# TMMSRec: Time-interval-aware Multi-Modal Sequential Recommender

PINGPING WANG, Jilin University, China
YUANBO XU*, Jilin University, China
YIHENG JIANG, Jilin University, China
HANGTONG XU, Jilin University, China
FUZHEN ZHUANG, Beihang University, China

Conventional multi-modal sequential recommenders usually employ sequence models (e.g., SASRec and GRU4Rec) as the recommender framework for sequential dependency modeling and multi-modal information as add-on knowledge for fine-grained preference learning. However, the existing methods ignore the time interval information in the interaction sequences, which reflects the user's preference transition. Consequently, they acquire biased preference representations, leading to suboptimal performance. Along these lines, we concentrate on incorporating the time interval into the multi-modal sequential recommendation and investigating the internal influences across multiple modalities. Firstly, we treat the time interval as an independent modality and exploit a Time Interval Encoder (TIE) to quantify the time interval in the sequences. Secondly, we consider the heterogeneity between multiple modalities and design a two-stage fusion strategy. The first stage injects time intervals into each modality (ID, image, and text) and the second stage aggregates the user preferences from each modality to get the user's sequence preferences. To this end, we build a model-agnostic framework for the multi-modal sequential recommendation, namely the **T**ime-interval-aware **M**ulti-**M**odal **S**equential **Rec**ommender (short for TMMSRec). The empirical validations on three real-world datasets prove the effectiveness of our method, where the maximum performance improvement against several state-of-the-art baselines achieves up to 16.05%.

CCS Concepts: • **Information systems** → **Recommender systems**.

Additional Key Words and Phrases: Time-interval-aware, Multi-modal Learning, Recommender Systems

**ACM Reference Format:**
Pingping Wang, Yuanbo Xu, Yiheng Jiang, Hangtong Xu, and Fuzhen Zhuang. 2026. TMMSRec: Time-interval-aware Multi-Modal Sequential Recommender. 1, 1 (January 2026), 24 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

## 1 Introduction

Recommenders play a pivotal role in addressing the challenges posed by the data explosion on e-commerce platforms, where the sheer volume and diversity of available information make it difficult to provide personalized and relevant suggestions to users. As these platforms evolve, they increasingly involve large amounts of multimedia data, such as images, text, and even videos, which reflect the diverse nature of user preferences and behaviors. To address this complexity, more and more research [1, 6, 7, 47, 54] has focused on incorporating multi-modal information into

*Corresponding author.

Authors' Contact Information: Pingping Wang, Jilin University, Changchun, China, wangpp23@mails.jlu.edu.cn; Yuanbo Xu, Jilin University, Changchun, China, yuanbox@jlu.edu.cn; Yiheng Jiang, Jilin University, Changchun, China, jiangyh22@mails.jlu.edu.cn; Hangtong Xu, Jilin University, Changchun, China, xuht21@mails.jlu.edu.cn; Fuzhen Zhuang, Beihang University, Beijing, China, zhuangfuzhen@buaa.edu.cn.


Manuscript submitted to ACM

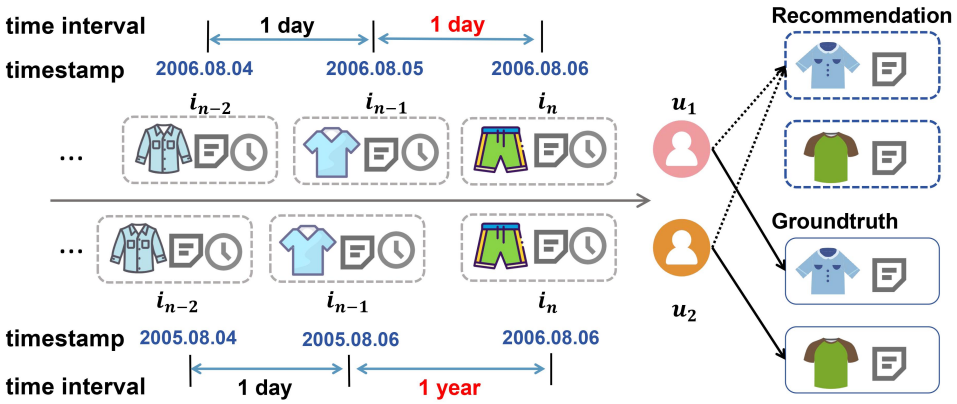


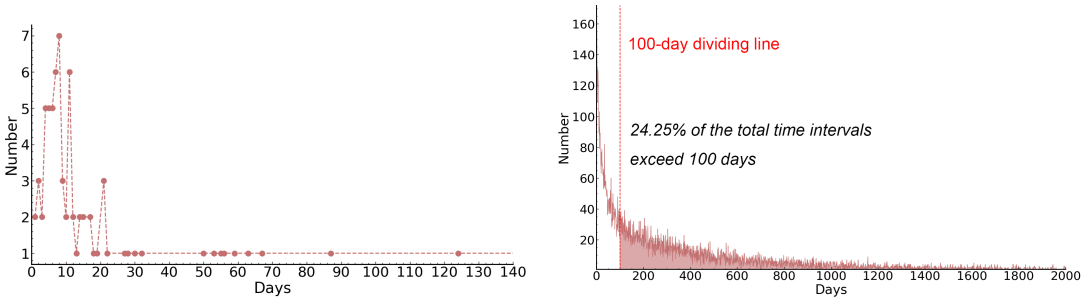

Fig. 1. The influence of interaction time intervals on recommendation results. Two users exhibited the same interaction histories, engaging with *two blue t-shirts* and *a pair of green shorts*, with the only difference being the timestamp of the interaction with $i_n$. Without considering the time interval's effect on learning user preferences, existing recommenders utilize the same sequential order and multi-modal data to predict a similar next item (Recommendation) for both users, which may result in inaccurate results (compared with Groundtruth).

recommender systems. These models aim to integrate various modalities—such as textual descriptions, product images, and user reviews—into a unified framework to enhance the accuracy and relevance of recommendations.

The multi-modal recommendation usually aggregates multi-modal information to obtain richer representations of items. For instance, *VBPR* [11] integrates visual features into the Bayesian Personalized Ranking model, which can capture the preference for image information to improve performance. In addition, *MMGCN* [44] and *MGAT* [38] establish a graph neural network for each modality to learn the specific user and item representations of each modality. To better capture the dynamic preferences of users, recent research [24, 42, 50] considers introducing the interaction order of items in multi-modal recommenders, called multi-modal sequential recommenders. These models capitalize on the sequential order of user-item interactions to model user preferences and utilize multi-modal information to get more accurate item representations or user representations.

One significant limitation arises from the absence of temporal information in the multi-modal recommendation process. The temporal dynamics inherent in user-item interactions offer valuable insights into evolving user preferences on multiple modalities. As depicted in Figure 1, the time interval between the last two interactions of $u_1$ is only one day, while for $u_2$, it is one year. We argue that the larger the time intervals between historical interactions, the more likely user preferences will change [17, 41]. Thus, for $u_1$, historical interactions may reflect his preference for the next item, but this may not hold for $u_2$. Neglecting the time interval may lead to suboptimal recommendation performance, as it fails to account for shifts in user interests and preferences over different periods.

In Figure 2, we investigate the distribution of time intervals in the All_Beauty dataset, and we also calculate different time intervals for the same pair of items in different interaction sequences. In Figure 2 (a), 24.25% of the total time intervals exceed 100 days, which means that many users suddenly have interaction records after a long period of inactivity. We argue that user preferences have changed during this period of inactivity. In Figure 2 (b), we select items with ASIN as *'B00W259T7G'* and *'B0010ZBORW'*, and we observe that the time intervals vary from 1 day to 210 days.




(a) Time intervals in All_Beauty (b) Time intervals of adjacent items in All_Beauty

Fig. 2. The statistics of the time intervals in All_Beauty dataset. Figure (a) represents the time interval between all adjacent items in all sequences. Figure (b) represents the time interval of the same two items in all sequences.

We argue that different time intervals mean different shifts in user preferences. The time intervals between items can be used to dig deeper into the temporal relationship between two items in the interaction sequences.

Moreover, an effective multi-modal fusion method requires consideration of the correlation between the different modalities and how to deal with the differences between the modalities. The crude fusion strategy may result in valuable information loss and failure to fully exploit the complementary nature of multi-modal data. To sum up, relying solely on the interaction order of items in the interaction sequences cannot accurately capture user preferences, and adopting a simple approach to fuse multi-modal information cannot address the inherent heterogeneity between modalities.

Building upon this insight, we propose a model-agnostic framework called **T**ime-interval-aware **M**ulti-**M**odal **S**equential **R**ecommender (TMMSRec) for multi-modal sequential recommendation tasks. TMMSRec treats the time interval as one modality and integrates it into the recommender alongside the other three modalities (ID, text, and image). Specifically, we design the Time Interval Encoder (TIE) to handle the time interval in three levels (day-level, month-level, and year-level). TIE can capture the time difference of three levels to obtain the corresponding independent relative periodic patterns. By integrating temporal information into the recommendation process, models can adeptly adapt to changing user preferences over time. To fully exploit the four modalities, we devise a two-stage fusion strategy. The first stage injects time intervals into each modality to get the user preferences in that specific modality over time. The second stage aggregates the preferences of multiple modalities to obtain the user's sequential preferences. The two-stage strategy can better mine multi-modal information and avoid conflicts between different modalities, thereby improving recommendation accuracy.

To summarize, we make the following contributions in this paper.

- We reveal the importance of time intervals between items in a sequence in the MMSRs domain and propose introducing the time interval as one modality into the multi-modal sequential recommendation model.
- We design the TIE that encodes time intervals comprehensively in three levels, which helps us to precisely capture temporal differences across diverse levels and obtain fine-grained user preferences.
- We propose a two-stage fusion strategy to fuse different modalities in different stages, which can enhance the role of temporal information in user modeling and avoid conflicts between different modalities.
- We conduct experiments on three public datasets to demonstrate the effectiveness of our proposed TMMSRec, which improves the performance of TMMSRec compared to other baseline models by up to 16.05%.

The organization of this paper is as follows: In Section 2, we introduce the basic definitions and provide a detailed description of the TMMSRec method, including its key components and design rationale. Section 3 presents the experimental validation of our approach, where we conduct extensive evaluations on several public datasets and discuss the results. In Section 4, we review related work about positional encoding, sequential recommendation, and multi-modal recommendation. Finally, Section 5 concludes the paper, summarizing our findings and suggesting potential avenues for future research.

## 2 Methodology

In the previous section, we discussed the importance of time intervals in MMSRs and highlighted the shortcomings of current multi-modal information fusion methods. To further improve the performance of MMSRs, we propose TMMSRec, whose structure is shown in Figure 3. In this section, we formally describe the Top-K multi-modal sequential recommender and describe the computational process of each module in detail. Table 1 lists all the important notation and corresponding descriptions.

### 2.1 Notations

We define the set of users as $U = \{u\}$ and the set of items as $I = \{i\}$. For each item $i$ in the interaction sequences, we record its ID, image, text, and interaction timestamp as $i^{id}$, $i^{im}$, $i^{te}$, and $i^{ti}$. The interaction sequence of user $u$ on modality $m$ is defined in chronological order as $S_u^m = \{i_1^m, i_2^m, \ldots, i_n^m\}$, where $n$ is the maximum sequence length. The complete notation table is placed in Table 1.

### 2.2 Problem Definition

Given a multi-modal historical interaction sequence, which includes different types of data such as ID, text, image, and the corresponding timestamp, the MMSRs are based on these data to learn the user's interest patterns and preference evolution laws. Its task is to realize the prediction of the items that the user may be interested in the future. It can be described as the following equation:

$$TopK_u = \text{MMSRec}(S_u^m), \tag{1}$$

where $TopK_u$ is the Top-K recommendation list containing items that user $u$ may interact with. $S_u^m$ is the historical interaction sequence of user $u$ on modality $m$. MMSRec$(\cdot)$ is an abstract function representation symbol to signify any MMSRs, which takes the user's historical interaction sequence and multi-modal data as input and obtains a list of Top-K items that the user may interact with.

### 2.3 Item Embedding Learning

Given the multi-modal data corresponding to each item $(i^{id}, i^{te}, i^{im}, i^{ti})$, we use two pre-trained feature extractors to extract text features and image features. We adopt ResNet50 [9] as our image feature extractor and Recformer [21] as our text feature extractor. The corresponding ID features are typically extracted from the learnable embedding table, usually initialized randomly. The extraction process can be summarized as follows:

$$\mathbf{E}^{te} = \text{TextExtractor}(I^{te}), \tag{2}$$

$$\mathbf{E}^{im} = \text{ImageExtractor}(I^{im}), \tag{3}$$

$$\mathbf{E}^{id} = \text{Embedding}(I^{id}), \tag{4}$$

Table 1. Notations.

| Notations | Descriptions |
|---|---|
| $U, u$ | User set, user |
| $I, i$ | Item set, item |
| $i_{id}, i_{im}, i_{te}, i_{ti}$ | ID, image, text, interaction timestamp of item $i$ |
| $S, s$ | Interaction sequence set, interaction sequence |
| $m$ | Modality |
| $n$ | Maximum sequence length of interaction sequence |
| $i_t^m$ | Item on modality $m$ at $t$-th time step |
| $S_u^m$ | Interaction sequence of user $u$ on modality $m$ |
| $d$ | Embedding dimension |
| $\mathbf{e}_i^m$ | Embedding of item $i$ on modality $m$ |
| $\mathbf{E}^m$ | Embedding of all items on modality $m$ |
| $\tilde{\mathbf{S}}^{ti,m}$ | Positional embedding of user interaction sequences |
| $\tilde{\mathbf{S}}^m$ | Modality embedding of user interaction sequences |
| $P_{i^{ti},\Theta}$ | Time aware positional encoding matrix |
| $i_d, i_h, i_y$ | Day, month, year of interaction |
| $\hat{\mathbf{S}}^m$ | Time-interval-aware embedding of interaction sequences |
| $\mathbf{F}^m$ | User preference representations on modality $m$ |
| $r_{u,i}^m$ | The preference score of user $u$ for item $i$ on modality $m$ |
| $r_{u,i}$ | Preference score of user $u$ for item $i$ |
| $o_t, g_t$ | Output and negative item at $t$-th time step |



where $\mathbf{E}^{te} \in \mathbb{R}^{|I|\times d}$, $\mathbf{E}^{im} \in \mathbb{R}^{|I|\times d}$, and $\mathbf{E}^{id} \in \mathbb{R}^{|I|\times d}$ denote text feature matrix, image feature matrix, and ID embedding matrix respectively. Besides, $d$ is the dimension of the image feature, text feature, and ID embedding. In addition, we utilize our proposed Time Interval Encoder (TIE) to obtain positional embedding. The encoding process can be summarized as follows:

$$\tilde{\mathbf{S}}^{ti,m} = \text{TimeIntervalEncoder}(S^m, \mathbf{E}^m, S^{ti}), \tag{5}$$

where $\tilde{\mathbf{S}}^{ti,m} \in \mathbb{R}^{|U|\times n\times d}$ represents the positional embedding corresponding to the user interaction sequence set $S^m$, and $m \in \{id, te, im\}$. For each user $u$, the positional embedding of interacted item $i$ on modality $m$ can be represented by $\mathbf{e}_i^{ti,m}$.

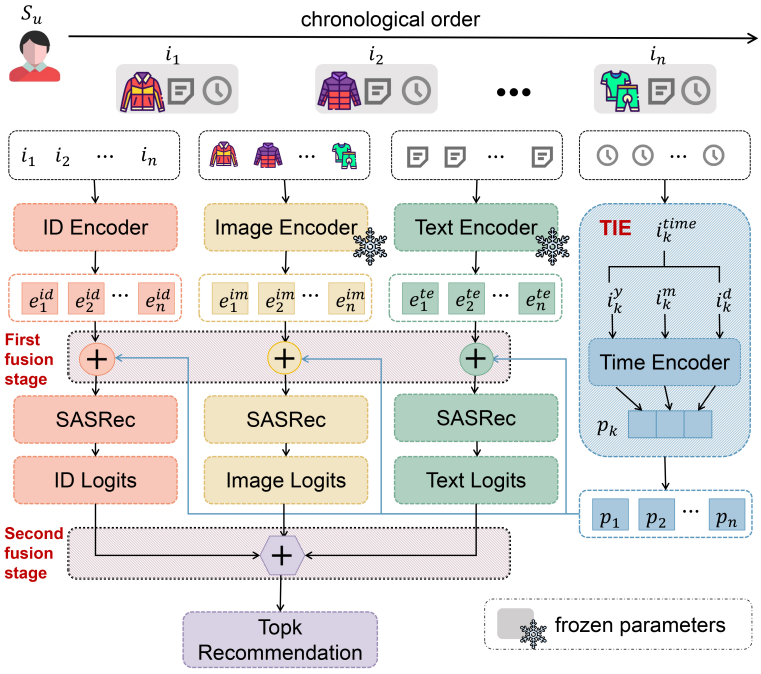



Fig. 3. The framework of our proposed TMMSRec.

### 2.4 Time Interval Encoder (TIE)

Time Interval Encoder (TIE) is inspired by the previous research [17, 41]. Compared to encoding 1-D temporal information [17], TIE encodes 3-D temporal information, processing day-level, month-level, and year-level temporal information. Firstly, we decouple the timestamp $i^{ti}$ corresponding to the item by day, month, and year as $i^d$, $i^h$, and $i^y$.

Given a user interaction sequence $S_u^m$, we can find the corresponding modality embedding from the modality embedding matrix, as follows:

$$\tilde{\mathbf{S}}_u^m = \text{LookUp}(S_u^m, \mathbf{E}^m), \tag{6}$$

where $\tilde{\mathbf{S}}_u^m \in \mathbb{R}^{|U| \times n \times d}$ represents the modality embedding of interaction sequence, and $m \in \{id, te, im\}$. For each item $i$ in the interaction sequences, we can obtain the modality embedding $\mathbf{e}_i^m$ and calculate the positional embedding. TIE injects temporal information as follows:

$$\mathbf{e}_i^{ti,m} = \mathbf{e}_i^m \cdot \mathbf{P}_{i^{ti},\Theta}, \tag{7}$$

where $\mathbf{e}_i^{ti,m}, \mathbf{e}_i^m \in \mathbb{R}^{1 \times d}$, and $d$ is exactly divisible by 6. In addition, $\Theta = \{\theta_j, j = [1, 2, \ldots, d/6]\}$ is the pre-defined parameters, $\mathbf{P}_{i^{ti},\Theta}$ is the time aware positional encoding matrix, $i^{ti}$ is the timestamp of item $i$ in this interaction sequence,

and $\mathbf{e}_i^{ti,m}$ is the positional embedding of item $i$ on modality $m$. Specifically, $\mathbf{P}_{i^{ti},\Theta}$ follows the diagonal form like:

$$\mathbf{P}_{i^{ti},\Theta} = \begin{bmatrix} \mathbf{P}_{i^{ti},\theta_1} & \mathbf{O} & \cdots & \mathbf{O} \\ \mathbf{O} & \mathbf{P}_{i^{ti},\theta_2} & \cdots & \mathbf{O} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{O} & \mathbf{O} & \cdots & \mathbf{P}_{i^{ti},\theta_{d/6}} \end{bmatrix}, \tag{8}$$

where $\mathbf{O} \in \mathbb{R}^{6\times 6}$ is zero matrix. Each sub-matrix $\mathbf{P}_{i^{ti},\theta_j} \in \mathbb{R}^{6\times 6}$ aims at injecting the temporal information of different levels into different latent sub-spaces of dimension $d$, as follows:

$$\mathbf{P}_{i^{ti},\theta_j} = \begin{bmatrix} \cos(i^y\theta_j) & -\sin(i^y\theta_j) & 0 & 0 & 0 & 0 \\ \sin(i^y\theta_j) & \cos(i^y\theta_j) & 0 & 0 & 0 & 0 \\ 0 & 0 & \cos(i^h\theta_j) & -\sin(i^h\theta_j) & 0 & 0 \\ 0 & 0 & \sin(i^h\theta_j) & \cos(i^h\theta_j) & 0 & 0 \\ 0 & 0 & 0 & 0 & \cos(i^d\theta_j) & -\sin(i^d\theta_j) \\ 0 & 0 & 0 & 0 & \sin(i^d\theta_j) & \cos(i^d\theta_j) \end{bmatrix}, \tag{9}$$

where $\theta_j = 10000^{-6(j-1)/d}$, $j \in [1, 2, \ldots, d/6]$. We require an embedding dimension $d$ that is divisible by 6 because of $\mathbf{P}_{i^{ti},\Theta}$ consists of $d/6$ 6x6 sub-matrices. Only when $d$ is divisible by 6 can these sub-matrices fill the entire embedding space completely and uniformly.

In conclusion, using sinusoidal transformation, TIE simultaneously encodes day-level, month-level, and year-level time intervals into $d$-dimensional vectors. In the complex space, the sinusoidal transformation can rotate the vector by $\Theta$ according to the mold length $i^{\{d,h,y\}}$. This method not only preserves the hierarchical nature of time but also maintains the relative relationships between different time intervals, enabling the model to discern subtle temporal dependencies in user behavior. This positional coding method can provide a richer temporal feature basis for constructing dynamic user portraits.

### 2.5 User Sequence Modeling

Given user historical interaction sequences $S^m$ on specific modality $m$, we obtain the time-interval-aware embedding of user interaction sequences $\hat{\mathbf{S}}^m$ in the first fusion stage.

Sequential recommenders obtain user representations from historical interactions. We adopt SASRec [19] as our backbone to model user sequences. It applies the self-attention mechanism to look for relevant items from historical interactions and uses them to predict the next item. Using time-interval-aware embedding of the user interaction sequence, we obtain user preference representations as follows:

$$\mathbf{F}^m = \text{SASRec}^m(\hat{\mathbf{S}}^m, \Psi_{\text{SASRec}^m}), \tag{10}$$

where $\hat{\mathbf{S}}^m \in \mathbb{R}^{|U|\times n\times d}$, $\mathbf{F}^m \in \mathbb{R}^{|U|\times d}$, and $d$ is the dimension of the user preference representations, the same as the ID, text, and image features. $\mathbf{F}^m$ represents the user preference representations on modality $m$. $\text{SASRec}^m$ denotes the user encoder used for user modeling on modality $m$, and $\Psi_{\text{SASRec}^m}$ denotes all the learnable parameters in $\text{SASRec}^m$.

### 2.6 Two-Stage Multi-Modal Fuison

*2.6.1 The First Fusion Stage.* The first fusion stage aims to inject time intervals into three modalities to learn the user preferences over time on each modality. For each item, we add the positional embedding to the modality embedding as

the time-interval-aware modality representation corresponding to the user interaction sequence, as follows:

$$\hat{\mathbf{S}}^m = \tilde{\mathbf{S}}^m + \tilde{\mathbf{S}}^{ti,m}. \tag{11}$$

*2.6.2 The Second Fusion Stage.* The second fusion stage merges the preferences of each modality to obtain the overall preferences. After we obtain modality embedding $\mathbf{e}_i^m$ and user preference representation $\mathbf{f}_u^m$, we can calculate the prediction score on modality $m$ as follows:

$$r_{u,i}^m = (\mathbf{f}_u^m)^\top \mathbf{e}_i^m, \tag{12}$$

where $r_{u,i}^m$ is the preference score of user $u$ for item $i$ on modality $m$. To combine the preferences of each modality, we add the preference scores of modalities as follows:

$$r_{u,i} = \sum_{m \in \{id,te,im\}} r_{u,i}^m, \tag{13}$$

where $r_{u,i}$ is the overall preference score of user $u$ for item $i$.

## 2.7 Training

In every training epoch, we randomly pick one negative item for each time step in each user sequence. These negative items represent items that users have not interacted with before. We use binary cross-entropy loss as our objective function. At each time step, the loss penalizes low probabilities for positive and high probabilities for negative interactions, excluding $< pad >$ placeholders. This approach helps our model learn from positive and negative examples in user interaction sequences, improving its ability to predict future interactions. The formula is as follows:

$$\mathcal{L} = -\sum_{S_u \in S} \sum_{t \in [1,2,\ldots,n]} \log\left(\sigma(r_{u,o_t})\right) + \sum_{g_t \notin S_u} \log\left(1 - \sigma(r_{u,g_t})\right), \tag{14}$$

where $t$ is the time step, $o_t$ is the expected output at $t$-th time step, and $g_t$ is the negative item at $t$-th time step. The pseudo-code for our entire algorithm is in Algorithm 1. Our code is available at: https://anonymous.4open.science/r/TMMSRec-81E7

**Algorithm 1** TMMSRec

**Input:** User historical interaction sequence $S$, multi-modal information of historical interacted item $I^{id}$, $I^{im}$, $I^{te}$, and $I^{ti}$.
**Output:** Trained model $f_\psi$.

```
Randomly initialize parameters of model f_ψ
for epoch in 1,...,max epochs do
    Generate one negative item for each time step.
    for batch in 1,...,batch number do
        Generate the embedding on three modalities E^id, E^im, and E^te by Eq.(1), Eq.(2) and Eq.(3).
        Generate the positional embedding of three modalities from TIE by Eq.(4).
        Fuse the positional embedding with three modalities by Eq.(11).
        Learn user preference representations on three modalities by Eq.(10).
        Calculate the preference score by Eq.(12).
        Fuse the multi-modal preference score by Eq.(13).
        Calculate the loss by Eq.(14) and update f_ψ.
return Trained model f_ψ.
```

## 3 Validations and Discussions

In the previous section, we presented the technical details and key advantages of our proposed multi-modal recommendation method. To thoroughly validate its effectiveness and provide deeper insights into its superior performance, we conduct extensive experiments to investigate the following four research questions:

- **RQ1**: Does our proposed TMMSRec perform better than state-of-the-art baselines?
- **RQ2**: How do the components in the framework contribute to performance?
- **RQ3**: How do the different positional encoders and multi-modal fusion methods affect performance?
- **RQ4**: How do the different backbones affect performance?
- **RQ5**: How do the hyperparameters affect performance?

### 3.1 Datasets

We use three datasets to evaluate the performance of each method: All_beauty, Luxury_Beauty, and Prime_Pantry. These datasets come from an e-commerce platform named Amazon[1], and the three datasets respectively represent the three product categories on the e-commerce platform. In these public datasets, most items have corresponding images and text information, and we filter out items that lack images and text information. In addition, we also filter the user sequence with fewer than five interactions. The details of the three datasets are presented in Table 2. We select the ' *category*,' ' *brand*,' and ' *title*' of the item as the text information, while we select the ' *imageURLHighRes*' as the image information. For items with multiple images, we only keep the first image as the image information of the item. We use two pre-trained modality encoders to extract modality features. For partitioning, we split the user sequence into three parts: (1) the most recent interaction for testing, (2) the second most recent interaction for validation, and (3) all remaining interactions for training. When validating, the input sequence includes the training interactions, and similarly, when testing, the input sequence includes training and validation interactions.

### 3.2 Evaluation Metrics

We primarily employ next-item prediction tasks to assess the effectiveness of each method. To evaluate the performance of each method, we adopt Hit Rate@$K$ (HR@$K$) and Normalized Discounted Cumulative Gain@$K$ (NDCG@$K$) as the metrics and set $K$=10 by default, where the metrics are commonly used in the recommendation tasks. The calculation formula of the two metrics is as follows:

$$\text{HR@}K = \frac{1}{S}\sum_{i=1}^{S}\text{hit}(i), \tag{15}$$

$$\text{NDCG@}K = \frac{1}{S}\sum_{i=1}^{S}\frac{1}{\log_2(\text{p}(i)+1)}, \tag{16}$$

where $S$ represents the size of the set of items expected to be recommended, hit$(\cdot)$ represents whether the $i$-th requirement item is included in the list of items recommended by the model, and p$(i)$ represents the position of the $i$-th requirement item in the list of items recommended by the model. HR@$K$ emphasizes the accuracy of model recommendations, that is, whether the user's requirement items are included in the model's recommendation items, while NDCG@$K$ emphasizes the position of the user's requirement items in the model's recommendation list, with the higher the position, the better. Given that each user has only one item for testing, HR@$K$ is equivalent to Recall@$K$ and is closely related

[1]https://nijianmo.github.io/amazon/index.html

Table 2. Statistics of three experimental datasets

| Dataset | #Users | #Items | #Interactions | #*sparsity* |
|---|---|---|---|---|
| *All_Beauty* | 273 | 800 | 1,663 | *99.24%* |
| *Luxury_Beauty* | 5,230 | 5,196 | 43,682 | *99.84%* |
| *Prime_Pantry* | 14,893 | 7,787 | 144,354 | *99.88%* |

to Precision@$K$. Our ranking strategy adopts the full-rank approach, wherein all items are ranked according to the predicted scores of the recommendation model, and the top@$K$ items are selected for recommendations.

### 3.3 Baselines

To evaluate our proposed TMMSRec, we select three groups of baselines: the sequential recommendation models, the multi-modal recommendation models, and the multi-modal sequential recommendation models. We briefly describe these models as follows:

- **SASRec** [19]: This model uses a self-attention mechanism to capture long-term semantics like an RNN, but makes predictions based on relatively few actions, similar to a Markov Chain. It balances model complexity and efficiency by leveraging attention to identify relevant items from a user's action history.
- **Bert4Rec** [35]: This model uses a bidirectional transformer to capture users' sequential preferences, addressing the limitations of unidirectional RNNs by conditioning on both left and right context. It employs the Cloze task to train the model, allowing for more diverse training samples and stronger sequential representations.
- **TriMLP** [16]: This model introduces a Triangular Mixer in an MLP-like architecture to enhance cross-token communication by simplifying the operation and blocking anti-chronological connections. It alternates between global and local mixing to capture both long-range dependencies and short-term preferences.
- **VBPR** [11]: This model introduces a scalable factorization approach that integrates visual signals from product images, leveraging pre-trained deep networks to extract visual features. By uncovering the visual dimensions that influence user feedback, it enhances personalized ranking accuracy and helps address cold start problems.
- **VBPR$_{\mathbf{ITV}}$**: The text modality is introduced on the original VBPR model, and the text modality adopts the same fusion method as the image modality.
- **S3-Rec** [53]: This model employs self-supervised learning with a self-attentive architecture to improve sequential recommendation. It uses four auxiliary self-supervised objectives based on mutual information maximization (MIM) to enhance data representations, particularly addressing data sparsity.
- **S3-Rec$_{\mathbf{ITV}}$**: The image modality is introduced on the original S3-Rec model, and the image modality adopts the same fusion method as the text modality.
- **MGAT** [38]: This model leverages Graph Neural Networks (GNNs) on multi-modal interaction graphs to adaptively capture user preferences across different modalities. By using a gated attention mechanism, it assigns varying importance scores to each modality, allowing for more precise and robust user-item representations.
- **MAML** [27]: This model introduces Multi-modal Attentive Metric Learning (MAML) to capture users' diverse preferences by using an attention network that integrates multi-modal item features. It leverages metric learning

Table 3. The results of six models on three datasets, with the best results bolded and the second best results underlined. The "Paired t-test p value" is calculated based on the 10 repetitions of TMMSRec and the second best baseline for each metric.

| Dataset | All_beauty | | Luxury_beauty | | Prime_Pantry | |
|---|---|---|---|---|---|---|
| Methods | HR@10 ↑ | NDCG@10 ↑ | HR@10 ↑ | NDCG@10 ↑ | HR@10 ↑ | NDCG@10 ↑ |
| SASRec [19] | 0.3553 | 0.2646 | <u>0.2703</u> | 0.1526 | 0.0597 | 0.0400 |
| Bert4Rec [35] | 0.3370 | 0.2366 | 0.1954 | 0.1261 | 0.0309 | 0.0164 |
| TriMLP [16] | 0.2472 | 0.1622 | 0.0433 | 0.0201 | 0.0152 | 0.0075 |
| VBPR [11] | 0.2637 | 0.1640 | 0.1099 | 0.0459 | 0.0152 | 0.0074 |
| $\text{VBPR}_{\text{ITV}}$ | 0.1831 | 0.1344 | 0.0819 | 0.0384 | 0.0170 | 0.0103 |
| S3-Rec [53] | 0.1795 | 0.1324 | 0.1346 | 0.1028 | 0.0535 | 0.0296 |
| $\text{S3-Rec}_{\text{ITV}}$ | 0.1319 | 0.0923 | 0.0929 | 0.0801 | 0.0479 | 0.0257 |
| MGAT [38] | 0.2491 | 0.1839 | 0.1696 | 0.0834 | 0.0136 | 0.0070 |
| MAML [27] | 0.3407 | 0.1840 | 0.0281 | 0.0109 | 0.0035 | 0.0015 |
| Freedom [55] | 0.1941 | 0.1421 | 0.1816 | 0.1141 | 0.0199 | 0.0102 |
| MMMLP [25] | <u>0.3700</u> | <u>0.2984</u> | 0.2676 | <u>0.1902</u> | <u>0.0598</u> | <u>0.0449</u> |
| TMMSRec (Ours) | **0.4139** | **0.3113** | **0.2757** | **0.1914** | **0.0694** | **0.0452** |
| *Improved* | *11.86%* | *4.32%* | *2.00%* | *0.63%* | *16.05%* | *0.67%* |
| *Paired t-test p value* | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ |

to overcome the limitations of dot product similarity and outperforms state-of-the-art methods on large-scale datasets.

- **FREEDOM** [55]: This model utilizes a frozen item-item graph structure and denoises the user-item interaction graph to enhance recommendations. It employs a degree-sensitive edge pruning method to remove noisy interactions, simplifying user preference learning, and is grounded in graph theory, ensuring efficiency in multi-modal recommendation tasks.
- **MMMLP** [25]: This model introduces an MLP-based architecture for sequential recommendation, efficiently processing multi-modal sequences. It leverages a Feature Mixer Layer, Fusion Mixer Layer, and Prediction Layer to combine and predict user preferences while maintaining linear computational complexity.

### 3.4 Implementation Details

We implement our work in PyTorch. In our model, the network is optimized by the Adam optimizer [20], which is a variant of Stochastic Gradient Descent (SGD) with adaptive moment estimation. Additionally, we use the Exponential Learning Rate Schedule to dynamically adjust the learning rate according to the training epoch, where the gamma is set to 0.9. For all baselines, we set the feature dimension as 128 and the batch size as 200. We set the maximum length of the interaction sequence to 50, and the interaction sequences less than 50 are filled with $< pad >$.

### 3.5 Performance Comparison Analysis (RQ1)

This study compares our proposed TMMSRec with several baselines on three datasets. Table 3 shows the results of different methods on all datasets. We have the following observations:

- Our proposed method performs best on the three datasets, proving that the TIE and two-stage fusion strategy are effective. Introducing time intervals as the fourth modality enhances the temporal relation between items to obtain fine-grained user preferences, improving recommendation performance. The two-stage fusion strategy considers heterogeneity between multiple modalities and avoids resulting conflicts between them.
- MMMLP has the second-best performance on three datasets. MMMLP introduces multi-modal information based on MLP4Rec. By introducing multi-modal information, MMMLP can process richer and more diverse data to obtain more comprehensive user behavior characteristics. Therefore, MMMLP can better understand user behavior characteristics and make more accurate recommendations during the recommendation process.
- SASRec, Bert4Rec, and TriMLP are all models for sequential recommendation tasks. They are good at extracting semantic and contextual information from user historical interaction sequences to better understand user interaction and preferences and make personalized recommendations. Compared to MMMLP, they do not use multi-modal information, resulting in inferior performance.
- MAML and MGAT use multi-modal information but do not take advantage of sequential information. In recommendation tasks, sequential information is often critical because users' behaviors have chronological order, and the order of items in the sequence may impact the user preferences. Therefore, MAML and MGAT may not perform as well as SASRec and BERT4Rec on some datasets.
- VBPR introduces image features into the traditional recommender, while S3-Rec introduces text features to the traditional recommender. By comparing the original model with the full modality version, we can find that simply adding modality information does not necessarily improve the recommendation performance. Multi-modal recommendation models require effective multi-modal processing methods to address various potential issues that may arise from modality fusions.

### 3.6 Ablation Analysis (RQ2)

In this study, we analyze the impact of four modalities and the TIE on the performance of our proposed TMMSRec model. To compare the performance of our model with other variants, we prepared seven different variants, including:

- ***w/* ID**, which only utilizes ID embedding as the input of the user encoder.
- ***w/* ID&TIE**, which utilizes the fusion result of ID embedding and positional embedding as the input of the user encoder.
- ***w/* ID&T**, which utilizes ID embedding and text feature as the input of two user encoders.
- ***w/* ID&T&TIE**, which utilizes the fusion result of ID embedding and positional embedding, and the fusion result of text feature and positional embedding as the input of two user encoders, respectively
- ***w/* ID&V**, which utilizes ID embedding and image feature as the input of two user encoders.
- ***w/* ID&V&TIE**, which utilizes the fusion result of ID embedding and positional embedding and the fusion result of image feature and positional embedding as the input of two user encoders.
- ***w/* ID&T&V**, which utilizes ID embedding, text feature, and image feature as the input of three user encoders.

Table 4 shows the results of these variants performing the sequential recommendation task on three datasets. First, introducing multimedia (text and image) features can improve performance compared to only ID embedding,

Table 4. The results of the ablation study with key components, with the best results bolded and the second best results underlined. The "Paired t-test p value" is calculated based on the 10 repetitions of TMMSRec and the second best baseline for each metric.

| Dataset | All_beauty | | Luxury_beauty | | Prime_Pantry | |
|---|---|---|---|---|---|---|
| Methods | HR@10 ↑ | NDCG@10 ↑ | HR@10 ↑ | NDCG@10 ↑ | HR@10 ↑ | NDCG@10 ↑ |
| *w/* ID | 0.3810 | 0.2941 | 0.2683 | 0.1862 | 0.0596 | 0.0396 |
| *w/* ID&TIE | 0.3407 | 0.2547 | 0.2704 | 0.1892 | 0.0638 | 0.0422 |
| *w/* ID&T | 0.3810 | 0.2989 | 0.2711 | 0.1881 | 0.0669 | 0.0429 |
| *w/* ID&T&TIE | 0.3883 | 0.2990 | 0.2625 | 0.1739 | **0.0699** | **0.0457** |
| *w/* ID&V | 0.3810 | 0.3010 | 0.2650 | 0.1818 | 0.0677 | 0.0440 |
| *w/* ID&V&TIE | 0.3883 | 0.2990 | <u>0.2746</u> | <u>0.1911</u> | 0.0254 | 0.0127 |
| *w/* ID&V&T | <u>0.4029</u> | <u>0.3062</u> | 0.2633 | 0.1845 | 0.0687 | 0.0442 |
| TMMSRec (Ours) | **0.4139** | **0.3113** | **0.2757** | **0.1914** | <u>0.0694</u> | <u>0.0452</u> |
| *Improved* | *3.00%* | *11.86%* | *4.32%* | *1.54%* | *2.00%* | *0.63%* |
| *Paired t-test p value* | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ | $< 0.01$ |

indicating the necessity of multi-modal information. In addition, introducing the time interval as the fourth modality is valid in most cases, which means that using the time interval to quantify the temporal relation between items in the interaction sequence can help us obtain fine-grained user preferences over time. It is worth noting that only using the ID embedding as the input to TMMSRec lacks positional embedding compared to the vanilla SASRec, which results in improved performance on two datasets. A similar phenomenon was also observed in the paper [19], where removing positional embedding can improve performance on some datasets. The SASRec without positional embedding ignores the interaction order and may be better suited for shorter user sequences.

### 3.7 Components Analysis (RQ3)

*3.7.1 Different Positional Encoders Comparison Analysis.* In this study, we analyze and evaluate the impact of replacing the TIE with several different positional encoders that use only order relations in historical interaction sequences rather than time intervals on the performance of TMMSRec. We prepared three variants depending on the different positional encoders, including:

- **TMMSRec-NP**, which does not utilize positional embedding.
- **TMMSRec-RP**, which uses no regular or fixed function to generate positional embedding, only randomly.
- **TMMSRec-VP**, which utilizes vanilla positional embedding, derived from the Transformer [40] model.

Table 5 shows the results of different positional encoders on three datasets. Our proposed TMMSRec works best on three datasets, which proves that introducing the time interval is necessary. Our proposed TIE can effectively use the time interval to capture time differences at different levels and obtain fine-grained user preferences. All three variants only use order relation in the sequence rather than time information, which results in a degradation in their performance. On Luxury_Beauty and Prime_Pantry, TMMSRec-NP performs better than TMMSRec-RP and TMMSRec-VP, indicating that inappropriate positional embedding can harm the model's performance. In most cases, TMMSRec-RP performs

Table 5. The results of different positional encoders, with the best results bolded and the second best results underlined. The "Paired t-test p value" is calculated based on the 10 repetitions of TMMSRec and the second best baseline for each metric.

| Dataset | All_beauty | | Luxury_beauty | | Prime_Pantry | |
|---|---|---|---|---|---|---|
| Methods | HR@10 ↑ | NDCG@10 ↑ | HR@10 ↑ | NDCG@10 ↑ | HR@10 ↑ | NDCG@10 ↑ |
| TMMSRec-NP | 0.3553 | 0.2801 | <u>0.2633</u> | <u>0.1845</u> | <u>0.0596</u> | <u>0.0396</u> |
| TMMSRec-RP | <u>0.3663</u> | <u>0.2811</u> | 0.2149 | 0.1355 | 0.0358 | 0.0198 |
| TMMSRec-VP | 0.3333 | 0.2420 | 0.1811 | 0.0980 | 0.0361 | 0.0256 |
| TMMSRec (Ours) | **0.4139** | **0.3113** | **0.2757** | **0.1914** | **0.0694** | **0.0452** |
| *Improved* | *12.99%* | *10.74%* | *4.71%* | *3.74%* | *16.44%* | *14.14%* |
| *Paired t-test p value* | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 |

Table 6. The results of different multi-modal fusions, with the best results bolded and the second best results underlined. The "Paired t-test p value" is calculated based on the 10 repetitions of TMMSRec and the second best baseline for each metric.

| Dataset | All_beauty | | Luxury_beauty | | Prime_Pantry | |
|---|---|---|---|---|---|---|
| Methods ↑ | HR@10 ↑ | NDCG@10 ↑ | HR@10 ↑ | NDCG@10 ↑ | HR@10 ↑ | NDCG@10 ↑ |
| TMMSRec-add | <u>0.3480</u> | 0.2580 | 0.2109 | 0.1321 | <u>0.0440</u> | <u>0.0306</u> |
| TMMSRec-linear | 0.3333 | <u>0.2626</u> | <u>0.2310</u> | <u>0.1446</u> | 0.0377 | 0.0197 |
| TMMSRec-mlp | 0.3370 | 0.2438 | 0.2011 | 0.1111 | 0.0377 | 0.0189 |
| TMMSRec (Ours) | **0.4139** | **0.3113** | **0.2757** | **0.1914** | **0.0694** | **0.0452** |
| *Improved* | *18.93%* | *18.55%* | *19.35%* | *32.37%* | *57.73%* | *47.71%* |
| *Paired t-test p value* | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 | < 0.01 |

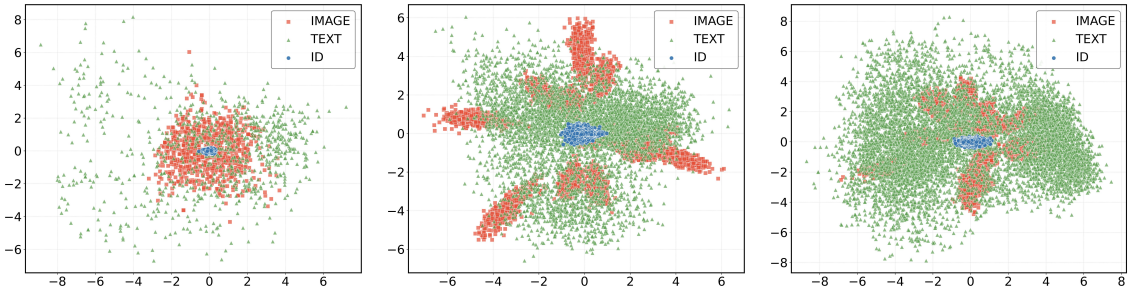


better than TMMSRec-VP because the randomly generated positional embedding introduces noise that helps the model generalize better.

*3.7.2 Different Fusion Methods Comparison Analysis.* In this study, we analyze and evaluate the impact of different fusion methods on the performance of our proposed TMMSRec model. We use the early fusion as comparisons and input the results of the fusion of the four modalities into the user encoder. We prepared three variants depending on the different early fusion methods, including:

- **TMMSRec-add**, which adds the four modalities together and then inputs the fusion results into the user encoder.
- **TMMSRec-linear**, which splices the multi-modal embedding and processes them with a linear layer.
- **TMMSRec-mlp**, which splices the multi-modal embedding and processes them with a multilayer perceptron [31].

In Table 6, we show the results of different early fusion methods on three datasets. Our proposed TMMSRec works best on three datasets, so our two-stage fusion strategy is valid. The two-stage fusion strategy uses different treatments for different modalities to avoid conflicts between modalities. For TMMSRec-add, it may ignore the interactions between




(a) All_Beauty (b) Luxury_Beauty (c) Prime_Pantry

Fig. 4. Visualization of multi-modal features

modalities, limiting the model's learning ability. For TMMSRec-linear, it may not be able to handle nonlinear relations well, which limits the expressiveness of the model. TMMSRec-mlp has more advantages when dealing with nonlinear relations. The performance of the three variants on the datasets is different due to the different modality feature distributions of the datasets.

*3.7.3 Multi-modal Features Visualization Analysis.* In this study, we conduct a visualization of the multi-modal features of three datasets to demonstrate the necessity of the two-stage fusion strategy. As shown in Figure 4, we can observe that different modalities exhibit different distributions, which indicates that the multi-modal features are heterogeneous. This heterogeneity is essentially due to the fact that the three modalities employ different feature extraction methods. The ID embedding is randomly initialized, while the text features and image features are obtained through different pre-trained encoders. The two-stage fusion strategy alleviates the adverse effects caused by heterogeneity by splitting the preference learning of the three modalities.

*3.7.4 Ensemble Analysis.* In this study, we aim to investigate whether the observed performance improvement of the model is attributable to the model ensemble approach. To achieve this, we prepare variants based on two ID-based recommendation methods, including:

- **SASRec$_{\text{semble}}$**, which uses three SASRecs to learn user representations on three modalities, respectively.
- **TriMLP$_{\text{semble}}$**, which uses three TriMLPs to learn user representations on three modalities, respectively.

To ensure the fairness of the comparison, the ensemble models adopt the same ensemble method as TMMSRec, that is, using three models to learn user preferences on specific modalities, and finally fusing the preference scores on the three modalities. We also adopt the same addition method as our fusion method of modality preference scores. We then compare the performance of these ensemble models with TMMSRec and the original models, with the results summarized in Figure 5.

As shown in Table 5, the ensemble model can bring about performance improvement to a certain extent, but this improvement is limited and unstable. Notably, the ensemble model consistently underperforms relative to TMMSRec, which achieves better results overall. This behavior can be explained by the fact that when multiple models in the ensemble learn similar patterns, it introduces redundancy and reduces diversity, which is crucial for improving performance. Additionally, too much similarity among models can increase the risk of overfitting, especially with larger

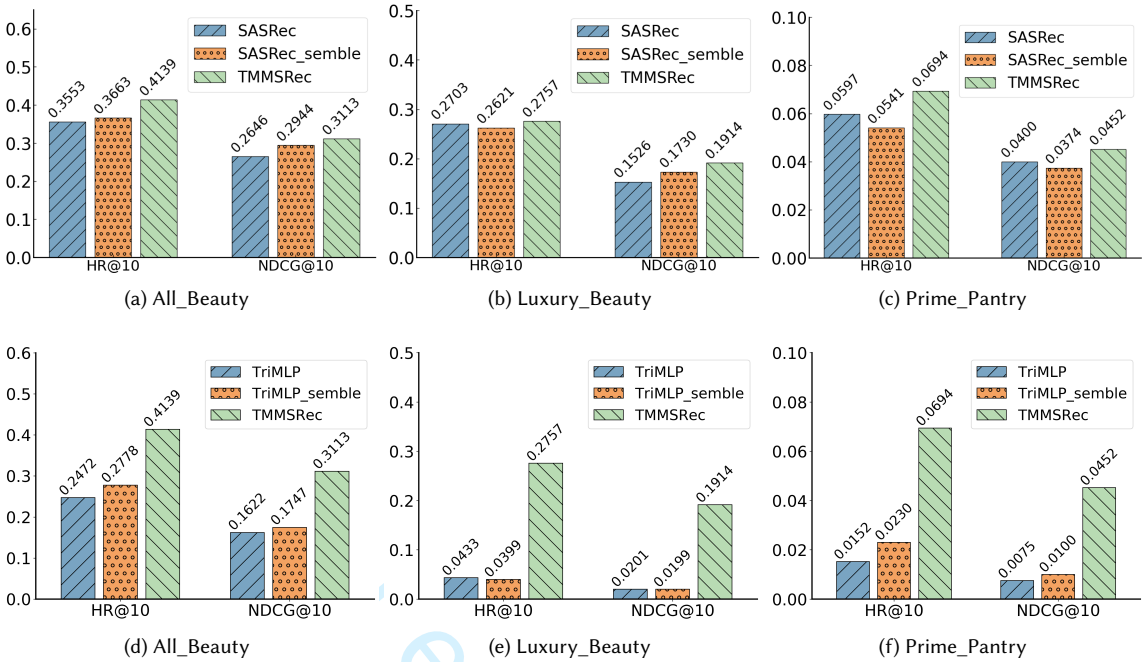



Fig. 5. Comparison of different ensemble models

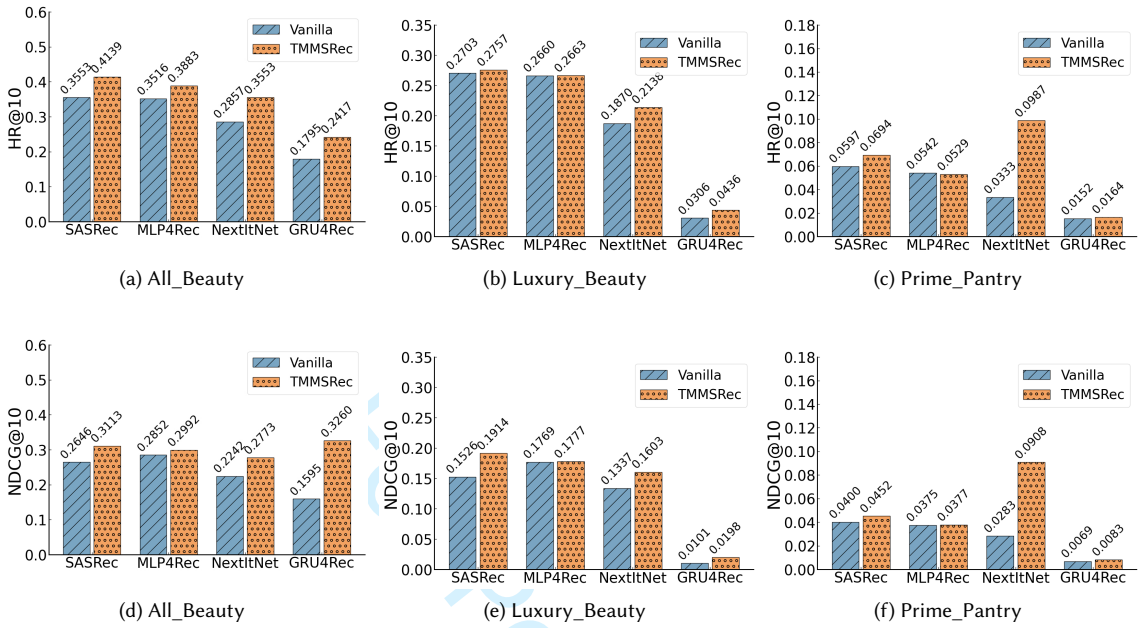
datasets. Thus, while ensemble methods can be beneficial in certain cases, they may not always improve performance, particularly when models lack diversity or overfitting becomes an issue with larger datasets.

### 3.8 Backbone Analysis (RQ4)

This study analyzes the proposed model-agnostic framework, TMMSRec, to verify its universality across different backbones. We selected four models as our backbones in TMMSRec, namely SASRec [19], MLP4Rec [23], NextItNet [48], and GRU4Rec [12]. The codes used in this experiment come from MICRecBox[2], where the loss functions are uniformly replaced with the loss function in SASRec. The parameters of the four backbones are consistent, such as the learning rate set to 0.001, the dropout rate set to 0.5, the sequence max length set to 50, and the feature dimension set to 128. The experimental results are shown in Figure 6. And we have the following observations:

- In most cases, the models with the TMMSRec framework outperform those without the TMMSRec framework. This proves the effectiveness of our framework. The model performance can be improved effectively by introducing multimedia and temporal information.
- We can observe that GRU4Rec with TMMSRec has the greatest improvement compared to vanilla GRU4Rec because the recurrent neural structure of vanilla GRU4Rec is suitable for dealing with long-term dependencies in sequences. After introducing the time interval, GRU4Rec is more likely to capture positional relations in the sequence. In addition to this, the inclusion of multimedia information also improves model performance.

[2]https://github.com/MICLab-Rec/RecBox




(a) All_Beauty (b) Luxury_Beauty (c) Prime_Pantry

(d) All_Beauty (e) Luxury_Beauty (f) Prime_Pantry

Fig. 6. Comparison of different backbones

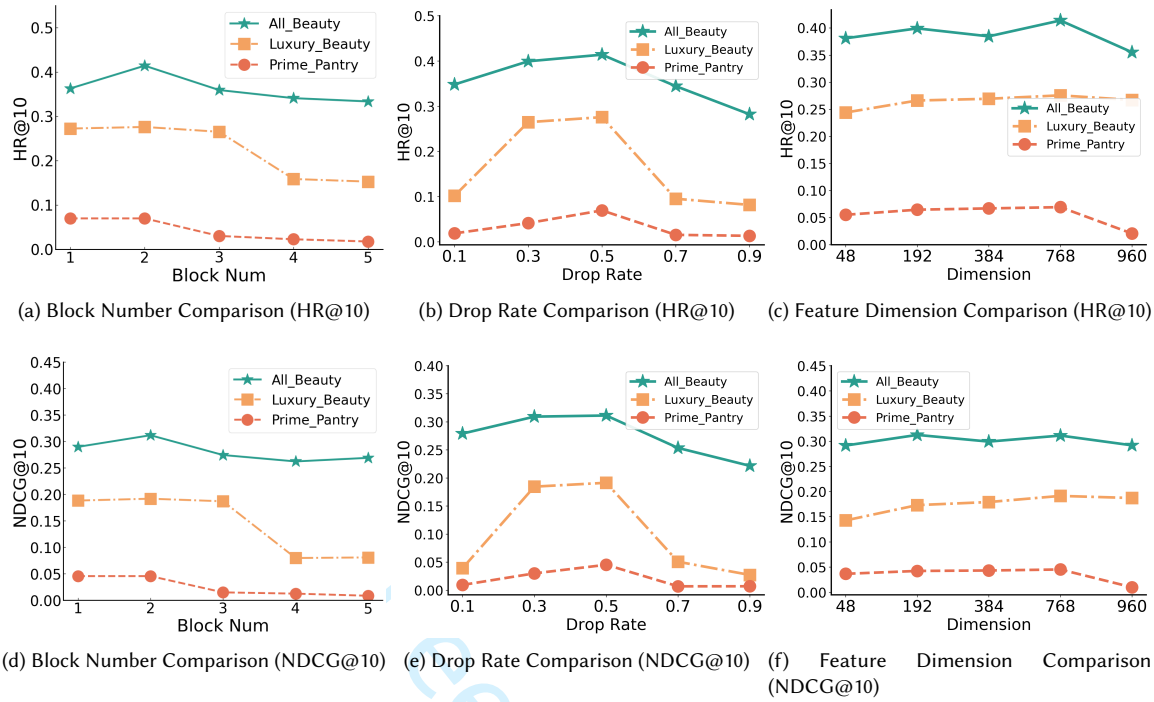
- Additionally, the MLP4Rec with TMMSRec has a minor improvement compared to vanilla MLP4Rec because vanilla MLP4Rec is a simple feedforward neural network that is not sensitive to temporal information in the sequence, so the improvement is relatively small.

### 3.9 Hyperparameter Analysis (RQ5)

In this study, we not only investigate the drop rate inside the TIE module on the performance of TMMSRec, but also investigate the impact of varying the number of blocks in the SASRec model on the performance of TMMSRec.

*3.9.1 Block Number Analysis.* As shown in Figure 7 (a) and (d), when SASRec adopts a two-layer module structure, the TMMSRec model exhibits the optimal performance indicators. It is notable that when the number of modules is further increased to 3 layers or more, the model performance shows a significant downward trend. This performance degradation mainly stems from two factors: Firstly, the deep architecture is prone to causing overfitting of the model on the training set, that is, overfitting the noise features in the training data; Secondly, an increase in the number of parameters will introduce more redundant calculations, affecting the reasoning efficiency of the model. The ablation experiment confirmed that the two-layer architecture achieved the best balance between model capacity and generalization ability. It can not only fully capture the temporal characteristics in the user behavior sequence, but also maintain good generalization performance.

*3.9.2 Drop Rate Analysis.* As shown in Figure 7 (b) and (e), model performance exhibits a unimodal trend with increasing dropout rates, peaking at 0.5 before declining. Specifically, when the discard rate gradually increased from 0 to 0.5, the model performance continued to improve. This phenomenon confirms that moderate discarding can effectively inhibit




(a) Block Number Comparison (HR@10) (b) Drop Rate Comparison (HR@10) (c) Feature Dimension Comparison (HR@10)

(d) Block Number Comparison (NDCG@10) (e) Drop Rate Comparison (NDCG@10) (f) Feature Dimension Comparison (NDCG@10)

Fig. 7. Hyperparameter analysis experiments

the cooperative adaptation of neurons and enhance the generalization ability of the model. However, when the discard rate exceeds 0.5, the performance begins to decline significantly. This indicates that an excessively high discard rate will disrupt the critical path of feature extraction, resulting in the model's inability to effectively learn the essential laws of user behavior. Based on the comprehensive experimental results, a discard rate setting of 0.5 can optimally balance the requirements of feature retention and regularization.

*3.9.3 Feature Dimension Analysis.* As shown in Figure 7 (c) and (f), the model performs the best when the feature dimension is 768. If the feature dimension is too small, it will lead to information compression and reduce the model's expressive ability; if the dimension is too large, it will introduce noise, increase the computational burden, and cause overfitting. We hope to select the smallest dimension while retaining sufficient multi-modal information, thereby reducing the burden of the model.

## 4 Related Work

### 4.1 Positional Encoding

In sequence modeling tasks, such as natural language processing and time series analysis, the model must effectively capture the sequential relationship of the input elements. Traditional recurrent neural networks (RNNs) and Long Short-Term Memory networks (LSTMs) implicitly encode location information through their inherent temporal computing mechanisms. However, with the wide application of the *Transformer* [40], the self-attention mechanism has become

mainstream due to its efficient parallel computing capabilities. However, self-attention itself has permutation invariance, that is, it cannot distinguish the input order at different positions. Therefore, researchers introduced the Positional Embedding technique to explicitly encode position information, enabling the model to perceive the sequential structure of the sequence.

The Transformer proposes and compares two positional embedding methods. One of them mainly adopts learnable random initialization vectors, that is, assigns independent embedding parameters to each position and optimizes them during the training process. Its advantage lies in its simple implementation and adaptability to the requirements of different tasks. However, this randomly initialized positional embedding has obvious limitations, especially its poor generalization ability for long sequences. Since positional embedding is only optimized for sequences of fixed length during training, when encountering longer inputs during testing, the model may not be able to handle unseen position information effectively. Furthermore, this method lacks explicit modeling of the relative relationships between positions, making it difficult for the model to learn distance-related inductive biases.

Another positional encoding method proposed in the Transformer is sinusoidal positional encoding. This method uses predefined trigonometric functions (sine and cosine) to generate positional embeddings, so that the embeddings at different positions have unique patterns while maintaining a certain smoothness. An important feature of sinusoidal position coding is that it can achieve the encoding of relative positions through linear transformation, thereby supporting the model extrapolation to longer sequences to a certain extent. However, although sinusoidal coding has good generalization in theory, in practical applications, its fixed mathematical form may limit the model's adaptability to the data distribution of different tasks. Furthermore, since sinusoidal positional encoding is an absolute position coding, it may still have limitations when dealing with long-range dependencies.

In recent years, researchers have focused on improving positional embedding methods. Among them, Rotary Position Embedding (RoPE) [17, 41] receives extensive attention due to its excellent performance. The core idea of RoPE is to transform the Query and Key vectors through the rotation matrix, thereby naturally introducing the relative position information in the self-attention calculation. Compared with absolute position coding, RoPE can model the relative relationships between different positions more flexibly and shows stronger generalization ability in long sequence tasks. Experiments show that RoPE significantly outperforms traditional positional embedding methods in tasks such as language modeling and machine translation, especially having obvious advantages when dealing with extremely long sequences. Furthermore, RoPE has a relatively high computational efficiency and is compatible with existing Transformer optimization techniques, making it one of the mainstream position coding schemes in current large language models [3, 39].

### 4.2 Squential Recommendation

Traditional collaborative filtering-based recommenders [8, 30] typically rely on the assumption that user preferences can be inferred from their past interactions with items, focusing mainly on historical data to predict future behavior. However, this approach overlooks the fact that user preferences are not static, and they evolve as users interact with new items and experience changing contexts. This realization has led to sequential recommendation models, which aim to capture the dynamic nature of user preferences by incorporating contextual information derived from the sequences of user-item interactions. These models are designed to handle the temporal aspects of user behavior, allowing for more accurate and context-aware recommendations.

Previous methods [10, 33] of sequential recommendation were primarily based on Markov chain models, which assumed that the future behaviors of users depended on only a few of their recent interactions. With the development of

deep learning technology, researchers began to explore the application of deep learning methods [23, 37] in sequential recommendation. Among them, models [12, 13] based on recurrent neural networks (RNN) and long short-term memory networks (LSTM) have become mainstream, which can effectively capture long-term dependencies in interaction sequences. Later, the self-attention mechanism has been widely used in sequential recommendation [29, 45], such as *SASRec* [19] and *Bert4Rec* [35]. *SASRec* and *Bert4Rec* dynamically learn the importance of different positions in the sequence to better capture the user preference evolution. TriMLP [16] introduces an MLP-based architecture for sequential recommendation, using a Triangular Mixer with global and local layers to enforce chronological order and capture long- and short-term dependencies. Introducing these models enriches the choice of sequential recommendation methods and provides recommender systems with more possibilities and flexibility.

More sequential recommenders have begun incorporating side information to enhance recommendation accuracy further. For instance, *FDSA* [52] integrates textual information such as item types and brands. However, *NOVA* [26] argues that direct fusion of side information can be invasive. As a solution, they propose using side information as a bootstrap rather than directly fusing it. Additionally, *S3Rec* [53] enhances data representation by learning correlations between attributes, items, sub-sequences, and sequences, thus improving the effectiveness of sequence recommendations. While these models undoubtedly enhance recommendation accuracy, they often utilize multi-modal information, including various data types such as images, text, and audio.

### 4.3 Multi-modal Recommendation

Traditional recommendations are based on IDs [19, 46]. Subsequently, with the development of multimedia platforms, multi-modal recommendation systems emerge. Multi-modal recommendation systems have gained significant attention in recent years as they incorporate diverse types of information, such as images [18, 43], text [22, 36], attributes [4, 28], and audio—into traditional recommendation models. This approach aims to improve the quality and accuracy of recommendations by providing more affluent, more comprehensive representations of items and users. This integration of multi-modal information typically involves a fusion process, broadly categorized into two main methods: early fusion and late fusion [14, 15, 32, 51]. Early fusion combines multi-modal features at an early stage, usually through simple operations such as addition or concatenation. In contrast, late fusion combines the learning results of each modality at a later stage.

Previous studies, such as *VBPR* [11], use early fusion by integrating image features with ID embedding before preference prediction. Similarly, *MMGCN* [44] utilizes the interaction information of each modality to construct modality-specific graphs that facilitate the learning of the representations of items and users on each modality and ultimately fuse the representations of all modalities. *MGAT* [38] introduces the attention mechanism based on *MMGCN*, which dynamically adjusts the attention weights according to the importance of nodes in different graphs. *MAML* [27] concatenates text and image features and then feeds them into a multi-layer neural network for fusion in a way that has been used in many works [34, 49]. *MILK* uses invariant learning to ensure that user content preferences remain stable and consistent, even in environments where specific modalities of items are missing. In the realm of sequential recommenders, researchers have recognized the significance of multi-modal information in item representation learning. For instance, *MV-RNN* [5] adopts three distinct methods to fuse ID, text, and image features, generating comprehensive item representations. These representations are employed to capture user representations through recurrent structures. Similarly, *M3SRec* [2] leverages modality-specific mixture-of-experts layers to capture sequential patterns on each modality and subsequently fuses multi-modal information using a cross-modal mixture-of-experts layer. *PMMRec*

[24] removes ID embeddings completely from the recommendation process and still achieves good results using only multi-modal information.

In addition to conventional early and late fusion methods, recent research has increasingly focused on developing more sophisticated fusion techniques to better integrate multi-modal information in recommendation systems. *ODMT* [15] introduces an ID-aware Multi-modal Transformer module designed to fuse multi-modal information, which mitigates conflicts among different features through masks. These innovative multi-modal fusion methods promise to enhance recommendation systems by effectively leveraging diverse types of information.

## 5 Concluding Remarks

In this paper, we propose a model-agnostic framework for multi-modal sequential recommendation tasks, namely TMMSRec, which aims to handle the impact of the time interval on user preference modeling that is ignored in traditional MMSRs. TMMSRec introduces the TIE to quantify the time interval in the sequence at three levels and employs a two-stage fusion strategy to avoid conflicts between different modalities. Experimental results demonstrate that TMMSRec improves performance by up to 16.05% compared to state-of-the-art models on three public datasets, proving the effectiveness of our approach. Additionally, we examine the significance of key components and confirm the universality of our framework across different backbones.

In future work, we will investigate more effective feature engineering methods and advanced representation learning techniques to better extract and characterize multi-modal features of users and items, capturing their complex associations and user preferences more accurately. Additionally, we will introduce richer auxiliary information and contextual signals to further improve model performance and generalization capabilities.



## References


[1] Haoyue Bai, Le Wu, Min Hou, Miaomiao Cai, Zhuangzhuang He, Yuyang Zhou, Richang Hong, and Meng Wang. 2024. Multimodality Invariant Learning for Multimedia-Based New Item Recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Washington DC, USA) *(SIGIR '24)*. Association for Computing Machinery, New York, NY, USA, 677–686. doi:10.1145/3626772.3658596

[2] Shuqing Bian, Xingyu Pan, Wayne Xin Zhao, Jinpeng Wang, Chuyuan Wang, and Ji-Rong Wen. 2023. Multi-modal Mixture of Experts Represetation Learning for Sequential Recommendation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management* (, Birmingham, United Kingdom,) *(CIKM '23)*. Association for Computing Machinery, New York, NY, USA, 110–119. doi:10.1145/3583780.3614978

[3] Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, USVSN Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. 2022. GPT-NeoX-20B: An Open-Source Autoregressive Language Model. arXiv:2204.06745 [cs.CL] https://arxiv.org/abs/2204.06745

[4] Jingwu Chen, Fuzhen Zhuang, Xin Hong, Xiang Ao, Xing Xie, and Qing He. 2018. Attention-driven Factor Model for Explainable Personalized Recommendation. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval* (Ann Arbor, MI, USA) *(SIGIR '18)*. Association for Computing Machinery, New York, NY, USA, 909–912. doi:10.1145/3209978.3210083

[5] Qiang Cui, Shu Wu, Qiang Liu, Wen Zhong, and Liang Wang. 2020. MV-RNN: A Multi-View Recurrent Neural Network for Sequential Recommendation. *IEEE Transactions on Knowledge and Data Engineering* 32, 2 (2020), 317–331. doi:10.1109/TKDE.2018.2881260

[6] Wenzhe Du, Su Haoyang, Nguyen Cam-Tu, and Jian Sun. 2023. Enhancing Product Representation with Multi-form Interactions for Multimodal Conversational Recommendation. In *Proceedings of the 31st ACM International Conference on Multimedia* (, Ottawa ON, Canada,) *(MM '23)*. Association for Computing Machinery, New York, NY, USA, 6491–6500. doi:10.1145/3581783.3613755

[7] Junchen Fu, Xuri Ge, Xin Xin, Alexandros Karatzoglou, Ioannis Arapakis, Jie Wang, and Joemon M. Jose. 2024. IISAN: Efficiently Adapting Multimodal Representation for Sequential Recommendation with Decoupled PEFT. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Washington DC, USA) *(SIGIR '24)*. Association for Computing Machinery, New York, NY, USA, 687–697. doi:10.1145/3626772.3657725

[8] Wei Guo, Fuzhen Zhuang, Xiao Zhang, Yiqi Tong, and Jin Dong. 2024. A comprehensive survey of federated transfer learning: challenges, methods and applications. *Front. Comput. Sci.* 18, 6 (July 2024), 34 pages. doi:10.1007/s11704-024-40065-x

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 770–778. doi:10.1109/CVPR.2016.90

[10] Ruining He, Wang-Cheng Kang, and Julian McAuley. 2018. Translation-based recommendation: a scalable method for modeling sequential behavior. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence* (Stockholm, Sweden) *(IJCAI'18)*. AAAI Press, 5264–5268.

[11] Ruining He and Julian McAuley. 2016. VBPR: visual Bayesian Personalized Ranking from implicit feedback. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence* (Phoenix, Arizona) *(AAAI'16)*. AAAI Press, 144–150.

[12] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2016. Session-based Recommendations with Recurrent Neural Networks. arXiv:1511.06939 [cs.LG]

[13] Balázs Hidasi, Massimo Quadrana, Alexandros Karatzoglou, and Domonkos Tikk. 2016. Parallel Recurrent Neural Network Architectures for Feature-rich Session-based Recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems* (Boston, Massachusetts, USA) *(RecSys '16)*. Association for Computing Machinery, New York, NY, USA, 241–248. doi:10.1145/2959100.2959167

[14] Yupeng Hou, Zhankui He, Julian McAuley, and Wayne Xin Zhao. 2023. Learning Vector-Quantized Item Representation for Transferable Sequential Recommenders. In *Proceedings of the ACM Web Conference 2023* (, Austin, TX, USA,) *(WWW '23)*. Association for Computing Machinery, New York, NY, USA, 1162–1171. doi:10.1145/3543507.3583434

[15] Wei Ji, Xiangyan Liu, An Zhang, Yinwei Wei, Yongxin Ni, and Xiang Wang. 2023. Online Distillation-enhanced Multi-modal Transformer for Sequential Recommendation. In *Proceedings of the 31st ACM International Conference on Multimedia* (, Ottawa ON, Canada,) *(MM '23)*. Association for Computing Machinery, New York, NY, USA, 955–965. doi:10.1145/3581783.3612091

[16] Yiheng Jiang, Yuanbo Xu, Yongjian Yang, Funing Yang, Pengyang Wang, Chaozhuo Li, Fuzhen Zhuang, and Hui Xiong. 2024. TriMLP: A Foundational MLP-Like Architecture for Sequential Recommendation. *ACM Trans. Inf. Syst.* 42, 6, Article 157 (Oct. 2024), 34 pages. doi:10.1145/3670995

[17] Yiheng Jiang, Yongjian Yang, Yuanbo Xu, and En Wang. 2024. Spatial-Temporal Interval Aware Individual Future Trajectory Prediction. *IEEE Transactions on Knowledge and Data Engineering* 36, 10 (2024), 5374–5387. doi:10.1109/TKDE.2023.3332929

[18] Yannis Kalantidis, Lyndon S. Kennedy, and Li-Jia Li. 2013. Getting the look: clothing recognition and segmentation for automatic product suggestions in everyday photos. *Proceedings of the 3rd ACM conference on International conference on multimedia retrieval* (2013). https://api.semanticscholar.org/CorpusID:1357489

[19] Wang-Cheng Kang and Julian McAuley. 2018. Self-Attentive Sequential Recommendation. In *2018 IEEE International Conference on Data Mining (ICDM)*. 197–206. doi:10.1109/ICDM.2018.00035

[20] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *CoRR* abs/1412.6980 (2014). https://api.semanticscholar.org/CorpusID:6628106

[21] Jiacheng Li, Ming Wang, Jin Li, Jinmiao Fu, Xin Shen, Jingbo Shang, and Julian McAuley. 2023. 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* (, Long Beach, CA, USA,) *(KDD '23)*. Association for Computing Machinery, New York, NY, USA, 1258–1267. doi:10.1145/3580305.3599519

[22] Jiacheng Li, Ming Wang, Jin Li, Jinmiao Fu, Xin Shen, Jingbo Shang, and Julian McAuley. 2023. 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* (Long Beach, CA, USA) *(KDD '23)*. Association for Computing Machinery, New York, NY, USA, 1258–1267. doi:10.1145/3580305.3599519

[23] Muyang Li, Xiangyu Zhao, Chuan Lyu, Minghao Zhao, Runze Wu, and Ruocheng Guo. 2022. MLP4Rec: A Pure MLP Architecture for Sequential Recommendations. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, Lud De Raedt (Ed.). International Joint Conferences on Artificial Intelligence Organization, 2138–2144. doi:10.24963/ijcai.2022/297 Main Track.

[24] Youhua Li, Hanwen Du, Yongxin Ni, Pengpeng Zhao, Qi Guo, Fajie Yuan, and Xiaofang Zhou. 2024. Multi-Modality is All You Need for Transferable Recommender Systems. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*. 5008–5021. doi:10.1109/ICDE60146.2024.00380

[25] Jiahao Liang, Xiangyu Zhao, Muyang Li, Zijian Zhang, Wanyu Wang, Haochen Liu, and Zitao Liu. 2023. MMMLP: Multi-modal Multilayer Perceptron fornbsp;Sequentialnbsp;Recommendations. In *Proceedings of the ACM Web Conference 2023* (<conf-loc>, <city>Austin</city>, <state>TX</state>, <country>USA</country>, </conf-loc>) *(WWW '23)*. Association for Computing Machinery, New York, NY, USA, 1109–1117. doi:10.1145/3543507.3583378

[26] Chang Liu, Xiaoguang Li, Guohao Cai, Zhenhua Dong, Hong Zhu, and Lifeng Shang. 2021. Noninvasive self-attention for side information fusion in sequential recommendation. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 4249–4256.

[27] Fan Liu, Zhiyong Cheng, Changchang Sun, Yinglong Wang, Liqiang Nie, and Mohan Kankanhalli. 2019. User Diverse Preference Modeling by Multimodal Attentive Metric Learning. In *Proceedings of the 27th ACM International Conference on Multimedia* (Nice, France) *(MM '19)*. Association for Computing Machinery, New York, NY, USA, 1526–1534. doi:10.1145/3343031.3350953

[28] Fan Liu, Zhiyong Cheng, Lei Zhu, Chenghao Liu, and Liqiang Nie. 2022. An Attribute-Aware Attentive GCN Model for Attribute Missing in Recommendation. *IEEE Transactions on Knowledge and Data Engineering* 34, 9 (2022), 4077–4088. doi:10.1109/TKDE.2020.3040772

[29] Qiao Liu, Yifu Zeng, Refuoe Mokhosi, and Haibin Zhang. 2018. STAMP: Short-Term Attention/Memory Priority Model for Session-based Recommendation. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (London, United Kingdom) *(KDD '18)*. Association for Computing Machinery, New York, NY, USA, 1831–1839. doi:10.1145/3219819.3219950

[30] Huishi Luo, Fuzhen Zhuang, Ruobing Xie, Hengshu Zhu, Deqing Wang, Zhulin An, and Yongjun Xu. 2024. A survey on causal inference for recommendation. *The Innovation* 5, 2 (2024), 100590. doi:10.1016/j.xinn.2024.100590

[31] J. Nievergelt. 1969. R69-13 Perceptrons: An Introduction to Computational Geometry. *IEEE Trans. Comput.* C-18, 6 (1969), 572–572. doi:10.1109/T-C.1969.222718

[32] Xingyu Pan, Yushuo Chen, Changxin Tian, Zihan Lin, Jinpeng Wang, He Hu, and Wayne Xin Zhao. 2022. Multimodal Meta-Learning for Cold-Start Sequential Recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management* (Atlanta, GA, USA) *(CIKM '22)*. Association for Computing Machinery, New York, NY, USA, 3421–3430. doi:10.1145/3511808.3557101

[33] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing personalized Markov chains for next-basket recommendation. In *Proceedings of the 19th International Conference on World Wide Web* (Raleigh, North Carolina, USA) *(WWW '10)*. Association for Computing Machinery, New York, NY, USA, 811–820. doi:10.1145/1772690.1772773

[34] Nitish Srivastava and Ruslan Salakhutdinov. 2012. Multimodal learning with deep Boltzmann machines. In *Journal of machine learning research*. https://api.semanticscholar.org/CorpusID:710430

[35] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (Beijing, China) *(CIKM '19)*. Association for Computing Machinery, New York, NY, USA, 1441–1450. doi:10.1145/3357384.3357895

[36] Yunzhi Tan, Min Zhang, Yiqun Liu, and Shaoping Ma. 2016. Rating-boosted latent topics: understanding users and items with ratings and reviews. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence* (New York, New York, USA) *(IJCAI'16)*. AAAI Press, 2640–2646.

[37] Jiaxi Tang and Ke Wang. 2018. Personalized Top-N Sequential Recommendation via Convolutional Sequence Embedding. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining* (Marina Del Rey, CA, USA) *(WSDM '18)*. Association for Computing Machinery, New York, NY, USA, 565–573. doi:10.1145/3159652.3159656

[38] Zhulin Tao, Yinwei Wei, Xiang Wang, Xiangnan He, Xianglin Huang, and Tat-Seng Chua. 2020. MGAT: Multimodal Graph Attention Network for Recommendation. *Information Processing Management* 57, 5 (2020), 102277. doi:10.1016/j.ipm.2020.102277

[39] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. LLaMA: Open and Efficient Foundation Language Models. arXiv:2302.13971 [cs.CL] https://arxiv.org/abs/2302.13971

[40] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Long Beach, California, USA) *(NIPS'17)*. Curran Associates Inc., Red Hook, NY, USA, 6000–6010.

[41] En Wang, Yiheng Jiang, Yuanbo Xu, Liang Wang, and Yongjian Yang. 2022. Spatial-Temporal Interval Aware Sequential POI Recommendation. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. 2086–2098. doi:10.1109/ICDE53745.2022.00202

[42] Jinpeng Wang, Ziyun Zeng, Yunxiao Wang, Yuting Wang, Xingyu Lu, Tianxiang Li, Jun Yuan, Rui Zhang, Hai-Tao Zheng, and Shu-Tao Xia. 2023. MISSRec: Pre-training and Transferring Multi-modal Interest-aware Sequence Representation for Recommendation. In *Proceedings of the 31st ACM International Conference on Multimedia* (, Ottawa ON, Canada,) *(MM '23)*. Association for Computing Machinery, New York, NY, USA, 6548–6557. doi:10.1145/3581783.3611967

[43] Suhang Wang, Yilin Wang, Jiliang Tang, Kai Shu, Suhas Ranganath, and Huan Liu. 2017. What Your Images Reveal: Exploiting Visual Contents for Point-of-Interest Recommendation. In *Proceedings of the 26th International Conference on World Wide Web* (Perth, Australia) *(WWW '17)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 391–400. doi:10.1145/3038912.3052638

[44] Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, Richang Hong, and Tat-Seng Chua. 2019. MMGCN: Multi-modal Graph Convolution Network for Personalized Recommendation of Micro-video. In *Proceedings of the 27th ACM International Conference on Multimedia* (Nice, France) *(MM '19)*. Association for Computing Machinery, New York, NY, USA, 1437–1445. doi:10.1145/3343031.3351034

[45] Liwei Wu, Shuqing Li, Cho-Jui Hsieh, and James Sharpnack. 2020. SSE-PT: Sequential Recommendation Via Personalized Transformer. In *Proceedings of the 14th ACM Conference on Recommender Systems* (Virtual Event, Brazil) *(RecSys '20)*. Association for Computing Machinery, New York, NY, USA, 328–337. doi:10.1145/3383313.3412258

[46] Yuanbo Xu, En Wang, Yongjian Yang, and Yi Chang. 2022. A Unified Collaborative Representation Learning for Neural-Network Based Recommender Systems. *IEEE Transactions on Knowledge and Data Engineering* 34, 11 (2022), 5126–5139. doi:10.1109/TKDE.2021.3054782

[47] Guipeng Xv, Xinyu Li, Ruobing Xie, Chen Lin, Chong Liu, Feng Xia, Zhanhui Kang, and Leyu Lin. 2024. Improving Multi-modal Recommender Systems by Denoising and Aligning Multi-modal Content and User Feedback. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Barcelona, Spain) *(KDD '24)*. Association for Computing Machinery, New York, NY, USA, 3645–3656. doi:10.1145/3637528.3671703

[48] Fajie Yuan, Alexandros Karatzoglou, Ioannis Arapakis, Joemon M. Jose, and Xiangnan He. 2019. A Simple Convolutional Generative Network for Next Item Recommendation. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining* (Melbourne VIC, Australia) *(WSDM '19)*. Association for Computing Machinery, New York, NY, USA, 582–590. doi:10.1145/3289600.3290975

[49] Hanwang Zhang, Yang Yang, Huanbo Luan, Shuicheng Yan, and Tat-Seng Chua. 2014. Start from Scratch: Towards Automatically Identifying, Modeling, and Naming Visual Attributes. *Proceedings of the 22nd ACM international conference on Multimedia* (2014). https://api.semanticscholar.org/CorpusID:10615988

[50] Lingzi Zhang, Xin Zhou, and Zhiqi Shen. 2023. Multimodal pre-training framework for sequential recommendation via contrastive learning. *arXiv preprint arXiv:2303.11879* (2023).

[51] Tingting Zhang, Pengpeng Zhao, Yanchi Liu, Victor S. Sheng, Jiajie Xu, Deqing Wang, Guanfeng Liu, and Xiaofang Zhou. 2019. Feature-level Deeper Self-Attention Network for Sequential Recommendation. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization, 4320–4326. doi:10.24963/ijcai.2019/600

[52] Tingting Zhang, Pengpeng Zhao, Yanchi Liu, Victor S Sheng, Jiajie Xu, Deqing Wang, Guanfeng Liu, Xiaofang Zhou, et al. 2019. Feature-level deeper self-attention network for sequential recommendation.. In *IJCAI*. 4320–4326.

[53] Kun Zhou, Hui Wang, Wayne Xin Zhao, Yutao Zhu, Sirui Wang, Fuzheng Zhang, Zhongyuan Wang, and Ji-Rong Wen. 2020. S3-Rec: Self-Supervised Learning for Sequential Recommendation with Mutual Information Maximization. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management* (Virtual Event, Ireland) *(CIKM '20)*. Association for Computing Machinery, New York, NY, USA, 1893–1902. doi:10.1145/3340531.3411954

[54] Xin Zhou and Zhiqi Shen. 2023. A Tale of Two Graphs: Freezing and Denoising Graph Structures for Multimodal Recommendation. In *Proceedings of the 31st ACM International Conference on Multimedia* (Ottawa ON, Canada) *(MM '23)*. Association for Computing Machinery, New York, NY, USA, 935–943. doi:10.1145/3581783.3611943

[55] Xin Zhou and Zhiqi Shen. 2023. A Tale of Two Graphs: Freezing and Denoising Graph Structures for Multimodal Recommendation. In *Proceedings of the 31st ACM International Conference on Multimedia* (Ottawa ON, Canada) *(MM '23)*. Association for Computing Machinery, New York, NY, USA, 935–943. doi:10.1145/3581783.3611943