Multi-Sequence Attentive User Representation Learning for Side-information Integrated Sequential Recommendation

Xiaolin Lin
College of Computer Science and Software Engineering, Shenzhen University
Shenzhen, China
linxiaolin2021@email.szu.edu.cn

Jinwei Luo
Tencent Music Entertainment
Shenzhen, China
jettluo@tencent.com

Junwei Pan
Tencent
Shenzhen, China
jonaspan@tencent.com

Weike Pan*
Shenzhen University
Shenzhen, China
panweike@szu.edu.cn

Zhong Ming*
Shenzhen University, Shenzhen Technology University, Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ)
Shenzhen, China
mingz@szu.edu.cn

Xun Liu
Tencent
Shenzhen, China
reubenliu@tencent.com

Shudong Huang
Tencent
Shenzhen, China
ericdhuang@tencent.com

Jie Jiang
Tencent
Shenzhen, China
zeus@tencent.com

ABSTRACT

Side-information integrated sequential recommendation incorporates supplementary information to alleviate the issue of data sparsity. The state-of-the-art works mainly leverage some side information to improve the attention calculation to learn user representation more accurately. However, there are still some limitations to be addressed in this topic. Most of them merely learn the user representation at the item level and overlook the association of the item sequence and the side-information sequences when calculating the attentions, which results in the incomprehensive learning of user representation. Some of them learn the user representations at both the item and side-information levels, but they still face the problem of insufficient optimization of multiple user representations. To address these limitations, we propose a novel model, i.e., Multi-Sequence Sequential Recommender (MSSR), which learns the user's multiple representations from diverse sequences. Specifically, we design a multi-sequence integrated attention layer to learn more attentive pairs than the existing works and adaptively fuse these pairs to learn user representation. Moreover, our user representation alignment module constructs the self-supervised signals to optimize the representations. Subsequently, they are further refined by our side information predictor during training. For item prediction, our MSSR extra considers the side information of the candidate item, enabling a comprehensive measurement of the user's preferences. Extensive experiments on four public datasets show that our MSSR outperforms eleven state-of-the-art baselines. Visualization and case study also demonstrate the rationality and interpretability of our MSSR.

CCS CONCEPTS

• **Information systems** → **Recommender systems**.

KEYWORDS

Sequential Recommendation, Side Information, Attention Mechanism, User Representation Learning

ACM Reference Format:
Xiaolin Lin, Jinwei Luo, Junwei Pan, Weike Pan, Zhong Ming, Xun Liu, Shudong Huang, and Jie Jiang. 2024. Multi-Sequence Attentive User Representation Learning for Side-information Integrated Sequential Recommendation. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining (WSDM '24), March 4–8, 2024, Merida, Mexico.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3616855.3635815

*co-corresponding authors

WSDM '24, March 4–8, 2024, Merida, Mexico

ACM ISBN 979-8-4007-0371-3/24/03…$15.00
https://doi.org/10.1145/3616855.3635815

1 INTRODUCTION

User representation learning is a fundamental problem in sequential recommendation (SR). The works on SR leverage the item sequence to learn a vector representation of the user, which is then used to calculate the item prediction score. With the widespread adoption of deep learning, many deep sequential recommenders [1, 10, 12, 26, 27] have been proposed. Notably, the works [17, 21, 22] based

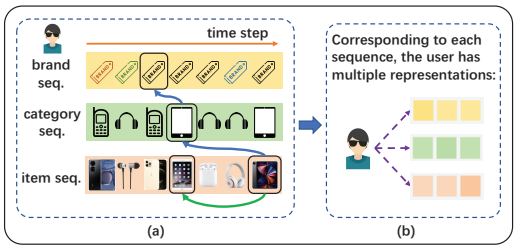

Figure 1: A user's multiple representations corresponding to the brand, category and item sequences. (a) A user's multiple sequences. The green arrow indicates that the interaction of the current item is influenced by the previous item, while the blue arrows indicate that it is also influenced by the relevant category and brand. (b) A user's multiple representations.

on the self-attention mechanism [28] and contrastive learning [6] have exhibited superior performance.

Recently, there have been some works to learn user representation for side-information integrated sequential recommendation (SISR). Intuitively, the additional side information can mitigate the issue of data sparsity and enhance the learning of user representation. The works on SISR can be broadly categorized into two branches. The first branch merely learns user representation at the item level. NOVA-SR [19] and DIF-SR [30] utilize some side information to improve the attention distributions on the item sequences, yielding the more refined user representations at the item level. The second branch extends the learning of user representations at both the item and side-information levels. CAFE [16], FDSA [35] and FDSA-CL [7], utilize independent attention layers to capture fine-grained interests in item sequences and coarse-grained dynamics in side-information sequences. However, there are still two limitations to be addressed in these works:

- **Incomprehensive learning of user representation**. Most of existing works only focus on the attention within a single sequence and overlook the intrinsic association of entities from distinct sequences, such as the attention between a category and a brand. Taking Figure 1(a) as an example, the current iPad is influenced not only by the previous purchase of another but also by the specific category or brand of the product. This oversight hampers a general understanding of the user preferences by neglecting the potential inter-sequence interactions. Moreover, they solely learn the representation at the item level, which falls short in comprehensively representing the user.
- **Insufficient optimization of multiple user representations**. Some works consider the user's interaction sequence as heterogeneous sequences and extract the user's representations at both item and side-information levels. Taking Figure 1(b) as an example, the user has three representations corresponding to his brand, category and item sequences. However, existing works have not adequately addressed the need for more supervision signals when optimizing the multiple representations learned from the same user.

To address the above limitations, we propose a novel model called Multi-Sequence Sequential Recommender (MSSR) to model a user's item sequence and the corresponding side-information sequences. **For the first limitation**, we design a multi-sequence integrated attention (MSIA) layer to exploit the interactions within and between sequences to comprehensively represent a user. In MSIA layers, we calculate the intra-sequence attentions and the inter-sequence attentions to derive multiple attention matrices. We then design adaptive attention fusion to assign the weights for different matrices and fuse them to obtain the final attention matrices for learning the same user's representations at both the item and side-information levels. **For the second limitation**, we design a user representation alignment module to optimize a user's multiple representations. As the representations capture the same user's preferences from various perspectives, we assume that they tend to have relatively high similarity in the representation space. Hence, we employ the contrastive learning training scheme to moderately enhance their similarity and further optimize them. Specifically, we construct the self-supervised signals and introduce contrastive losses to align the user representation at the item level with the representation at each side-information level. Moreover, we design a side information predictor to refine the user representations. After addressing the limitations, for item prediction, we leverage the commonly available side information of the candidate item to consider the user's preference at both the item and side-information levels.

In conclusion, we summarize our main contributions as follows:

- We propose a novel model called Multi-Sequence Sequential Recommender (MSSR), which effectively learns the user's multiple representations from mutually interconnected sequences.
- We design a multi-sequence integrated attention layer to adaptively leverage both intra- and inter-sequence interactions. Moreover, we propose a user representation alignment module for optimizing the representations of the same user and a side information predictor to further refine them. At last but not least, we leverage the side information of the candidate item to make a comprehensive user preference measurement.
- We conduct extensive experiments on four public datasets, demonstrating that our MSSR outperforms both the SR and SISR baselines. Additionally, visualization and case study provide insights into the rationality and interpretability of our MSSR.

2 PROBLEM DEFINITION

We have a set of users $\mathcal{U}$ and a set of items $\mathcal{V}$. Side information comprises item-related information, such as category, as well as behavior-related information, such as position. Suppose there are K types of side information, and the set is denoted as $\mathcal{A} = \{a_1, \ldots, a_k, \ldots, a_K\}$. We use C_k to represent the set of all the concrete values about the side information a_k.

For a user $u \in \mathcal{U}$, his or her side-information integrated interaction sequence is represented as $\mathcal{S}^u = \{\boldsymbol{s}^u[1], \ldots, \boldsymbol{s}^u[i], \ldots, \boldsymbol{s}^u[n]\}$, where n is the sequence length. Each interaction can be denoted as $\boldsymbol{s}^u[i] = (\boldsymbol{v}^u[i], \boldsymbol{c}^u_{a_1}[i], \ldots, \boldsymbol{c}^u_{a_k}[i], \ldots, \boldsymbol{c}^u_{a_K}[i])$, where $\boldsymbol{v}^u[i] \in \mathcal{V}$ denotes the item ID at time step i, and $\boldsymbol{c}^u_{a_k}[i] \in C_k$ denotes the specific value of the side information a_k. Then $\mathcal{S}^u$ can be conceptualized as the integration of the multiple sequences: $\mathcal{S}^u = (\boldsymbol{v}^u, \boldsymbol{c}^u_{a_1}, \ldots, \boldsymbol{c}^u_{a_K})$. Given the multiple sequences of user u, the task of side-information integrated sequential recommendation is to predict the next item $\boldsymbol{v}^u[n+1]$ for user u, which can be formulated as:

$$\boldsymbol{v}^u[n+1] = \arg\max_{v_i \in \mathcal{V}} P\left(\boldsymbol{v}^u[n+1] = v_i \,\middle|\, \boldsymbol{v}^u, \boldsymbol{c}^u_{a_1}, \ldots, \boldsymbol{c}^u_{a_K}\right). \quad (1)$$

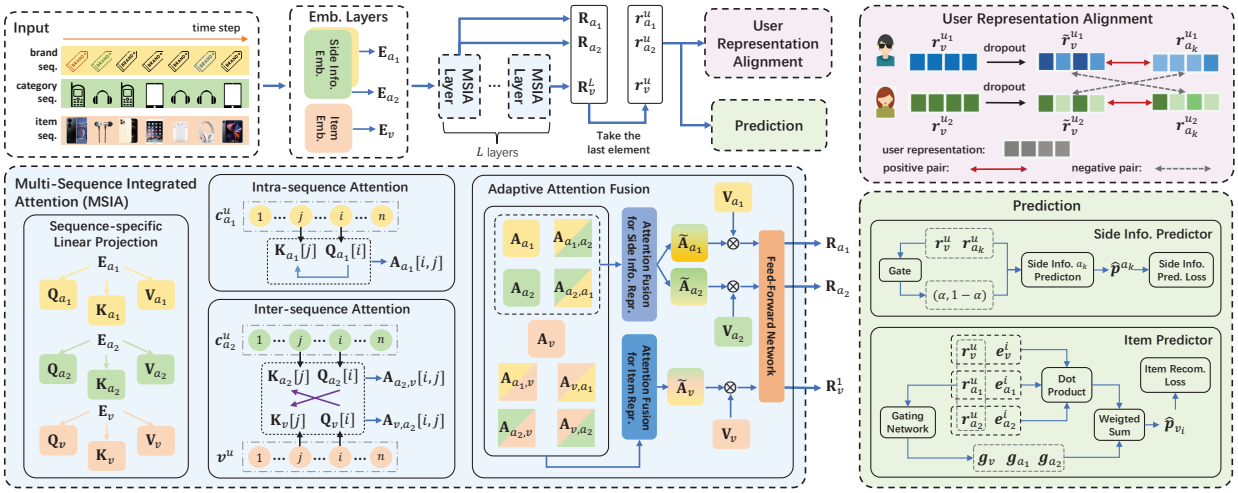

Figure 2: The overview of our MSSR. Our MSSR takes multiple sequences as input and obtains the embedding matrices via the embedding layers. These matrices are fed into the multi-sequence integrated attention layers to learn the user's multiple representations, which are further optimized by the user representation alignment module. Finally, the prediction module utilizes the learned representations to perform side information prediction (during training) and item prediction.

3 METHODOLOGY

As illustrated in Figure 2, our MSSR has three key modules: (1) multi-sequence integrated attention (MSIA) layers; (2) user representation alignment (URA); and (3) prediction. First, we embed multiple sequences into corresponding embedding matrices $\mathbf{E}_v \in \mathbb{R}^{n\times d}$, $\mathbf{E}_{a_1}, \ldots, \mathbf{E}_{a_K} \in \mathbb{R}^{n\times d'}$, where d and d' are the latent dimensionality of item and side information, respectively. Next, we introduce the three key modules.

3.1 Multi-Sequence Integrated Attention Layers

We input embedding matrices into the MSIA layers to attain the representation matrices. Among them, side-information representation matrices are learned at the first MSIA layer, which is described in detail at Sec. 3.1.4. Note that we follow [19, 30] and input the original side-information embedding matrices into each MSIA layer. The overall process can be formulated as follows,

$$\mathbf{R}_v^1, \mathbf{R}_{a_1}, \ldots, \mathbf{R}_{a_K} = \text{MSIA}^{(1)}\left(\mathbf{E}_v,\ \mathbf{E}_{a_1}, \ldots, \mathbf{E}_{a_K}\right), \tag{2}$$

$$\mathbf{R}_v^\ell = \text{MSIA}^{(\ell)}\left(\mathbf{R}_v^{\ell-1},\ \mathbf{E}_{a_1}, \ldots, \mathbf{E}_{a_K}\right),\ \ell \in \{2, \ldots, L\}. \tag{3}$$

By L MSIA layers, we attain the representation matrices of each sequence, i.e., $\mathbf{R}_v^L \in \mathbb{R}^{n\times d}, \mathbf{R}_{a_1}, \ldots, \mathbf{R}_{a_K} \in \mathbb{R}^{n\times d'}$. Following [22], we take the last element of $\mathbf{R}_v^L$ as the user representation at the item level, i.e., $\boldsymbol{r}_v^u = \mathbf{R}_v^L[n] \in \mathbb{R}^d$. Similarly, we have the user representations at various side-information levels, i.e., $\boldsymbol{r}_{a_1}^u, \ldots, \boldsymbol{r}_{a_K}^u \in \mathbb{R}^{d'}$. Next, we elaborate on the MSIA layer, using the learning process of representation matrices at the first layer.

3.1.1 ***Sequence-specific Projection.*** To decouple the later attention calculation [30], we perform sequence-specific projections on each embedding matrix to get query, key and value matrices of M different attention heads. We take the query matrix of each sequence at the m-th attention head as an example,

$$\mathbf{Q}_v^m = f_{Q_v^m}(\mathbf{E}_v),\ \mathbf{Q}_{a_k}^m = f_{Q_{a_k}^m}(\mathbf{E}_{a_k}), \tag{4}$$

where $\mathbf{Q}_v^m \in \mathbb{R}^{n\times \frac{d}{M}}$, $\mathbf{Q}_{a_k}^m \in \mathbb{R}^{n\times \frac{d'}{M}}$, $a_k \in \mathcal{A}$ is a type of side information, and both $f_{Q_v^m}(\cdot)$ and $f_{Q_{a_k}^m}(\cdot)$ denote a linear layer that projects the embedding matrix of a specific sequence. Similarly, we attain $\mathbf{K}_v^m$, $\mathbf{V}_v^m$ as the key and value matrices of the item sequence, and $|\mathcal{A}|$ pairs of key and value matrices $(\mathbf{K}_{a_k}^m, \mathbf{V}_{a_k}^m)$ of the side-information sequences.

3.1.2 ***Intra-sequence Attention.*** We utilize dot product to calculate attentions of entity pairs in a single sequence. Given two items at the positions i and j in an item sequence, we can calculate their attention as follows,

$$\mathbf{A}_v^m[i,j] = \mathbf{Q}_v^m[i]\left(\mathbf{K}_v^m[j]\right)^\top, \tag{5}$$

where $\mathbf{A}_v^m \in \mathbb{R}^{n\times n}$ denotes the attention matrix of an item sequence at the m-th attention head. Similarly, we attain $\mathbf{A}_{a_1}^m, \ldots, \mathbf{A}_{a_k}^m$ as the attention matrices of the other sequences. We attain $|\mathcal{A}|+1$ matrices by calculating the intra-sequence attention on all the sequences.

3.1.3 ***Inter-sequence Attention.*** To capture the association of two entities from distinct sequences, we propose the inter-sequence attention. For each user, we have an item sequence and the corresponding $|\mathcal{A}|$ side-information sequences. Without loss of generality, we utilize two of these sequences and denote them with the notations x and y, respectively, where $x, y \in \mathcal{A}\cup\{v\}$ and $x \neq y$.

As the two entity representations from distinct sequences are derived from different embedding spaces, inspired by [13], we introduce a parameter matrix $\mathbf{W}$ and construct a bi-linear layer to handle this heterogeneity. Given the entity x_i from sequence x and the entity y_j from sequence y, their attention is calculated as follows,

$$\mathbf{A}_{x,y}^m[i,j] = \mathbf{Q}_x^m[i]\mathbf{W}\left(\mathbf{K}_y^m[j]\right)^\top, \tag{6}$$

where $\mathbf{A}_{x,y}^m \in \mathbb{R}^{n\times n}$ denotes the attention matrix of the sequence pair (x, y) at the m-th attention head. Note that we attain $|\mathcal{A}|(|\mathcal{A}|+1)$ matrices by pairwise combination of different sequences, because the bi-linear layer is asymmetric.

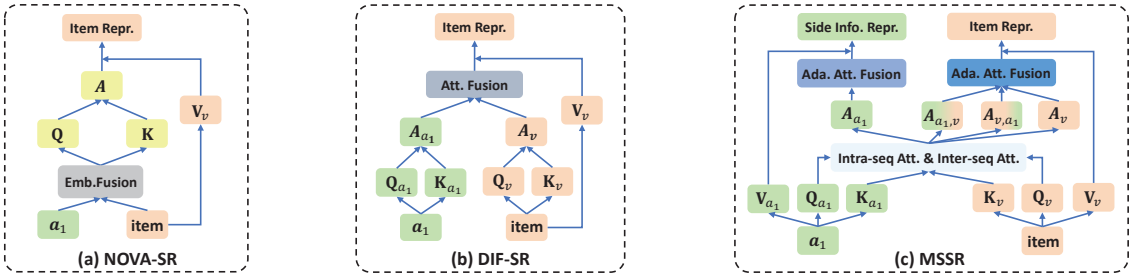

Figure 3: The comparison of the representation learning processes of NOVA-SR, DIF-SR and our MSSR. Note that we use only one kind of side information for a clear illustration.

3.1.4 ***Adaptive Attention Fusion.*** We can attain $(|\mathcal{A}|+1)^2$ attention matrices after calculating the intra-sequence and inter-sequence attentions on all the sequences. Then we conduct the fusion operation on these attention matrices, and the matrix set for learning item representation matrix is formulated as $\mathfrak{A}^v = \left\{\mathbf{A}_z^m \,|z \in \mathcal{A}\cup\{v\}\right\} \cup \left\{\mathbf{A}_{x,y}^m \,|x,y \in \mathcal{A}\cup\{v\}, x \neq y\right\}$. Inspired by [37], we introduce an extra learnable parameter for each matrix to adaptively assign the corresponding fusion weight to get the fused attention matrix as follows,

$$\tilde{\mathbf{A}}_v^m = \mathcal{F}\left(\boldsymbol{w}[1]\mathfrak{A}_1^v, \ldots, \boldsymbol{w}[i]\mathfrak{A}_i^v, \ldots, \boldsymbol{w}[B]\mathfrak{A}_B^v\right), \tag{7}$$

where $B = (|\mathcal{A}|+1)^2$, $\mathfrak{A}_i^v \in \mathfrak{A}^v$, $\tilde{\mathbf{A}}_v^m \in \mathbb{R}^{n\times n}$, $\boldsymbol{w} \in \mathbb{R}^B$ is a vector containing learnable weights that sum to one, and the fusion function $\mathcal{F}$ is explored in [19], including addition, concatenation and gating.

We also learn the side-information representation matrices. Different from Eq.(7) that takes all the matrices, we merely use the matrices that are calculated by side-information query and key matrices. Because the side information, such as the category, is relatively coarser-grained compared with the item ID, the attention fusion may not require the fine-grained information from the items. The matrix set for learning the side-information representation matrix $\mathbf{R}_{a_k}$ is formulated as $\mathfrak{A}^{a_k} = \left\{\mathbf{A}_z^m | z \in \mathcal{A}\right\} \cup \left\{\mathbf{A}_{x,y}^m | x,y \in \mathcal{A}, x \neq y\right\}$. Similarly, the fused attention matrix for learning side-information representation matrix is calculated as follows,

$$\tilde{\mathbf{A}}_{a_k}^m = \mathcal{F}\left(\boldsymbol{w}'[1]\mathfrak{A}_1^{a_k}, \ldots, \boldsymbol{w}'[i]\mathfrak{A}_i^{a_k}, \ldots, \boldsymbol{w}'[B']\mathfrak{A}_{B'}^{a_k}\right), \tag{8}$$

where $B' = |\mathcal{A}|^2$, $\mathfrak{A}_i^{a_k} \in \mathfrak{A}^{a_k}$, $\tilde{\mathbf{A}}_{a_k}^m \in \mathbb{R}^{n\times n}$, $\boldsymbol{w}' \in \mathbb{R}^{B'}$ is a vector with learnable weights, and the sum of which also equals one. We calculate the side-information representation matrices only at the first layer as Eq.(2) to save computation, because the embedding matrices of side information inputted at each layer are the same.

We attain the output of the m-th attention head using the fused attention matrix and the item value matrix, then concatenate the outputs of all the attention heads as the input for a feed-forward network to attain the item representation matrix as follows,

$$\mathbf{H}_v^m = \text{softmax}\left(\frac{\tilde{\mathbf{A}}_v^m \otimes \Delta}{\sqrt{d}}\right)\mathbf{V}_v^m, \tag{9}$$

$$\mathbf{R}_v^1 = \text{FFN}\left(\text{concat}\left(\mathbf{H}_v^1, \ldots, \mathbf{H}_v^m, \ldots, \mathbf{H}_v^M\right)\mathbf{W}_v\right), \tag{10}$$

where $\mathbf{W}_v \in \mathbb{R}^{d\times d}$ is a parameter matrix, and Δ is the causality mask for preventing the future information in the sequence [12].

Similarly, we can attain side-information representation matrices $\mathbf{R}_{a_1}, \ldots, \mathbf{R}_{a_K} \in \mathbb{R}^{n\times d'}$.

We show the representation learning processes of NOVA-SR [19], DIF-SR [30] and our MSSR in Figure 3 to illustrate their differences and relations. We also keep the value matrices non-invasive, following NOVA-SR and DIF-SR, but we further propose the inter-sequence attention and adaptive attention fusion to learn the representations at both the item and side-information levels.

3.2 User Representation Alignment

We construct the self-supervised signals and adopt the contrastive learning training scheme to effectively optimize a user's multiple representations. Inspired by [5], we employ a dropout layer [25] as a model-level augmentation to construct the positive sample pairs in the contrastive loss.

Unlike the prior work [22] that re-encodes the sequence and derives a new representation, we directly utilize a dropout layer to attain the augmented item-level representations to reduce the computational complexity, i.e., $\tilde{\boldsymbol{r}}_v^u = \text{Dropout}(\boldsymbol{r}_v^u)$. Then, for each representation at the side-information level $\boldsymbol{r}_{a_k}^u$, we treat $(\tilde{\boldsymbol{r}}_v^u, \boldsymbol{r}_{a_k}^u)$ as a positive pair and align the user's two representations via a contrastive loss. Finally, we finish $|\mathcal{A}|$ pairs of alignment to achieve the overall alignment of the user's representations.

Considering a user mini-batch $\mathcal{B} = \left\{u_1, \ldots, u_{|\mathcal{B}|}\right\}$ and a type of side information a_k, we have $2|\mathcal{B}|$ representations $[\tilde{\boldsymbol{r}}_v^{u_1}, \boldsymbol{r}_{a_k}^{u_1}, \ldots, \tilde{\boldsymbol{r}}_v^{u_{|\mathcal{B}|}}, \boldsymbol{r}_{a_k}^{u_{|\mathcal{B}|}}]$. For a positive pair $(\tilde{\boldsymbol{r}}_v^{u_1}, \boldsymbol{r}_{a_k}^{u_1})$, its corresponding negative pair can be denoted as $(\tilde{\boldsymbol{r}}_v^{u_1}, \boldsymbol{r}^-)$, where $\boldsymbol{r}^- \in \mathcal{R}_{u_1}^- = \left\{\boldsymbol{r}_{a_k}^{u_2}, \ldots, \boldsymbol{r}_{a_k}^{u_{|\mathcal{B}|}}\right\}$. Therefore, to align the representation pair $(\tilde{\boldsymbol{r}}_v^u, \boldsymbol{r}_{a_k}^u)$, the contrastive loss is formulated as follows,

$$\mathcal{L}_{cl}(\tilde{\boldsymbol{r}}_v^u, \boldsymbol{r}_{a_k}^u) = -\log\frac{f(\tilde{\boldsymbol{r}}_v^u, \boldsymbol{r}_{a_k}^u)}{f(\tilde{\boldsymbol{r}}_v^u, \boldsymbol{r}_{a_k}^u) + \sum_{\boldsymbol{r}^-\in\mathcal{R}_u^-} f(\tilde{\boldsymbol{r}}_v^u, \boldsymbol{r}^-)}, \tag{11}$$

$$f(\tilde{\boldsymbol{r}}_v^u, \boldsymbol{r}_{a_k}^u) = \exp\left(\tilde{\boldsymbol{r}}_v^u \mathbf{W}'(\boldsymbol{r}_{a_k}^u)^\top\right), \tag{12}$$

$$f(\tilde{\boldsymbol{r}}_v^u, \boldsymbol{r}^-) = \exp\left(\tilde{\boldsymbol{r}}_v^u \mathbf{W}'(\boldsymbol{r}^-)^\top\right), \tag{13}$$

where $\mathbf{W}' \in \mathbb{R}^{d\times d'}$ is the parameter matrix for dimension transformation. Based on Eq.(11), the overall contrastive loss that considers all the pairs of representation alignment is defined as follows,

$$\mathcal{L}_{cl} = \sum_{a_k\in\mathcal{A}} \mathcal{L}_{cl}(\tilde{\boldsymbol{r}}_v^u, \boldsymbol{r}_{a_k}^u). \tag{14}$$

By minimizing $\mathcal{L}_{cl}$, we realize the alignment of the user's multiple representations, which utilizes the self-supervised signals to

effectively optimize the user's multiple representations learned from MSIA layers.

3.3 Prediction

3.3.1 ***Side Information Predictor.*** Inspired by [30], we design the side information predictor to further refine the user's multiple representations during training. We use the learnable gated weights to quantify the relative importance of the user representations at the item and side-information levels. And the gated weight is calculated as $\alpha = \sigma\left(\left[\boldsymbol{r}_v^u; \boldsymbol{r}_{a_k}^u\right]\mathbf{W}_g\right)$, where $\mathbf{W}_g \in \mathbb{R}^{(d+d')\times 1}$. For the side information a_k, the prediction is formulated as follows,

$$\hat{\boldsymbol{p}}^{a_k} = \sigma\left(\alpha \cdot \boldsymbol{r}_v^u \mathbf{W}'_v + (1-\alpha)\cdot \boldsymbol{r}_{a_k}^u \mathbf{W}'_{a_k}\right), \tag{15}$$

where $\mathbf{W}'_v \in \mathbb{R}^{d\times|C_k|}$ and $\mathbf{W}'_{a_k} \in \mathbb{R}^{d'\times|C_k|}$ are learnable parameters, and $\hat{\boldsymbol{p}}^{a_k}$ is a $|C_k|$-dimensional probability. To support multi-labeled side information, we follow [30] and adopt the binary cross-entropy loss for the task of side information prediction as follows,

$$\mathcal{L}_{a_k} = -\sum_{j=1}^{|C_k|} \boldsymbol{p}_j^{a_k}\log(\hat{\boldsymbol{p}}_j^{a_k}) + (1-\boldsymbol{p}_j^{a_k})\log(1-\hat{\boldsymbol{p}}_j^{a_k}), \tag{16}$$

where $\boldsymbol{p}^{a_k}$ is a $|C_k|$-dimensional multi-hot vector of the ground truth. Minimizing $\mathcal{L}_{a_k}$ and training to predict the side information compel both the item and side-information level representations to incorporate useful information from the supervised signal of the ground truth, which helps to further refine them.

3.3.2 ***Item Predictor.*** Typical sequential recommenders only consider item ID information of the candidate items and measure the user's preference at the item level. In addition to ID information, we also use the commonly accessible side information of the candidate items to comprehensively measure the user's preference at both the item and side-information levels.

We calculate the dot products between the user's different representations and the corresponding types of embeddings w.r.t the candidate item, and then leverage their learnable weighted summation as the final item prediction score. A gating network is exploited to learn the weight vector as follows,

$$\boldsymbol{g} = \text{softmax}\left(\text{MLP}(\left[\boldsymbol{r}_v^u; \boldsymbol{r}_{a_1}^u; \ldots; \boldsymbol{r}_{a_K}^u\right])\right), \tag{17}$$

where $\boldsymbol{g} = \left[\boldsymbol{g}_v, \boldsymbol{g}_{a_1}, \ldots, \boldsymbol{g}_{a_K}\right] \in \mathbb{R}^{1+|\mathcal{A}|}$ contains the weights that sum to one. Considering a user u and a candidate item v_i, the side-information fused item prediction score is defined as follows,

$$\hat{\boldsymbol{p}}_{v_i} = \boldsymbol{g}_v \cdot \boldsymbol{r}_v^u (\boldsymbol{e}_v^i)^\top + \sum_{a_k\in\mathcal{A}} \boldsymbol{g}_{a_k} \cdot \boldsymbol{r}_{a_k}^u (\boldsymbol{e}_{a_k}^i)^\top, \tag{18}$$

where $\boldsymbol{e}_v^i$ is the item embedding of v_i, and $\boldsymbol{e}_{a_k}^i$ is the corresponding embedding of the side information a_k, such as the category embedding. By calculating the prediction scores of all the candidate items, we attain the probability distribution $\hat{\boldsymbol{p}} = \text{softmax}(\hat{\boldsymbol{p}}_{v_1}, \ldots, \hat{\boldsymbol{p}}_{v_{|\mathcal{V}|}}) \in \mathbb{R}^{|\mathcal{V}|}$. Then we use the cross-entropy loss to minimize the distance between $\hat{\boldsymbol{p}}$ and the ground truth $\boldsymbol{p} \in \mathbb{R}^{|\mathcal{V}|}$ (a one-hot vector) for the task of item recommendation as follows,

$$\mathcal{L}_{rec} = -\sum_{i=1}^{|\mathcal{V}|} \boldsymbol{p}[i]\ \log(\hat{\boldsymbol{p}}[i]). \tag{19}$$

Table 1: Statistics of the four processed datasets.

Dataset	Yelp	Toys	Beauty	Sports
# Users	30,499	19,412	22,363	35,598
# Items	20,068	11,924	12,101	18,357
# Avg. Interactions / User	10.4	8.6	8.9	8.3
# Avg. Interactions / Item	15.8	14.1	16.4	16.1
# Interactions	317,182	167,597	198,502	296,337
Sparsity	99.95%	99.93%	99.93%	99.95%

3.4 Model Training and Inference

We adopt the multi-task learning strategy to train our MSSR by minimizing the summation of the item recommendation loss, the contrastive loss and the side-information prediction loss,

$$\mathcal{L} = \mathcal{L}_{rec} + \lambda_1\mathcal{L}_{cl} + \lambda_2 \sum_{a_k\in\mathcal{A}} \mathcal{L}_{a_k}, \tag{20}$$

where λ_1 and λ_2 are tradeoff parameters.

We utilize our MSSR to calculate the prediction scores of all the candidate items via Eq.(18) and recommend the item with the highest score as the next item for user u:

$$\boldsymbol{v}^u[n+1] = \arg\max_{v_i\in\mathcal{V}} \hat{\boldsymbol{p}}_{v_i}. \tag{21}$$

4 EXPERIMENTS

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

- **RQ1**: How does our MSSR perform compared with state-of-the-art baselines?
- **RQ2**: What is the effect of different modules in our MSSR?
- **RQ3**: What is the impact of some other options for the design of our attention layer and item predictor in our MSSR?
- **RQ4**: How is the rationality of the attention fusion weight distribution in MSIA?
- **RQ5**: How is the model interpretability of our MSSR?

4.1 Experimental Setup

4.1.1 ***Datasets and Evaluation Metrics.*** We follow [30] and conduct experiments on four public and real-world datasets. One is a widely used dataset for the business recommendation, **Yelp**[1]. And the other three are **Toys, Beauty, Sports**, which are constructed from the Amazon review datasets[2]. We follow [30] and utilize the categories and the positions as the side information for all the datasets. As for dataset preprocessing, we follow [30] to remove the users and items that occur less than five times, then we adopt the leave-one-out principle to attain train data, validation and test data for all the datasets. The statistics of the processed datasets are summarized in Table 1.

For the performance evaluation, we follow [30] and use Recall@k and NDCG@k as the metrics, where $k \in \{10, 20\}$. We follow the suggestions in [3, 15] and evaluate the model performance in a full ranking setting rather than only on some sampled items for a more reasonable comparison.

[1]https://www.yelp.com/dataset
[2]http://jmcauley.ucsd.edu/data/amazon/

Table 2: The performance comparison of our MSSR against two groups of baselines on four datasets. Note that the best result of each row is marked in bold, and the second best one is underlined.

Datasets	Metrics	SR baselines				SISR baselines							MSSR
		GRU4Rec	SASRec	BERT4Rec	DuoRec	GRU4Rec$_F$	SASRec$_F$	S^3-Rec	FDSA-CL	NOVA-SR	CAFE	DIF-SR	
Yelp	Recall@10	0.0382	0.0612	0.0521	0.0641	0.0341	0.0417	0.0616	0.0625	0.0663	0.0633	0.0691	**0.0715**
	Recall@20	0.0621	0.0922	0.0756	0.0951	0.0556	0.0675	0.0934	0.0921	0.0972	0.0954	0.1004	**0.1063**
	NDCG@10	0.0192	0.0393	0.0328	0.0378	0.0175	0.0216	0.0396	0.0377	0.0406	0.0376	0.0413	**0.0427**
	NDCG@20	0.0252	0.0465	0.0387	0.0450	0.0229	0.0281	0.0462	0.0451	0.0483	0.0453	0.0493	**0.0513**
Toys	Recall@10	0.0481	0.0879	0.0504	0.0947	0.0494	0.0738	0.0889	0.0851	0.0955	0.0809	0.0989	**0.1033**
	Recall@20	0.0721	0.1202	0.0778	0.1297	0.0723	0.1058	0.1256	0.1169	0.1318	0.1055	0.1361	**0.1414**
	NDCG@10	0.0267	0.0431	0.0323	0.0487	0.0274	0.0411	0.0433	0.0417	0.0469	0.0476	0.0493	**0.0518**
	NDCG@20	0.0328	0.0499	0.0392	0.0575	0.0322	0.0491	0.0492	0.0507	0.0569	0.0543	0.0587	**0.0614**
Beauty	Recall@10	0.0542	0.0848	0.0608	0.0865	0.0561	0.0727	0.0865	0.0824	0.0869	0.0840	0.0891	**0.0915**
	Recall@20	0.0845	0.1189	0.0881	0.1251	0.0885	0.1011	0.1219	0.1115	0.1243	0.1159	0.1292	**0.1318**
	NDCG@10	0.0232	0.0409	0.0331	0.0441	0.0255	0.0427	0.0432	0.0424	0.0432	0.0437	0.0443	**0.0458**
	NDCG@20	0.0309	0.0498	0.0401	0.0539	0.0339	0.0506	0.0483	0.0497	0.0524	0.0514	0.0544	**0.0556**
Sports	Recall@10	0.0292	0.0471	0.0395	0.0483	0.0317	0.0458	0.0492	0.0447	0.0522	0.0429	0.0548	**0.0559**
	Recall@20	0.0466	0.0699	0.0586	0.0712	0.0479	0.0682	0.0694	0.0653	0.0751	0.0611	0.0798	**0.0820**
	NDCG@10	0.0152	0.0216	0.0194	0.0247	0.0165	0.0245	0.0223	0.0222	0.0246	0.0254	0.0255	**0.0262**
	NDCG@20	0.0195	0.0274	0.0232	0.0304	0.0218	0.0294	0.0287	0.0284	0.0302	0.0299	0.0318	**0.0328**

4.1.2 ***Baselines and Implementation Details.*** In order to show the superiority of our MSSR, we choose two groups of baselines to compare the recommendation performance, including sequential recommendation (SR) baselines and side-information integrated sequential recommendation (SISR) baselines.

SR baselines: (1) **GRU4Rec** [10] utilizes GRU layers to learn the pattern of the item sequences. (2) **SASRec** [12] uses the self-attention mechanism to learn user representation from the item sequences. (3) **BERT4Rec** [26] adopts a bidirectional self-attention network and conducts the Cloze task to predict the masked items based on the sequence context during the training process. (4) **DuoRec** [22] studies the problem of representation degeneration and proposes solutions with contrastive learning techniques.

SISR baselines: (1) **GRU4Rec$_F$** enhances the original GRU4Rec by taking the summation of the item embeddings and the side-information embeddings as the input of the model. (2) **SASRec$_F$** extends SASRec and concatenates the item and the side-information embeddings to be fed into the model. (3) **S^3-Rec** [41] devises four pre-training tasks with contrastive learning based on maximizing mutual information. (4) **FDSA-CL** [7] is an enhanced version of FDSA [35] that utilizes contrastive learning to exploit the beneficial interaction of representations from the side-information level and the item level. (5) **NOVA-SR** [19] devises a non-invasive attention mechanism to learn user representation. (6) **CAFE** [16] fuses the user representations at both the item and side-information levels by addition to predict the next item. (7) **DIF-SR** [30] decouples the attention calculation to attain the fused attention for learning user representation.

We implement and evaluate all the baselines and our MSSR based on RecBole [36] for a fair performance comparison. For all the baselines and our MSSR, we train them with the same optimizer Adam [14] and a learning rate of 0.0001. The latent dimensionality d is set to 256 for all the methods. We tune all the hyper-parameters on the validation data, following the suggestions in the original papers, and report the results on the test data. For the two tradeoff parameters in the Eq.(20), we choose λ_1 from $\{0.1, 0.2, 0.3, 0.4, 0.5\}$, and λ_2 from $\{2, 4, 6, 8, 10\}$. For the latent dimensionality d', we choose its value from $\{32, 64, 128, 256\}$. For the other hyper-parameters, we follow [30] and search the attention layer number L from $\{2, 3, 4\}$, and the attention head number M from $\{2, 4, 8\}$. All the model are trained using Tesla V100 PCIe GPU with 32 GB memory. Our source code is available at https://github.com/xiaolLIN/MSSR.

4.2 Overall Performance Comparison (RQ1)

We report the experimental results of eleven baselines and our MSSR in Table 2.

Among the SR baselines, we have the following observations: (1) Both SASRec and BERT4Rec outperform GRU4Rec on all the metrics across all the datasets, which demonstrates the superiority of the self-attention mechanism. (2) BERT4Rec does not beat SASRec, which is also observed in the previous works [3, 30, 41]. We attribute this phenomenon to the fact that the Cloze task of training BERT4Rec might not adapt well to the next-item prediction task in the full-ranking setting [3, 15]. (3) Compared with other SR methods, DuoRec achieves the best results in most cases, except that SASRec performs better in some cases on Yelp, which shows the effectiveness of contrastive learning techniques.

Among the methods for SISR, we have the following observations: (1) Simple side information fusion, such as addition, may not always be effective, which can be found by comparing the results of GRU4Rec$_F$ with its original version without side information. (2) The results of SASRec$_F$ are worse than SASRec on most metrics across the datasets. We attribute this phenomenon to the invasive fusion utilized in SASRec$_F$, which has the drawback of the compound embedding space [19]. However, the well-designed fusion strategy proposed in the recent works, such as S^3-Rec, NOVA-SR, DIF-SR and our MSSR, can help to improve the performance. (3) Our MSSR outperforms DIF-SR. It indicates that our MSIA layers capture more attentive pairs across distinct sequences and adaptively assign the fusion weight for each of them, which contributes to a comprehensive learning of user representations. (4) Our MSSR surpasses FDSA-CL and CAFE, which demonstrates the effectiveness our user representation alignment module for optimizing the

user's multiple representations. (5) Our MSSR outperforms all the baselines across all the metrics on four datasets, which shows the superiority of our MSSR.

We consider more side information to further study the superiority of our MSSR. Specifically, we add "city" as our side information for Yelp, and "brand" for Toys, Beauty and Sports. We compare our MSSR with three representative SISR baselines and report the results in Table 3. It is in line with our intuition that considering more side information brings performance improvement in most cases, which can be seen by comparing the results of the corresponding methods in Tables 2 and 3. Meanwhile, our MSSR consistently achieves the best results, which further verifies the effectiveness of our MSSR in dealing with more side information.

Table 3: The performance comparison of our MSSR and three representative baselines while using more side information.

Datasets	Metrics	Baselines			MSSR
		FDSA-CL	NOVA-SR	DIF-SR	
Yelp	Recall@10	0.0634	0.0679	0.0699	**0.0716**
	Recall@20	0.0936	0.0977	0.1031	**0.1069**
	NDCG@10	0.0379	0.0413	0.0423	**0.0431**
	NDCG@20	0.0461	0.0494	0.0505	**0.0518**
Toys	Recall@10	0.0856	0.0967	0.0997	**0.1035**
	Recall@20	0.1177	0.1322	0.1377	**0.1419**
	NDCG@10	0.0424	0.0486	0.0504	**0.0521**
	NDCG@20	0.0516	0.0575	0.0603	**0.0614**
Beauty	Recall@10	0.0829	0.0888	0.0898	**0.0916**
	Recall@20	0.1114	0.1250	0.1289	**0.1322**
	NDCG@10	0.0431	0.0442	0.0452	**0.0463**
	NDCG@20	0.0493	0.0535	0.0542	**0.0559**
Sports	Recall@10	0.0467	0.0531	0.0553	**0.0564**
	Recall@20	0.0685	0.0762	0.0795	**0.0819**
	NDCG@10	0.0233	0.0249	0.0253	**0.0261**
	NDCG@20	0.0295	0.0301	0.0319	**0.0327**

4.3 Ablation Study of MSSR (RQ2)

To analyze the contribution of different modules in our MSSR, we conduct a detailed ablation study and report the results in Table 4. Specifically, we study the contributions of the multi-sequence integrated attention (MSIA) layers, the user representation alignment (URA) module and the side information predictor (SIP). For MSIA, we further study on its inter-sequence attention (IA) and adaptive attention fusion (AAF). It is worth noting that MSSR(w/o MSIA&URA&SIP) is equivalent to SASRec$_F$. We have the following observations:

- MSIA contributes the most to the performance of our MSSR, which can be verified by the prominent improvement of MSSR(w/o URA&SIP) over MSSR(w/o MSIA&URA&SIP). It shows the effectiveness of the MSIA layers to learn the user's representations.
- MSSR(w/o IA&AAF&URA&SIP) retains the sequence-specific projections and the intra-sequence attention. It outperforms the variant from which the whole MSIA is removed, which demonstrates the importance of decoupled attention calculation. Moreover, the performance is further improved once we add AAF to the model. It shows the validness of adaptively assigning fusion weights in our MSIA.
- URA can further improve the performance, which can be seen from the fact that MSSR(w/o SIP) outperforms MSSR(w/o URA&SIP) on every dataset. It shows the effectiveness of constructing the positive and negative pair samples in the contrastive loss with the self-supervised signals.
- Comparing MSSR(w/o URA) with MSSR(w/o URA&SIP), the performance gap shows that our SIP can further refine the user's representations to improve the prediction performance.
- Our whole MSSR performs the best compared with all its variants, which demonstrates the complementary effect of all the designed modules in our MSSR.

Table 4: Ablation study of our MSSR (Recall@20).

Variants	Yelp	Toys	Beauty	Sports
w/o MSIA&URA&SIP	0.0675	0.1058	0.1011	0.0682
w/o IA&AAF&URA&SIP	0.0967	0.1301	0.1240	0.0757
w/o AAF&URA&SIP	0.0979	0.1313	0.1255	0.0764
w/o URA&SIP	0.0981	0.1321	0.1263	0.0770
w/o SIP	0.1041	0.1374	0.1280	0.0806
w/o URA	0.1005	0.1376	0.1298	0.0807
MSSR	**0.1063**	**0.1414**	**0.1318**	**0.0820**

4.4 Exploratory Study (RQ3)

In this section, we conduct the exploratory study to some other options for the design of our attention layer and item predictor.

4.4.1 ***Attention Layer.*** We compare our MSIA layer with the self-attention (SA) layer of SASRec$_F$, the non-invasive attention (NOVA) layer of NOVA-SR and the attention layer of DIF-SR (DIF). To directly and properly show the superiority of our MSIA, for DIF-SR, we remove the auxiliary loss. And for our MSSR, we retain our MSIA layers only to attain the user representation at the item level for item prediction, without any other auxiliary losses.

We show the performance results in Figure 4(a). We can observe that our MSIA layers outperform all the other attention layers, which further demonstrates their effectiveness of learning attentions from multiple sequences and adaptively fusing them to learn the user representations.

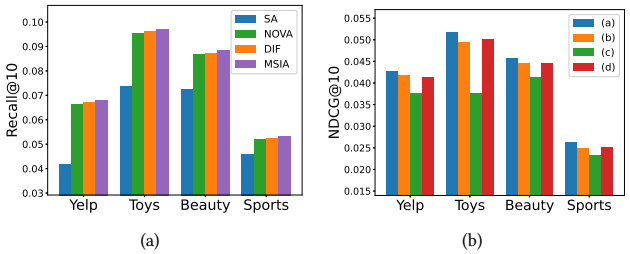

Figure 4: (a) The performance (Recall@10) of different attention layers on four datasets. (b) The performance (NDCG@10) of our MSSR using different item predictors for calculating the prediction scores on four datasets.

4.4.2 ***Item Predictor.*** We compare the following four predictors: (a) our learnable weighted summation as Eq.(18); (b) the direct summation, i.e., $\hat{\boldsymbol{p}}_{v_i} = \boldsymbol{r}_v^u \left(\boldsymbol{e}_v^i\right)^\top + \sum_{a_k \in \mathcal{A}} \boldsymbol{r}_{a_k}^u \left(\boldsymbol{e}_{a_k}^i\right)^\top$; (c) the dot product between the fused representation obtained by a full-connected layer and the item embedding [35], i.e, $\hat{\boldsymbol{p}}_{v_i} = \text{FC}\left(\left[\boldsymbol{r}_v^u; \boldsymbol{r}_{a_1}^u; \ldots; \boldsymbol{r}_{a_K}^u\right]\right)\left(\boldsymbol{e}_v^i\right)^\top$;

and (d) the dot product between the user representation at the item level and the item embedding, i.e., $\hat{p}_{v_i} = r_v^u \left(e_v^i\right)^\top$.

In Figure 4(b), we can see that the predictors (a), (b) and (d) outperform predictor (c) across all the datasets, which indicates that maintaining consistent types of information between representation and embedding is preferable when calculate dot products. Moreover, considering more dot products and learning weights to quantify the importance of different dot products help improve the performance, which can be proved by the fact that our predictor (a) outperforms both predictors (b) and (d). This exploratory study demonstrates the effectiveness of our item predictor.

4.5 Visualization and Case Study (RQ4&RQ5)

Our MISA layers employ a learnable vector to quantify the importance of different attentive pairs. In Figure 5, we visualize the weight distribution in vector $\boldsymbol{w}$ of Eq.(7) on Yelp. As we treat the categories and the positions as the side information, we have nine attentive pairs. Taking "ic" as an example, it denotes the fusion weight of the attention matrix that is calculated using the query matrix of the item sequence and the key matrix of the corresponding category sequence. We can see that "ii", "cc", and "ic" are the three most dominant pairs, which is in line with the intuition that we are more inclined to rely on the previously visited shops and their categories, when considering the next shop to visit. It shows the rationality of our MSSR that adaptively fuses various attentions of both intra- and inter-sequence interactions.

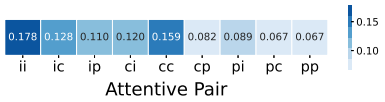

Figure 5: The visualization of the weight vector on Yelp.

In Figure 6, we visualize the attention distributions about the three aforementioned pairs on a specific interaction sequence which contains ten items and their corresponding categories. The target item to be predicted in this case is "China Impression", and its corresponding category is "Chinese Restaurants". We can observe that our MSSR can endow the prominent attention weights to the target category and the similar category "Chinese Noodles", which demonstrates the interpretability of our MSSR.

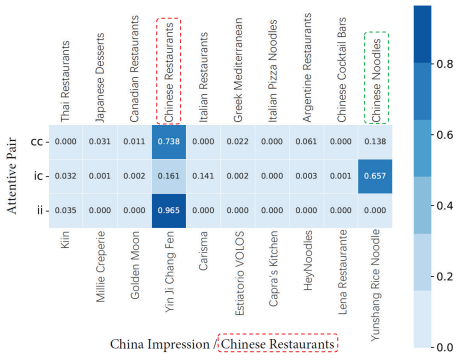

Figure 6: A case of attention distributions about three attentive pairs, i.e., "cc", "ic", "ii". The target and similar categories are highlighted with red and green borders, respectively.

5 RELATED WORK

Sequential Recommendation. Early works [8, 24] utilize Markov chains and matrix factorization to capture the sequential patterns. With the development of deep learning, some techniques are applied in SR, such as RNNs [9, 10, 23, 31], CNNs [27, 32] and GNNs [1, 20, 34]. Some works [12, 18, 26] migrate the self-attention mechanism from Transformers [28] and show promising performance. Recently, there are lots of works [2, 17, 21, 22, 29] proposing to apply contrastive learning to SR to learn the better user representation. However, these sequential recommenders only consider the item sequences and do not exploit the valuable side information.

Side-information Integrated Sequential Recommendation. The early work pRNN [11] feeds the concatenation of the item and side-information embeddings into GRU layers to learn the sequential patterns. S^3-Rec [41] designs pre-training self-supervised tasks to learn the correlations between the item and its side information. ICAI-SR [33] constructs heterogeneous graphs to attain the fused item embeddings and feed them to the sequential model. NOVA-SR [19] proposes a non-invasive attention mechanism to enhance the attention calculation. DIF-SR [30] decouples the attention calculation of items and side information to attain the fused attention matrices. However, these works on SISR only learn the user representation at the item level and seldom consider the complex association of multiple sequences. Some works learns the user representations at both the item and side-information levels. FDSA [35] and its enhanced version [7] use separate self-attention layers to extract the representations at two levels, which are then concatenated and fed into the prediction layer. Similarly, CAFE [16] takes the sum of two representations to calculate the prediction score. In the industrial scenarios of click-through rate prediction, there are also some works [4, 37–40] incorporating some side information to calculate the attention between the target item and each interaction in the user's historical sequence. Among them, TIN [40] proposes the target-aware temporal encoding, which helps to capture the semantic and temporal correlation between the target item and historical interactions.

6 CONCLUSION

In this paper, we propose a novel model named MSSR for side-information integrated sequential recommendation. Specifically, we design the multi-sequence integrated attention layer to adaptively leverage both intra-sequence and inter-sequence interaction to learn a user's multiple representations. Moreover, we introduce a user representation alignment module to optimize multiple representations of the same user by leveraging the self-supervised signals. And our side information predictor can further refine the user's representations during training. For item prediction, we consider the available side information to enable a comprehensive measurement of the user's preferences. Extensive experiments show the effectiveness of our MSSR. Visualization and case study also demonstrate its rationality and interpretability.

ACKNOWLEDGEMENT

We thank the support of National Natural Science Foundation of China Nos. 62172283 and 62272315.

A ETHICAL CONSIDERATIONS

During our research, we have consistently followed ethical guidelines. We used commonly available open-source datasets for our experiments, and any user-related information in these datasets has been anonymized, ensuring the protection of user privacy. Moreover, we strictly adhered to data usage guidelines. We utilize the datasets only to validate the effectiveness of our proposed model, without any involvement in commercial profit. As for the security considerations, we ensure that our item recommendations do not compromise the security or safety of the user.

REFERENCES

[1] Jianxin Chang, Chen Gao, Yu Zheng, Yiqun Hui, Yanan Niu, Yang Song, Depeng Jin, and Yong Li. 2021. Sequential Recommendation with Graph Neural Networks. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 378–387.

[2] Yongjun Chen, Zhiwei Liu, Jia Li, Julian J. McAuley, and Caiming Xiong. 2022. Intent Contrastive Learning for Sequential Recommendation. In *Proceedings of the 31st International Conference on World Wide Web*. 2172–2182.

[3] Alexander Dallmann, Daniel Zoller, and Andreas Hotho. 2021. A Case Study on Sampling Strategies for Evaluating Neural Sequential Item Recommendation Models. In *Proceedings of the 15th ACM Conference on Recommender Systems*. 505–514.

[4] Yufei Feng, Fuyu Lv, Weichen Shen, Menghan Wang, Fei Sun, Yu Zhu, and Keping Yang. 2019. Deep Session Interest Network for Click-Through Rate Prediction. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. 2301–2307.

[5] Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 6894–6910.

[6] Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality Reduction by Learning an Invariant Mapping. In *Proceedings of 2006 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1735–1742.

[7] Yongjing Hao, Tingting Zhang, Pengpeng Zhao, Yanchi Liu, Victor S. Sheng, Jiajie Xu, Guanfeng Liu, and Xiaofang Zhou. 2023. Feature-Level Deeper Self-Attention Network with Contrastive Learning for Sequential Recommendation. *IEEE Transactions on Knowledge and Data Engineering* 35, 10 (2023), 10112–10124.

[8] Ruining He and Julian McAuley. 2016. Fusing Similarity Models with Markov Chains for Sparse Sequential Recommendation. In *Proceedings of the 16th IEEE International Conference on Data Mining*. 191–200.

[9] Balázs Hidasi and Alexandros Karatzoglou. 2018. Recurrent Neural Networks with Top-k Gains for Session-based Recommendations. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 843–852.

[10] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2016. Session-based Recommendations with Recurrent Neural Networks. In *Proceedings of the 4th International Conference on Learning Representations*.

[11] 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*. 241–248.

[12] Wang-Cheng Kang and Julian J. McAuley. 2018. Self-Attentive Sequential Recommendation. In *Proceedings of the 18th IEEE International Conference on Data Mining*. 197–206.

[13] Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. 2018. Bilinear Attention Networks. In *Proceedings of the Annual Conference on Neural Information Processing Systems*. 1571–1581.

[14] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *Proceedings of the 3rd International Conference on Learning Representations*.

[15] Walid Krichene and Steffen Rendle. 2020. On Sampled Metrics for Item Recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1748–1757.

[16] Jiacheng Li, Tong Zhao, Jin Li, Jim Chan, Christos Faloutsos, George Karypis, Soo-Min Pantel, and Julian J. McAuley. 2022. Coarse-to-Fine Sparse Sequential Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2082–2086.

[17] Xuewei Li, Aitong Sun, Mankun Zhao, Jian Yu, Kun Zhu, Di Jin, Mei Yu, and Ruiguo Yu. 2023. Multi-Intention Oriented Contrastive Learning for Sequential Recommendation. In *Proceedings of the 16th ACM International Conference on Web Search and Data Mining*. 411–419.

[18] Jing Lin, Weike Pan, and Zhong Ming. 2020. FISSA: Fusing Item Similarity Models with Self-Attention Networks for Sequential Recommendation. In *Proceedings of the 14th ACM Conference on Recommender Systems*. 130–139.

[19] Chang Liu, Xiaoguang Li, Guohao Cai, Zhenhua Dong, Hong Zhu, and Lifeng Shang. 2021. Non-invasive Self-attention for Side Information Fusion in Sequential Recommendation. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*. 4249–4256.

[20] Chen Ma, Liheng Ma, Yingxue Zhang, Jianing Sun, Xue Liu, and Mark Coates. 2020. Memory Augmented Graph Neural Networks for Sequential Recommendation. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*. 5045–5052.

[21] Xiuyuan Qin, Huanhuan Yuan, Pengpeng Zhao, Junhua Fang, Fuzhen Zhuang, Guanfeng Liu, Yanchi Liu, and Victor S. Sheng. 2023. Meta-optimized Contrastive Learning for Sequential Recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 89–98.

[22] Ruihong Qiu, Zi Huang, Hongzhi Yin, and Zijian Wang. 2022. Contrastive Learning for Representation Degeneration Problem in Sequential Recommendation. In *Proceedings of the 15th ACM International Conference on Web Search and Data Mining*. 813–823.

[23] Massimo Quadrana, Alexandros Karatzoglou, Balázs Hidasi, and Paolo Cremonesi. 2017. Personalizing Session-based Recommendations with Hierarchical Recurrent Neural Networks. In *Proceedings of the 11th ACM Conference on Recommender Systems*. 130–137.

[24] 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*. 811–820.

[25] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *The Journal of Machine Learning Research* 15, 1 (2014), 1929–1958.

[26] 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*. 1441–1450.

[27] Jiaxi Tang and Ke Wang. 2018. Personalized top-N Sequential Recommendation via Convolutional Sequence Embedding. In *Proceedings of the 11th ACM International Conference on Web Search and Data Mining*. 565–573.

[28] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *Proceedings of the Annual Conference on Neural Information Processing Systems*. 5998–6008.

[29] Xu Xie, Fei Sun, Zhaoyang Liu, Shiwen Wu, Jinyang Gao, Jiandong Zhang, Bolin Ding, and Bin Cui. 2022. Contrastive Learning for Sequential Recommendation. In *38th IEEE International Conference on Data Engineering*. 1259–1273.

[30] Yueqi Xie, Peilin Zhou, and Sunghun Kim. 2022. Decoupled Side Information Fusion for Sequential Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1611–1621.

[31] Feng Yu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. A Dynamic Recurrent Model for Next Basket Recommendation. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 729–732.

[32] 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 12th ACM International Conference on Web Search and Data Mining*. 582–590.

[33] Xu Yuan, Dongsheng Duan, Lingling Tong, Lei Shi, and Cheng Zhang. 2021. ICAI-SR: Item Categorical Attribute Integrated Sequential Recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1687–1691.

[34] Mengqi Zhang, Shu Wu, Xueli Yu, Qiang Liu, and Liang Wang. 2023. Dynamic Graph Neural Networks for Sequential Recommendation. *IEEE Transactions on Knowledge and Data Engineering* 35, 5 (2023), 4741–4753.

[35] 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 28th International Joint Conference on Artificial Intelligence*. 4320–4326.

[36] Wayne Xin Zhao, Shanlei Mu, Yupeng Hou, Zihan Lin, Yushuo Chen, Xingyu Pan, Kaiyuan Li, Yujie Lu, Hui Wang, Changxin Tian, Yingqian Min, Zhichao Feng, Xinyan Fan, Xu Chen, Pengfei Wang, Wendi Ji, Yaliang Li, Xiaoling Wang, and Ji-Rong Wen. 2021. RecBole: Towards a Unified, Comprehensive and Efficient Framework for Recommendation Algorithms. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*. 4653–4664.

[37] Zuowu Zheng, Xiaofeng Gao, Junwei Pan, Qi Luo, Guihai Chen, Dapeng Liu, and Jie Jiang. 2022. AutoAttention: Automatic Field Pair Selection for Attention in User Behavior Modeling. In *Proceedings of the 22nd International Conference on Data Mining*. 803–812.

[38] Guorui Zhou, Na Mou, Ying Fan, Qi Pi, Weijie Bian, Chang Zhou, Xiaoqiang Zhu, and Kun Gai. 2019. Deep Interest Evolution Network for Click-Through Rate Prediction. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*.

5941–5948.

[39] Guorui Zhou, Xiaoqiang Zhu, Chengru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep Interest Network for Click-Through Rate Prediction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1059–1068.

[40] Haolin Zhou, Junwei Pan, Xinyi Zhou, Xihua Chen, Jie Jiang, Xiaofeng Gao, and Guihai Chen. 2023. Temporal Interest Network for Click-Through Rate Prediction. *arXiv preprint arXiv:2308.08487* (2023).

[41] 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 and Knowledge Management*. 1893–1902.