in the training set, we consider those that appear less than 10 times as cold-items, and remaining items are defined as warm-items. CLCRec, MASR, and MSM4SR$_{\text{w/o Pre-train}}$ are trained based on the complete data including both cold-items and warm-cold items, and are evaluated based on user sequences that take cold-items as the target item for prediction. MSM4SR is firstly pre-trained on warm-items and then fine-tuned with the complete data. The model performance is evaluated in the same way as other three baselines. Since cold-items do not have sufficient interaction data, item ID embeddings are discarded at the fine-tuning stage.

Table V shows the performance achieved by CLCRec, MASR, MSM4SR$_{\text{w/o Pre-train}}$, and MSM4SR, on cold-items. We can note that both MSM4SR$_{\text{w/o Pre-train}}$ and MSM4SR outperform two baseline methods, which demonstrate the effectiveness of leveraging multimodal information to alleviate the cold-start item problem. Overall, MSM4SR performs the best by a wide margin across all evaluation metrics. This result shows that cold-items can benefit from self-supervised multimodal pre-training tasks leveraging items with more interactions.

2) Cross-domain Recommendation Performance: We study the knowledge transfer capability of the pre-trained model under the MSM4SR framework. Specifically, we evaluate the cross-domain recommendation performance of MSM4SR, using Office dataset as the source domain, Pantry and Arts datasets as target domains. In the experiments, RecGURU, UniSRec and two variants of MSM4SR are used for comparison: 1) **RecGURU** [42]: this baseline model is a cross-domain sequential recommendation framework that exploits adversarial learning to construct a generalized user representation unified across different domains. 2) **UniSRec** [43]: authors utilize item texts to learn more transferable and universal representations from multiple domains for sequential recommendation. 3) **MSM4SR**$_{\text{w/o Pre-train}}$: we use the target domain data to train the proposed model from scratch based on the multimodal fine-tuning strategy. 4) **MSM4SR**$_{\text{Cross}}$: we use the source domain data to pre-train the MSM4SR framework and fine-tune it based on the target domain data. For UniSRec and MSM4SR variants, we perform parameter-efficient fine-tuning for target domains by fixing the parameters of the Transformer architecture and only fine-tuning the modality encoders.

The results of cross-domain recommendation performance are shown in Table VI. Compared with RecGURU, UniSRec, and MSM4SR$_{\text{w/o Pre-train}}$, MSM4SR$_{\text{Cross}}$ achieves the best performance in terms of most evaluation metrics. This indicates the proposed pre-training framework is effective in transfering knowledge from source domain to target domain. Additionally, the proposed contrastive learning tasks enable MSM4SR to learn generalized multimodal representations for user behavior sequences to benefit sequential recommendation. It is worth noting that RecGURU performs comparably with MSM4SR on the Arts dataset in terms of NDCG but performs the worst on the Pantry dataset. This can be attributed to the fact that Office and Arts have more common users (*i.e.*, 4,068) than Office and Pantry (*i.e.*, 1,525). As a result, RecGURU, which is designed to capture generalized user representations using adversarial learning, is more successful in knowledge transfer from Office to Arts, but fails from Office to Pantry.

V. CONCLUSION

This paper proposes a novel pre-training framework, called MSM4SR (*i.e.*, Multimodal Sequence Mixup for Sequential Recommendation), for boosting the sequential recommendation performance. In MSM4SR, item images are firstly represented by textual tokens to eliminate the discrepancy between text and image modalities. Then, MSM4SR employs a novel backbone network M^{2}SE (*i.e.*, Multimodal Mixup Sequence Encoder) to integrate items' multimodal content with the user behavior sequence, with a complementary sequence mixup strategy. Two contrastive learning losses are designed to help M^{2}SE learn generalized multimodal sequence representations. The experiments on real datasets demonstrate the proposed pre-training framework can help improve sequential recommendation performance under different settings.

ACKNOWLEDGMENT

This research is supported by Alibaba Group through Alibaba Innovative Research (AIR) Program and Alibaba-NTU Singapore Joint Research Institute (JRI), Nanyang Technological University, Singapore.

REFERENCES

[1] S. Wang, L. Hu, Y. Wang, L. Cao, Q. Sheng, and M. Orgun, "Sequential recommender systems: Challenges, progress and prospects," in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, 2019, pp. 6332–6338.

[2] T. Zhang, P. Zhao, Y. Liu, V. S. Sheng, J. Xu, D. Wang, G. Liu, and X. Zhou, "Feature-level deeper self-attention network for sequential recommendation," in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, 2019, pp. 4320–4326.

[3] Y. Liu, S. Yang, C. Lei, G. Wang, H. Tang, J. Zhang, A. Sun, and C. Miao, "Pre-training graph transformer with multimodal side information for recommendation," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 2853–2861.

[4] R. He and J. McAuley, "Vbpr: Visual bayesian personalized ranking from implicit feedback," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2016.

[5] Q. Xu, F. Shen, L. Liu, and H. T. Shen, "Graphcar: Content-aware multimedia recommendation with graph autoencoder," in *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2018, pp. 981–984.

[6] Y. Wei, X. Wang, L. Nie, X. He, R. Hong, and T.-S. Chua, "Mmgcn: Multi-modal graph convolution network for personalized recommendation of micro-video," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 1437–1445.

[7] Y. Wei, X. Wang, L. Nie, X. He, and T.-S. Chua, "Graph-refined convolutional network for multimedia recommendation with implicit feedback," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 3541–3549.

[8] J. Zhang, Y. Zhu, Q. Liu, S. Wu, S. Wang, and L. Wang, "Mining latent structures for multimedia recommendation," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 3872–3880.

[9] X. Han, Z. Zhang, N. Ding, Y. Gu, X. Liu, Y. Huo, J. Qiu, Y. Yao, A. Zhang, L. Zhang *et al.*, "Pre-trained models: Past, present and future," *AI Open*, vol. 2, pp. 225–250, 2021.

[10] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning*. PMLR, 2021, pp. 8748–8763.

[11] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 3982–3992.

[12] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "Factorizing personalized markov chains for next-basket recommendation," in *Proceedings of the Web Conference 2010*, 2010, pp. 811–820.

[13] J. Tang and K. Wang, "Personalized top-n sequential recommendation via convolutional sequence embedding," in *Proceedings of the 11th ACM International Conference on Web Search and Data Mining*, 2018, pp. 565–573.

[14] F. Yuan, A. Karatzoglou, I. Arapakis, J. M. Jose, and X. He, "A simple convolutional generative network for next item recommendation," in *Proceedings of the 12th ACM International Conference on Web Search and Data Mining*, 2019, pp. 582–590.

[15] T. Donkers, B. Loepp, and J. Ziegler, "Sequential user-based recurrent neural network recommendations," in *Proceedings of the 11th ACM Conference on Recommender Systems*, 2017, pp. 152–160.

[16] B. Peng, Z. Ren, S. Parthasarathy, and X. Ning, "Ham: Hybrid associations models for sequential recommendation," *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[17] J. Chang, C. Gao, Y. Zheng, Y. Hui, Y. Niu, Y. Song, D. Jin, and Y. Li, "Sequential recommendation with graph neural networks," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 378–387.

[18] Y. Zhang, Y. Liu, Y. Xu, H. Xiong, C. Lei, W. He, L. Cui, and C. Miao, "Enhancing sequential recommendation with graph contrastive learning," in *Proceedings of the 31st International Joint Conference on Artificial Intelligence*, 2022, pp. 2398–2405.

[19] W.-C. Kang and J. McAuley, "Self-attentive sequential recommendation," in *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2018, pp. 197–206.

[20] F. Sun, J. Liu, J. Wu, C. Pei, X. Lin, W. Ou, and P. Jiang, "Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer," in *Proceedings of the 28th ACM International Conference on Information & Knowledge Management*, 2019, pp. 1441–1450.

[21] K. Zhou, H. Wang, W. X. Zhao, Y. Zhu, S. Wang, F. Zhang, Z. Wang, and J.-R. Wen, "S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization," in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 2020, pp. 1893–1902.

[22] Z. Liu, Z. Fan, Y. Wang, and P. S. Yu, "Augmenting sequential recommendation with pseudo-prior items via reversely pre-training transformer," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 1608–1612.

[23] Y. Xie, P. Zhou, and S. Kim, "Decoupled side information fusion for sequential recommendation," in *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2022, p. 1611–1621.

[24] W.-C. Kang, C. Fang, Z. Wang, and J. McAuley, "Visually-aware fashion recommendation and design with generative image models," in *2017 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2017, pp. 207–216.

[25] S. Liu, Z. Chen, H. Liu, and X. Hu, "User-video co-attention network for personalized micro-video recommendation," in *Proceedings of the Web Conference 2019*, 2019, pp. 3020–3026.

[26] F. Liu, Z. Cheng, C. Sun, Y. Wang, L. Nie, and M. Kankanhalli, "User diverse preference modeling by multimodal attentive metric learning," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 1526–1534.

[27] Q. Wang, Y. Wei, J. Yin, J. Wu, X. Song, and L. Nie, "Dualgnn: Dual graph neural network for multimedia recommendation," *IEEE Transactions on Multimedia*, 2021.

[28] J. Yi and Z. Chen, "Multi-modal variational graph auto-encoder for recommendation systems," *IEEE Transactions on Multimedia*, vol. 24, pp. 1067–1079, 2021.

[29] X. Lin, S. Tiwari, S. Huang, M. Li, M. Z. Shou, H. Ji, and S.-F. Chang, "Towards fast adaptation of pretrained contrastive models for multi-channel video-language retrieval," *arXiv preprint arXiv:2206.02082*, 2022.

[30] N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. Le, G. Hinton, and J. Dean, "Outrageously large neural networks: The sparsely-gated mixture-of-experts layer," in *International Conference on Learning Representations*, 2017.

[31] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, pp. 5998–6008, 2017.

[32] Y. Zhu, Y. Xu, F. Yu, Q. Liu, S. Wu, and L. Wang, "Graph contrastive learning with adaptive augmentation," in *Proceedings of the Web Conference 2021*, 2021, pp. 2069–2080.

[33] J. Ni, J. Li, and J. McAuley, "Justifying recommendations using distantly-labeled reviews and fine-grained aspects," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 188–197.

[34] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang, "Lightgcn: Simplifying and powering graph convolution network for recommendation," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 639–648.

[35] Q. Cui, S. Wu, Q. Liu, W. Zhong, and L. Wang, "Mv-rnn: A multi-view recurrent neural network for sequential recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 2, pp. 317–331, 2018.

[36] Q. Tan, J. Zhang, J. Yao, N. Liu, J. Zhou, H. Yang, and X. Hu, "Sparse-interest network for sequential recommendation," in *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, 2021, pp. 598–606.

[37] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in Neural Information Processing Systems*, pp. 8026–8037, 2019.

[38] W. X. Zhao, S. Mu, Y. Hou, Z. Lin, Y. Chen, X. Pan, K. Li, Y. Lu, H. Wang, C. Tian *et al.*, "Recbole: Towards a unified, comprehensive and efficient framework for recommendation algorithms," in *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 2021, pp. 4653–4664.

[39] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations*, 2015.

[40] Y. Wei, X. Wang, Q. Li, L. Nie, Y. Li, X. Li, and T.-S. Chua, "Contrastive learning for cold-start recommendation," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 5382–5390.

[41] Y. Hu, Y. Liu, C. Miao, and Y. Miao, "Memory bank augmented long-tail sequential recommendation," in *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 2022, pp. 791–801.

[42] C. Li, M. Zhao, H. Zhang, C. Yu, L. Cheng, G. Shu, B. Kong, and D. Niu, "Recguru: Adversarial learning of generalized user representations for cross-domain recommendation," in *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, 2022, pp. 571–581.

[43] Y. Hou, S. Mu, W. X. Zhao, Y. Li, B. Ding, and J.-R. Wen, "Towards universal sequence representation learning for recommender systems," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, pp. 585–593.