Towards Multi-Interest Pre-training with Sparse Capsule Network

Zuoli Tang
tangzuoli@whu.edu.cn
Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education, School of Cyber Science and Engineering, Wuhan University
China

Lin Wang
fred.wl@antgroup.com
Ant Group
China

Lixin Zou[*]
zoulixin@whu.edu.cn
Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education, School of Cyber Science and Engineering, Wuhan University
China

Xiaolu Zhang
yueyin.zxl@antgroup.com
Ant Group
China

Jun Zhou
jun.zhoujun@antgroup.com
Ant Group
China

Chenliang Li[*]
cllee@whu.edu.cn
Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education, School of Cyber Science and Engineering, Wuhan University
China

ABSTRACT

The pre-training paradigm, *i.e.,* learning universal knowledge across a wide spectrum of domains, has increasingly become a new de-facto practice in many fields, especially for transferring to new domains. The recent progress includes universal pre-training solutions for recommendation. However, we argue that the common treatment utilizing the masked language modeling or simple data augmentation via contrastive learning is not sufficient for pre-training a recommender system, since a user's intent could be more complex than predicting the next word or item. It is more intuitive to go a step further by devising the multi-interest driven pre-training framework for universal user understanding. Nevertheless, incorporating multi-interest modeling in recommender system pre-training is non-trivial due to the dynamic, contextual, and temporary nature of the user interests, particularly when the users are from different domains. The limited effort on this line has greatly rendered it as an open question.

In this paper, we propose a novel **M**ulti-**I**nterest P**r**e-training with Sp**a**rse **C**apsu**le** framework (named MIRACLE). MIRACLE performs a universal multi-interest modeling with a sparse capsule network and an interest-aware pre-training task. Specifically, we utilize a text-aware item embedding module, including an MoE adaptor and a deeply-contextual encoding component, to model contextual and transferable item representations. Then, we propose a sparse interest activation mechanism coupled with a position-aware capsule network for adaptive interest extraction. Furthermore, an interest-level contrastive pre-training task is introduced to guide the sparse capsule network to learn universal interests precisely. We conduct extensive experiments on eleven real-world datasets and eight baselines. The results show that our method significantly outperforms a series of SOTA on these benchmark datasets. The code is available at https://github.com/WHUIR/Miracle.

CCS CONCEPTS

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

KEYWORDS

Pre-trained Model, Sequential Recommendation, Multi-Interest Learning

ACM Reference Format:
Zuoli Tang, Lin Wang, Lixin Zou, Xiaolu Zhang, Jun Zhou, and Chenliang Li. 2023. Towards Multi-Interest Pre-training with Sparse Capsule Network. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23), July 23–27, 2023, Taipei, Taiwan.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3539618.3591778

1 INTRODUCTION

Sequential recommendation, which predicts users' next interaction based on their historical interaction behaviors, has been widely adopted in plenty of online services, e.g., E-commerce, advertisement, social media, etc., for providing personalized suggestions. Focusing on modeling the historical interactions, traditional solutions, such as collaborative filtering (CF) [11, 18], sequential neural networks [7, 20], have achieved great success. However, all these methods assume a static representation for each user and encode the user's interests in a dense representation, which is conflict with

[*] Corresponding author. Work done when Zuoli Tang was an intern at Ant Group.

SIGIR '23, July 23–27, 2023, Taipei, Taiwan

ACM ISBN 978-1-4503-9408-6/23/07…$15.00
https://doi.org/10.1145/3539618.3591778

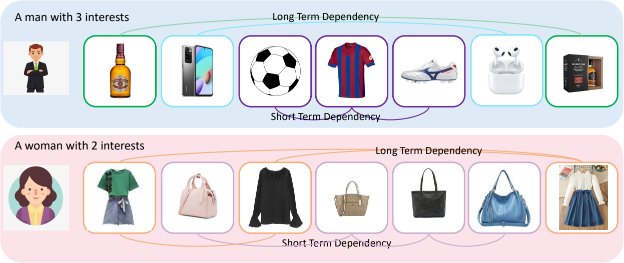

Figure 1: The example illustrates the multi-interest nature that is dynamic, contextual and temporal.

the fact that the user typically has multiple interests and leads in the difficulty of representing user's diverse tastes. Therefore, multi-interest modeling methods, such as attention mechanism [3, 21] and dynamic routing [13], are proposed to significantly push the frontier of sequential recommendation.

More recently, the great success of the pre-training followed by fine-tuning in natural language processing (NLP) [5] encourages the endeavour of leveraging the pre-training for sequential recommendation. Some pre-training frameworks have been proposed for better initializing the parameter in single-domain [20, 28] or transferring knowledge across different domains [8, 15]. Particularly, the cross-domain methods bridge different domains with the item's text information (e.g., product description, product name) for the universal item representation, which therefore leverages the knowledge of different domains and the semantic knowledge stored in the pre-trained language model [5]. As expected, these methods achieve state-of-the-art performance on the sequential recommendation. However, these methods represent each user as a fixed vector and fail to capture the user's diverse interests. Therefore, it is noteworthy to consider the user's multiple interests when pre-training a recommender model.

Though explicitly modeling the multi-interest in pre-training a recommender system is intuitive and beneficial, we need to overcome the following obstacles: **(1)** designing a unified neural network to model diverse, context-dependent, and temporal user interests in a universal space is challenging, especially for users from varying domains. Existing methods for multi-interest learning often require a fixed number of interests and overlook the crucial temporal patterns of the historical sequential sequence. However, this is not practical in real-world situations as the interests of users can vary significantly among different groups, especially across different domains. Additionally, the pre-trained models commonly adopt auto-regressive architecture liking GPT [2], potentially hindering the modeling of contextual information for universal user understanding. As shown in Figure 1, the man in our example has three interests, while the women's interests are more concentrated with only two interests. This indicates that the user's interest number should be dynamic, not pre-defined. Furthermore, as illustrated in the same example, long-term and short-term contexts are both useful for interest modeling, but their importance for next-item prediction may differ. **(2)** crafting a pre-training task for multi-interest learning is a novel problem with limited attempts in previous studies. Appropriate pre-training tasks combined with the universal multi-interest neural network can significantly enhance performance, while unsuitable tasks may not be beneficial or even hinder performance.

To overcome the above two challenges, in this paper, we propose a framework named Miracle, **M**ulti-**I**nterest P**re**-training with Sp**ar**se **C**apsu**le** Network. The framework includes a universal multi-interest modeling architecture consisting of a sparse capsule network and an interest-aware pre-training task. Specifically, for multi-interest modeling, Miracle utilizes a text-aware encoder for generating transferable item embeddings. Following earlier effort [8], this text-aware item encoder integrates cross-domain semantic information with an MoE adapter and a deeply contextual-aware bi-directional Transformer for item embedding generation. The sparse capsule network has two parts: sparse capsule activation and position-wise dynamic routing. Each capsule works as a proxy for the corresponding interest. The sparse capsule activation finds the number of interests per user and activates relevant capsules, computing the coupling coefficient for the first round of dynamic routing to avoid interest coupling caused by random or zero initialization. The position-wise dynamic routing, with trainable position embeddings and an attention mechanism, dynamically merges temporal information for multi-interest extraction. Finally, we devise a novel interest-level contrastive learning task in the pre-training stage to enhance universal multi-interest learning by establishing the connections between different interests across the domains. In a nutshell, we make the following contributions in this paper:

- This is the first attempt to explore the transferability of modeling multi-interest for pre-training driven sequential recommendation. We argue that performing universal preference transfer in the user interest level is more fruitful for recommender system pre-training.
- We propose a novel sparse multi-interest modeling framework that fuses temporal information and adaptively extracts the user's interests for better user understanding. Also, a new contrastive multi-interest pre-training objective is proposed to capture the correlation of items and interests in a cross-domain manner.
- We conduct comprehensive experiments on several real-world datasets to verify the effectiveness of our method over eight baselines. Extensive ablation studies and empirical analyses further verify our design choices.

2 RELATED WORK

This section provides a brief overview of representative efforts relevant to our work.

2.1 Sequential Recommendation

Sequential recommendation aims at predicting the next item given the user's historical items, which is a widely studied problem in recommender systems. The pioneering works assume the sequential behavior follows Markov Chain and approximate the item-item transition matrix for next item prediction [16]. With the prosperity of deep learning, various neural network architectures have been proposed for modeling long-term and short-term dependency at the same time. The earlier efforts include utilizing RNN Hidasi et al. [7] and CNN [22]. Recently, the Transformer's advantages in modeling long-term dependency made its variants (*e.g.*, SASRec [9], BERT4Rec [20]) the mainstream for sequential recommendation. Furthermore, due to the strength in modeling high-order dependence, the graph neural networks (GNN) have been used and achieved promising results [25, 26]. However, these solutions are all based on item id, which does not contain transferable semantics, so it is difficult to integrate knowledge from other domains or be utilized by other domains. Additionally, a user's preference is dynamic, thus a single fixed representation is not sufficient for user understanding.

2.2 Multi-Interest Learning

Due to the limitation of a single vector, there are some explorations for generating multiple vectors to cover users' multi-level interests. For example, MIND [13] utilizes dynamic routing [17] for multi-level interest extraction. ComiRec [3] adjusts the dynamic routing structure and propose a new multi-interest learning method based on self-attention. Furthermore, UMI [4] exploits the multiple attributes contained in the user profile to enhance multi-interest learning. MGNM [23] integrates user embedding into the construction of graph structure, and ingeniously combines graph neural network with dynamic routing to learn user's multiple interests from different levels. SINE [21] is an attention-based multi-interest learning method. It firstly construct the user's interest prototypes, and then uses an attention mechanism to extract user multi-interest between interactive items and interest prototypes. However, the aforementioned works need to pre-define the interest number for all users. It is hard to identify an optimal value globally. Furthermore, most of these multi-interest methods lack explicit modeling of temporal information.

2.3 Pre-training Recommender System

Recently, pre-training with sophisticated self-supervised learning tasks has gained increasing attention for recommender systems due to its great success in the NLP domain [2, 5]. Sun et al. [20] propose BERT4Rec by training an transformer-based model with an objective similar to the masked language modeling (MLM) [5]. Qiu et al. [15] further pre-train the model with users' reviews and leverages the content information for cross-domain recommendation. S^3-Rec [28] studies data sparsity problem and warms up the model parameters by maximizing mutual information between attributes, items and sub-sequence. For better cross-domain knowledge transfer, UniRec [8] choose to pre-train the model with text description. Compared with these works, we combine pre-training with multi-interest learning and propose a interest-level contrastive objective for better parameters warm up.

3 METHODOLOGY

This section outlines the sequential recommendation problem and provides a comprehensive introduction to the proposed Miracle, featuring its key components: text-aware item embedding and sparse capsule network, as shown in Figure 2. At end of this section, the pre-training and fine-tuning process of Miracle is discussed.

3.1 Problem Formulation

Given a set of users $\mathcal{U}$ and a set of items $\mathcal{V}$, for each user $u \in \mathcal{U}$, we can organize their interaction history as a chronological sequence $s^u = (v_1^u, v_2^u, ..., v_N^u)$, where $v_t^u \in \mathcal{V}$ is the t-th item interacted with by the user and N is the sequence maximum capacity. Additionally, each item $v \in \mathcal{V}$ is associated with a textual description, denoted as sequential tokens $T_v = (t_1, t_2, ..., t_{|T_v|})$. Formally, we can model a recommender as a scoring function f that maps the interaction history s^u and item v_i to a score $f(v_i, s^u)$. The goal of sequential recommendation is thereby formulated as generating a top-K list according to f that covers the next item.

Furthermore, for pre-training a recommender, there are two types of datasets: D_p and D_f, which represent the pre-training dataset and downstream dataset respectively. The model parameters of the recommender are first warmed up over the pre-training dataset D_p with pre-training objective $\ell_{pre-train}$. We then fine-tune the parameters and validate the model over the downstream dataset D_f with fine-tune objective $\ell_{fine-tune}$. Generally, D_p is a resource-rich domain or a mixup of multiple domains, while D_f contains fewer data. Moreover, it generally holds that there is a semantic gap between D_p and D_f. The key point is how to fully exploit the knowledge of D_p and transfer them to D_f for better recommendation performance of the latter.

3.2 Text-Aware Item Embedding

Considering the transferability and accessibility of text information, we follow the work in [8] to encode the textual description of an item as item embeddings. In the following, we will dive into the specifics of this process.

3.2.1 Pre-trained Language Model-based Text Encoder. We employ $BERT_{base}$ [5], a widely adopted pre-trained language model (PLM), to capture the semantic feature from text description T_{v_i} of item v_i. Specifically, $BERT_{base}$ is a 12-layer Transformer which contains 12 self-attention heads per layer. Following the common practice, we place a special token $[CLS]$ in front of T_{v_i}, and feed the resultant text sequence into $BERT_{base}$:

$$\boldsymbol{t}_i = BERT_{base}([[CLS], t_1, ..., t_{|T_{v_i}|}]), \tag{1}$$

$\boldsymbol{t}_i \in \mathbb{R}^{d_b}$ means the text representation of v_i, which is the latent feature output from the last transformer layer for token $[CLS]$, and d_b is the corresponding dimension size of $BERT_{base}$. It should be noted that $BERT_{base}$ is only used to encode the textual information and its parameters are frozen.

MoE Adaptor. To facilitate the universal preference transfer from different domains with the text representations of items, we merge multiple domain datasets to form the pre-training dataset D_p. However, due to the semantic gap between different domains, the representations output by $BERT_{base}$ are not in the same semantic space for items of different domains. Some existing work consider this phenomenon as domain bias [12]. Therefore, it is necessary to transform the semantic information of different domains into a uniform semantic space. An intuitive idea is to use a fully connected layer (FC) for semantic space transformation. However, using the same FC for all domains will weaken the representation ability. Therefore, we can utilize a mixture-of-expert (MoE) [19] network to enhance

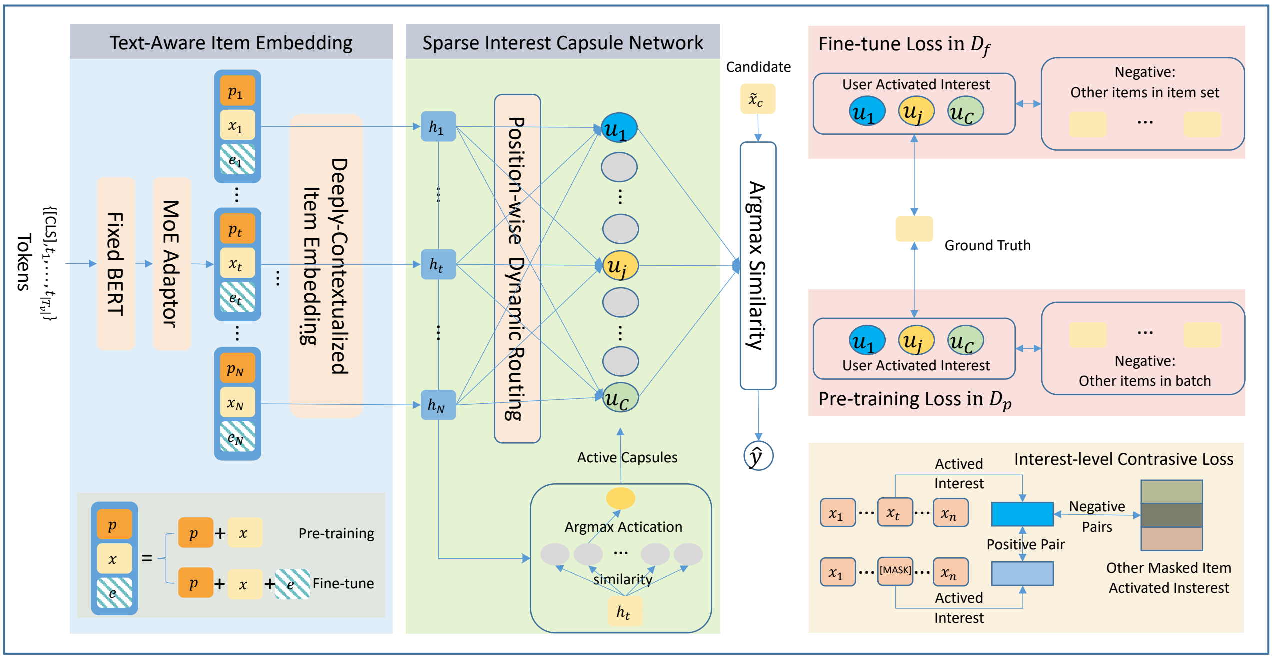

Figure 2: The overview of the proposed MIRACLE

the ability to eliminate domain bias as follows:

$$x_i = \sum_{k=1}^{G} g_k \cdot Expert_k(t_i), \tag{2}$$

Here, g_k is the gate weight for the k-th expert according to the input. $Expert_k(\cdot)$ is the k-th expert network. Particularly, the gate $\boldsymbol{g}$ and $Expert_k(\boldsymbol{t}_i)$ are derived as follows,

$$\boldsymbol{g} = softmax(\boldsymbol{t}_i \cdot W_1 + \boldsymbol{b}_1) \tag{3}$$

$$Expert_k(\boldsymbol{t}_i) = (\boldsymbol{t}_i - \boldsymbol{b}_2^k) \cdot W_2^k, \tag{4}$$

where $W_1 \in \mathbb{R}^{d_b \times G}, \boldsymbol{b_1} \in \mathbb{R}^G$ and $W_2^k \in \mathbb{R}^{d_b \times d_n}, \boldsymbol{b_2^k} \in \mathbb{R}^{d_n}$, d_n is the hidden size of each expert network, G is the expert number, and softmax($\cdot$) refers to the *softmax* function.

3.2.2 Deeply-contextualized Item Embedding. Given the sequential dense text representation of items $(\boldsymbol{x}_1, \dots, \boldsymbol{x}_N)$, we adopt Transformer [24] as the backbone network for modeling the long-short term dependency between items. Specifically, considering the sequential nature of the historical behaviors, we then add a trainable positional embedding $P \in \mathbb{R}^{N \times d_n}$ to the corresponding text representation:

$$F^0 = [\boldsymbol{x}_1 + P_1, \boldsymbol{x}_2 + P_2, .., \boldsymbol{x}_N + P_N], \tag{5}$$

Then, we model the contextual information for each item in the sequence with the L-layer Transformer as follows:

$$F^l = FFN(MHA(F^{l-1})), \tag{6}$$

where MHA (multi-head self-attention) and FFN (feed-forward network) are the components of Transformer. For concise, we omit the detail of MHA and FFN, which can be found in [24]. Finally, we obtain the output of last Transformer layer F^L as the contextualized text-aware item embeddings, denoted as $H = [\boldsymbol{h}_1, \dots, \boldsymbol{h}_N]$.

3.3 Sparse Interest Capsule Network

Given these contextualized item embeddings, we design a sparse interest capsule network to extract multiple interests. Compared with the existing multi-interest based methods that uitlize a vanilla capsule networks [3, 13], we employs a sparse interest capsule activation mechanism to flexibly determine the number of interests and initialize the coupling coefficient in its dynamic routing stage. In the following, we first present the sparse interest capsule activation mechanism. Then, we utilize position attention to emphasize the importance of different positions. Last, the dynamic routing mechanism is used to adaptively aggregate the user's multiple interests.

3.3.1 Sparse Interest Capsule Activation. Specifically, we first form a base vector matrix $A \in \mathbb{R}^{C \times d_n}$ to denote the universal interest space, where C is the interest number[1]. For a given user, it is intuitive that only a subset of capsules will be activated to reflect the user's diverse interests. Hence, we simply calculate the similarity s_{ij} between each item and each base vector,

$$s_{ij} = sim(\boldsymbol{h}_i, \boldsymbol{a}_j) \cdot (1 + \epsilon), \tag{7}$$

where $sim(\cdot, \cdot)$ is the inner product, $\mathbf{h}_i$ is the contextualized item embedding of i-th item in H, $\boldsymbol{a}_j$ is the j-th base vector, and ϵ is the Gaussian noise to enable exploration. Then, we activate one interest capsule for each item, and generate the corresponding capsule activation vector $\boldsymbol{m}_i$ as follows:

$$m_{ij} = \begin{cases} -\infty, & \text{if j not in any arg max}(\mathbf{s}_i); \\ 0, & \text{otherwise} \end{cases}, \tag{8}$$

Here, $m_{ij} = 0$ means that the j-th interest capsule is activated, and $\mathbf{s}_i = [s_{i1}, \cdots, s_{iC}]$.

[1] These C vectors are considered as basic universal interests across all domains.

3.3.2 Position-wise Capsule Network. The utilization of sequential information has been consistently overlooked in many existing solutions when a capsule networks is utilized for multi-interest extraction. Therefore, we utilize the attention mechanism to calculate the importance of items at different positions dynamically:

$$\boldsymbol{o} = softmax(tanh((H + \tilde{P})W_3 + \boldsymbol{b}_3)W_4 + \boldsymbol{b}_4), \tag{9}$$

where $\tilde{P} \in \mathbb{R}^{N \times d_n}$ is another trainable position embeddings tailored for multi-interest learning, $W_3 \in \mathbb{R}^{d_n \times d_n}$ and $W_4 \in \mathbb{R}^{d_n \times N}$ are the weight matrix, $\boldsymbol{b}_3 \in \mathbb{R}^{d_n}$ and $\boldsymbol{b}_4 \in \mathbb{R}^{N}$ are the bias, and $tanh(\cdot)$ indicates the nonlinear activation function.

3.3.3 Multi-Interest Extractor Layer. Then we further utilize an FC layer including residual connection [6] and layer normalization[1] to derive the latent interest features:

$$\boldsymbol{q}_i = LayerNorm(tanh(\boldsymbol{h}_i W_5 + \boldsymbol{b}_5) + \boldsymbol{h}_i), \tag{10}$$

where $W_5 \in \mathbb{R}^{d_n \times d_n}$ and $\boldsymbol{b}_5 \in \mathbb{R}^{d_n}$ are trainable parameters. Then, we calculate the output of the j-th capsule j as follows:

$$z_j = \sum_{i=1}^{N} c_{ij} o_i \boldsymbol{q}_i, \tag{11}$$

Here, c_{ij} is the coupling coefficient between the i-th item and the j-th capsule, o_i is the i-th element of $\mathbf{o}$. Recall that each historical item only activates a single interest capsule, we hence restrict the scope of dynamic routing over these activated ones. Specifically, the coupling coefficient is calculated iteratively as follows:

$$c_{ij} = \frac{exp(b_{ij} + m_{ij})}{\sum_{k=1}^{C} exp(b_{ik} + m_{ik})}, \tag{12}$$

$$\boldsymbol{u}_j = \frac{||z_j||^2}{||z_j||^2 + 1} \frac{z_j}{||z_j||}, \tag{13}$$

$$b_{ij} = b_{ij} + \boldsymbol{h}_i^T \boldsymbol{u}_j, \tag{14}$$

where b_{ij} is the agreement score indicating the relevance between the i-th item and the j-th interest capsule. Unlike previous works that initialize b_{ij} by random or zero values, we initialize $b_{ij} = s_{ij}$ to avoid interest coupling. Then, following the common setting in the dynamic routing mechanism, a nonlinear squashing function is used to normalize z_j (ref. Equation 13). We repeat the dynamic routing process τ times and the resultant $\mathbf{u}_j$ is taken as the interest representation of the user, where $j \in I_u$ and $I_u = \{j | \exists i : m_{ij} = 0\}$ contains all the activated interest capsule.

3.4 Model Training and Deployment

Here, we outline the training and inference protocol for our Miracle. First, to enhance universal multi-interest learning, we propose a novel interest-level contrastive learning task for the pre-training stage. Then, considering the coverage of interest capsules, we utilize two auxiliary losses as the regularizer, followed by the downstream fine-tuning.

3.4.1 Interest-Level Pre-training. Specifically, we augment each interaction sequence by randomly masking some items with the special token $[MASK]$. As to the original sequence, we use $\tilde{u}_i$ to represent the user interest representation corresponding to the interest activated by the item to be masked, say the i-th masked item in the same batch. Hence, the corresponding user interest representation $\tilde{u}_i'$ generated by the augmented counterpart is treated as positive sample. On the contrast, the other masked item in the sequences of the same batch are selected as negative samples. However, since the activated interest capsule is controlled by the sparse capsule activation mechanism, the capsule activated by the augmented sequence may not be consistent with the original one. Therefore, we force the capsule activated vector of the augmented sequence to be identical with the capsule activated vector of the original sequence. In this sense, we can perform the interest-level contrastive learning as follows:

$$\ell_i = \sum_{i=1}^{B_m} -log \frac{exp(\tilde{u}_i^T \tilde{u}_i')}{\sum_{j=1}^{B_m} exp(\tilde{u}_i^T \tilde{u}_j')}, \tag{15}$$

where B_m represents the total number of masked items in a batch.

3.4.2 Predicting Next Item. In addition to interest-level contrastive learning, we also follow the common practice of predicting the next item. For a given interaction sequence, its corresponding target item is treated as the positive sample, while the target items of the other interaction sequences in the same batch are treated as negative samples. We use the cross-entropy loss function to optimize the model as follows:

$$\ell_{rec} = \sum_{i=1}^{B} -log \frac{exp(\hat{y}_{u_i, v_i})}{\sum_{j=1}^{B} exp(\hat{y}_{u_i, v_j})}, \tag{16}$$

where B represents training batch size, $\hat{y}_{u_i, v_j}$ is the recommendation score for user u_i towards item v_j. Note that there are more than one interest expressed by the user. Hence, the recommendation score between the user and candidate item v_j is calculated by applying max-pooling as follows:

$$\hat{y}_{u_i, v_j} = max(\boldsymbol{u}_i^T \boldsymbol{x}_j | i \in I_{u_i}) \tag{17}$$

where the text representation $\mathbf{x}_j$ of item v_j is derived according to Equation 2.

3.4.3 Capsule Regularization. In the worst case, all interest capsules have the same meaning or all items may activate the same interest capsule, which makes it difficult to express multiple interest, so it is necessary to ensure orthogonality between interest capsules and encourage a balanced allocation across capsules. We add two auxiliary losses from the perspective of regularization. To guarantee the orthogonality between different interest capsules, we enforce that the base vectors in A as follows:

$$\ell_c = \sum_{j=1}^{C} -log \frac{cos(\boldsymbol{a}_j, \boldsymbol{a}_j)}{\sum_{j'=1}^{C} cos(\boldsymbol{a}_j, \boldsymbol{a}_{j'})}, \tag{18}$$

where $cos(\cdot, \cdot)$ represents the cosine similarity. Then, we also choose to enable an even distribution of activated capsules such that fine-grained interests are well covered by all interest capsules. Here, we decide to calculate the similarity distribution $\mathbf{w}$ between items and base vectors at first:

$$\mathbf{w} = \frac{1}{B} \sum_{i \in B} \mathbf{s}_i. \tag{19}$$

Then, the capsule activation distribution of the same batch is calculated as follows:

$$\mathbf{r} = \frac{1}{B} \sum_{i=1}^{B} \mathbf{m}_i. \tag{20}$$

The coverage regularization term is then calculated as follows:

$$\ell_b = C \cdot \mathbf{w}^T \mathbf{r}, \tag{21}$$

Pre-training. With the four tasks mentioned above, the final pre-training objective is defined as a linear cominbation of these four losses:

$$\ell_{pre-train} = \ell_{rec} + \theta_1 \ell_b + \theta_2 \ell_c + \theta_3 \ell_i, \tag{22}$$

where $\theta_1, \theta_2, \theta_3$ are the hyperparameters to control the impact of different losses.

3.4.4 Fine-tuning. In the fine-tuning stage, we freeze the parameters of the sequence encoder (ref. Equation 6) and fine-tune the other modules' parameters. We use cross-entropy to calculate the recommendation loss. It should be noted that in the fine-tuning stage, we create an item embedding table $V \in \mathbb{R}^{|V| \times d_n}$ for the items in the downstream dataset, where $|V|$ denotes the number of items in the downstream dataset. For item v_i, we sum the text representation x_i and the corresponding ID embedding e_i, and feed it into the transformer encoder for generating the contextualized text-aware item embedding. The fine-tuning loss is then formulated as follows:

$$\ell_{rec} = \frac{1}{B} \sum_{i=1}^{B} -log \frac{exp(\hat{y}_{u_i, v_i})}{\sum_{v_j \in \mathcal{V}} exp(\hat{y}_{u_i, v_j})}, \tag{23}$$

$$\ell_{fine-tune} = \ell_{rec} + \theta_4 \ell_b + \theta_5 \ell_c, \tag{24}$$

where θ_4, θ_5 are the hyperparameters.

4 EXPERIMENTS

To assess the effectiveness of MIRACLE, we conduct a series of experiments on real-world benchmark datasets. Specifically, we mainly focus on the following research questions:

- **RQ1**: How does the performance of our proposed MIRACLE compare with other state-of-the-art baselines?
- **RQ2**: Whether MIRACLE has achieved ideal results in pre-training related tasks?
- **RQ3**: Are the components devised in MIRACLE helpful to the final performance?
- **RQ4**: How sensitive is our model with respect to different hyper-parameters?

4.1 Experimental Setup

4.1.1 Datasets. To meet the requirement of pre-training mentioned in Section 3.1, we choose five cross-domain datasets sourced from Amazon review datasets [14] to form a pre-training dataset D_p. Then, we select five cross-domain datasets and one cross-platform dataset as the downstream datasets D_f. We fine-tune and evaluate the MIRACLE on each downstream dataset. The details of the dataset are described as follows:

- Pre-training dataset: we select five review datasets from the Amazon review collection[2] as the cross-domain dataset, which includes the following categories, "Grocery and Gourmet Food," "Home and Kitchen," "CDs and Vinyl," "Kindle Store," and "Movies and TV."

Table 1: The Statistics of the datasets. Online Retail is a cross-platform dataset.

Dataset		Users	Items	Interactions	Sparsity
Pre-Training	Food	115,349	39,670	1,027,413	-
	CDs	94,010	64,439	1,118,563	-
	Kindle	138,436	98,111	2,204,596	-
	Movies	281,700	59,203	3,226,731	-
	Home	731,913	185,552	6451,926	-
Fine-Tuning	Scientific	8,442	4,385	59,427	1.6‰
	Pantry	13,101	4,898	126,962	1.9‰
	Instruments	24,962	9,964	208,926	0.8‰
	Arts	45,486	21,019	395,150	0.4‰
	Office	87,436	25,986	684,837	0.3‰
	Online Retail	16,520	3,469	519,906	9.0‰

- Downstream datasets: we select another five review datasets from the same Amazon review collection as the downstream datasets for fine-tuning and evaluation. To ensure the consistency of metrics, the selection of the review dataset follows the setting in [8]. Specifically, we choose the "Prime Pantry", "Industrial and Scientific", "Musical Instruments", "Arts, Crafts and Sewing" and "Office Products" as the downstream datasets. Furthermore, we select a cross-platform dataset, *i.e.,* Online Retail, from another platform to verify the generalization ability of MIRACLE. The Online Retail contains all the transactions occurring between 01/12/2010 and 09/12/2011 for a UK-based and registered non-store online retail.

Following the setting of SOTA methods [8, 9, 28], we employ the 5-core setting for all datasets by removing users or items with less than 5 interactions. Additionally, we organize each user's interactions in ascending order by timestamp. For evaluation, we employ the leave-one-out strategy for the dataset split. The latest interaction is designated for testing, the penultimate interaction for validation, and the remaining interactions for training. Table 1 reports the statistics of the datasets after preprocessing. Remarkably, the interactions from the same user across different domains are not combined as a single sequence.

4.1.2 Comparing Methods. In this work, we evaluate various state-of-the-art models for the sequential recommendation, and classify these models according to with pre-training and without pre-training.

With pre-training:

- **S^3-Rec** [28] employs four auxiliary self-supervised objectives to learn the correlations among attributes, items, subsequences and sequences through mutual information maximization.
- **BERT4Rec** [20] uses a Cloze objective loss and bidirectional self-attention mechanism to model user interaction sequences.
- **UniSRec** [8] uses the associated textual description of items to replace item IDs and learns transferable representations across different domains. This model can be effectively transferred to new recommendation domains or platforms under either inductive or transductive settings.

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

Table 2: Performance Comparison of different methods across six datasets. The best and second-best performances are highlighted in bold and underlined, respectively. A symbol of $*$ denotes that the improvement in performance in comparison to the second-best result is statistically significant with the p-value ≤ 0.05. The column labeled "Improv" denotes the proposed method's relative improvement ratios compared to the best-performing baselines.

Dataset	Metrics	GRU4Rec	SASRec	FDSA	MIND	ComiRec-DR	S^3-Rec	BERT4Rec	UniSRec	Miracle	Improv
Scientific	Recall@10	0.0602	0.1003	0.0881	0.0632	0.0313	0.0684	0.0650	0.1237	**0.1343***	+8.57%
	Recall@50	0.1407	0.2010	0.1656	0.1375	0.0913	0.1464	0.1471	0.2416	**0.2562***	+6.04%
	NDCG@10	0.0330	0.0550	0.0589	0.0402	0.0176	0.0373	0.0345	0.0663	**0.0722***	+8.90%
	NDCG@50	0.0503	0.0769	0.0757	0.0561	0.0302	0.0541	0.0522	0.0919	**0.0988***	+7.50%
Pantry	Recall@10	0.0396	0.0488	0.0406	0.0384	0.0305	0.0426	0.0385	0.0751	**0.0822***	+9.45%
	Recall@50	0.1282	0.1339	0.1159	0.1125	0.1016	0.1243	0.1281	0.1862	**0.2027***	+8.86%
	NDCG@10	0.0197	0.0218	0.0215	0.0199	0.0155	0.0187	0.0188	0.0352	**0.0395***	+12.22%
	NDCG@50	0.0385	0.0400	0.0376	0.0358	0.0306	0.0363	0.0381	0.0591	**0.0656***	+11.00%
Instruments	Recall@10	0.0876	0.1089	0.1099	0.0988	0.0769	0.1052	0.0984	0.1260	**0.1335***	+5.95%
	Recall@50	0.1609	0.1988	0.1958	0.1752	0.1528	0.1939	0.1862	0.2375	**0.2468***	+3.92%
	NDCG@10	0.0610	0.0631	**0.0809**	0.0722	0.0462	0.0678	0.0580	0.0699	0.0776	-
	NDCG@50	0.0769	0.0826	0.0994	0.0887	0.0626	0.0870	0.0770	0.0914	**0.1021***	+11.7%
Arts	Recall@10	0.0757	0.1045	0.1015	0.0906	0.0507	0.0970	0.0854	0.1270	**0.1362***	+7.24%
	Recall@50	0.1472	0.1924	0.1832	0.1591	0.1125	0.1846	0.1660	0.2377	**0.2530***	+6.43%
	NDCG@10	0.0492	0.0599	0.0705	0.0634	0.0287	0.0572	0.0493	0.0678	**0.0791***	+12.20%
	NDCG@50	0.0647	0.0789	0.0882	0.0783	0.0419	0.0763	0.0668	0.0919	**0.1046***	+13.81%
Office	Recall@10	0.0941	0.1081	0.1132	0.0971	0.0641	0.1075	0.0934	0.1268	**0.1417***	+11.75%
	Recall@50	0.1445	0.1686	0.1740	0.1415	0.1087	0.1645	0.1447	0.2025	**0.2219***	+9.58%
	NDCG@10	0.0711	0.0656	0.0862	0.0739	0.0444	0.0715	0.0662	0.0775	**0.0918***	+6.50%
	NDCG@50	0.0821	0.0787	0.0994	0.0836	0.0540	0.0839	0.0773	0.0939	**0.1093***	+9.96%
Online Retail	Recall@10	0.1495	0.1465	0.1503	0.1463	0.1175	0.1461	0.1530	0.1554	**0.1729***	+11.26%
	Recall@50	0.3784	0.3700	0.3756	0.3331	0.2996	0.3841	0.3951	0.3906	**0.4478***	+13.34%
	NDCG@10	0.0720	0.0703	0.0729	0.0778	0.0578	0.0676	0.0711	0.0715	**0.0795**	+2.19%
	NDCG@50	0.1218	0.1190	0.1218	0.1185	0.0972	0.1195	0.1221	0.1228	**0.1398***	+13.84%

Without pre-training:

- **GRU4Rec** [7] utilizes a gate recurrent unit to model user interaction sequences.
- **SASRec** [9] employs a single-directional self-attention mechanism to model interaction sequences for recommendation.
- **FDSA** [27] takes into account the transition patterns between features and items. It applies separate self-attention blocks on item-level and feature-level sequences to model item and feature transition patterns.
- **MIND** [13] designs a multi-interest extractor based on capsule networks to capture diverse user interests.
- **ComiRec** [3] proposes a controllable multi-interest framework for sequential recommendation. It comprises a multi-interest extractor and an aggregation module to balance recommendation accuracy and diversity. ComiRec-DR and ComiRec-SA are two variants of the multi-interest extractor, the former using dynamic routing and the latter using self-attention. Due to space limitation, we select the stronger variant ComiRec-DR as the baseline.

Table 3: Model performance with different sequence encoder Transformer layers L

Dataset	Metrics	L		
		1	2	3
Scientific	Recall@10	13.64	13.43	12.62
	NDCG@10	7.31	7.22	6.87
Office	Recall@10	14.17	14.17	14.13
	NDCG@10	9.45	9.18	8.99
Online Retail	Recall@10	17.15	17.29	16.51
	NDCG@10	7.89	7.95	7.37

4.1.3 Evaluation Metrics. We employ two widely used evaluation metrics, namely recall and normalized discounted cumulative gain (NDCG). Recall measures the accuracy of the recommendation, while NDCG takes into account the positions of retrieved positive items, assigning higher weight to higher positions. To avoid biased sampling evaluations, we calculate the score for each user and all items, sort them in descending order, and generate a top-K recommendation list. The experiment is repeated five times, and the average results are reported. The metric scores are presented as the average over all users on the test set, and we chooes K to be 10 and 50 respectively.

4.1.4 Implementation Details. All baseline methods are implemented using Pytorch with an Adam [10] optimizer, and their hyperparameters are configured according to the original suggestion from their

Table 4: Model Performance with different capsule count C

Dataset	Metrics	C			
		16	24	32	40
Scientific	Recall@10	13.29	12.99	13.43	12.34
	NDCG@10	7.13	7.00	7.22	6.77
Office	Recall@10	14.04	14.10	14.17	14.12
	NDCG@10	8.89	8.96	9.18	9.48
Online Retail	Recall@10	16.64	16.66	17.29	16.99
	NDCG@10	7.49	7.64	7.95	7.98

papers. For the proposed MIRACLE, the maximum sequence length is set to 20, the number of experts is set to 8, the number of transformer layers in the sequence encoder is set to 2, the dimension of the embedding is set to 300, and the number of interest capsules is set to 32. In the pre-training stage, the learning rate is set to 0.001, and the batch size is set to 2048. In the fine-tuning stage, the learning rate is set to 0.001 and the batch size is set to 1024, the probability of a random masking in interest-level contrastive learning is set to 0.2.

4.2 Overall Comparison (RQ1)

In Table 2, we compare the MIRACLE with the baseline methods in the five cross-domain downstream datasets and one cross-platform downstream dataset. From the table, we have the following observations.

Pre-training is an effective scheme for transferring knowledge between different domains. The pre-training based approaches, UniSRec and MIRACLE, substantially outperform other baselines, demonstrating the effectiveness of pre-training in extracting text-based knowledge across different domains. Also, compared with the baseline methods, MIRACLE has achieved the best performance in almost all metrics of all datasets. We can observe consistent improvements in terms of Recall metric across all datasets both in K=10 and K=50. More specifically, the relative recall improvements range from 6.09% to 13.34%. For the NDCG metric, MIRACLE outperforms most baselines except NDCG@10 in the Scientific dataset. And the relative improvements in NDCG range from 2.19% to 13.84% for MIRACLE. The best baseline method is UniSRec, which achieves the second best in 16 out of 24 settings. Our method has substantially surpassed UniSRec in all settings, indicating the superiority of incorporating multi-interest learning under the pre-training scheme.

Auxiliary text information is beneficial for improving the sequential recommendation performance. Among all sequential baselines, SASRec and FDSA perform the best, FDSA gets higher NDCG than SASRec significantly in four datasets. Also, self-attention is effective in modeling sequence information. The comparison between GRU4Rec and SASRec indicates the superiority of self-attention in modeling sequence information. This observation is also consistent in the relevant literature. Due to the inconsistency between the Cloze task of BERT4Rec and the next item recommendation task, BERT4Rec does not achieve good results compared with other transformer based method on five datasets, such as SASRec and FDSA.

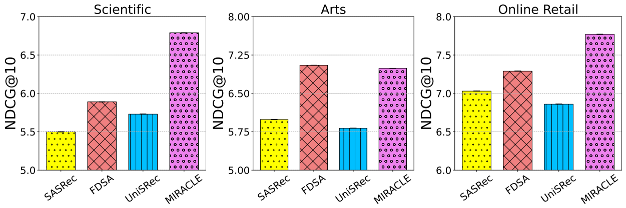

Figure 3: The experiment results under inductive setting with "%" omitted. SASRec and FDSA use the ID information.

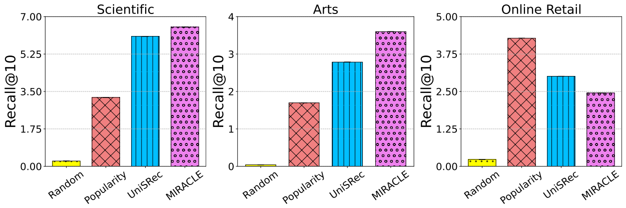

Figure 4: The experiment results under zero-shot setting on the validation set with "%" omitted.

4.3 The Effectiveness of Pre-training (RQ2)

We further analyze the pre-training effectiveness of MIRACLE under two evaluation settings: inductive recommendation and zero-shot recommendation. In inductive recommendation, the input representation of deeply-contextualized item embedding module is consistent with the pre-training, but we exclude item ID embeddings. And in zero-shot recommendation, we do not fine-tune MIRACLE in the downstream dataset and directly verify the pre-training model performance. The above two settings are verifying the effectiveness of pre-training from different perspectives.

4.3.1 *Inductive Recommendation.*

Considering that the model only uses text information in the pre-training stage, in the downstream datasets, we can use the inductive mode to solve the cold start problem of new items lacking id embedding. Therefore, when we fine-tune MIRACLE in the downstream datasets, we do not consider ID as auxiliary information, and only use text information in line with pre-training. Figure 3 shows the results under inductive settings, we choose SASRec, FDSA and UniSRec as the comparison methods. SASRec and FDSA are non-pretrained models that use id information, and UniSRec follows the inductive setting. From the results, we can find that despite the lack of ID information, MIRACLE can still significantly exceed SASRec and FDSA that utilize id information in the two datasets, and is equal to FDSA in another dataset. Compared with the transductive setting, which using ID information, the improvement of MIRACLE relative to UniSRec is more obvious, which proves that text information can effectively help multi-interest learning.

4.3.2 *Zero-Shot Recommendation.*

Considering the strong knowledge transfer ability in the pre-training model, we evaluate the zero-shot recommendation performance of MIRACLE. Figure 4 shows the results in Scientific, Arts, and Online Retail datasets. We choose three methods as the baseline methods: random recommendation, popularity-based recommendation, and UniSRec. It should be noted that the popularity-based is the simplest method that uses training

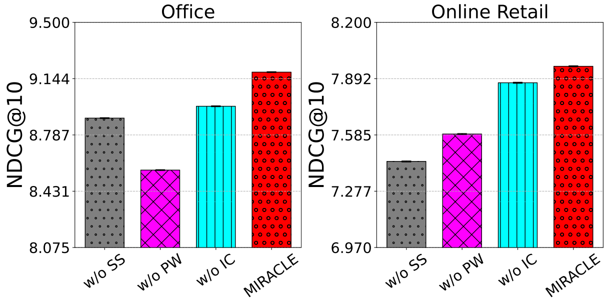

Figure 5: The ablation study of MIRACLE on Office and Online Retail dataset with "%" omitted

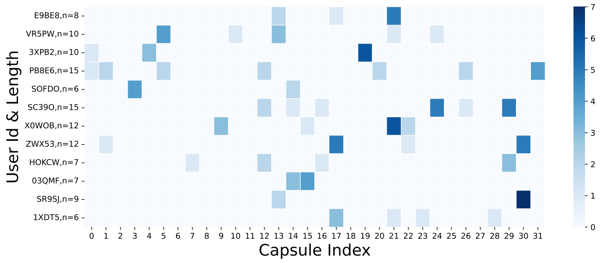

Figure 6: Visualization of Sparse Capsule Activation on Office Dataset (*Best viewed in color*)

data, which does not satisfy the zero-shot setting. Compared UniSRec with the other two methods, we found that the pre-training model can perform zero-shot recommendation, it even significantly outperforms popularity-based recommendation under the cross-domain datasets. And our methods greatly improves the performance of zero-shot recommendation on cross-domain datasets. When encountering the cross-platform dataset, due to the difference text distribution between the pre-training dataset and the downstream dataset, per-trained models perform worse than the popularity-based method, but it still has a certain ability to recommend. Recall in Table 2, by further performing fine-tuning over the in-domain data, we can achieve substantial performance improvement for cross-platform dataset.

4.4 Ablation Study (RQ3)

To verify the impact of different components (*i.e.,* interest contrastive learning, position-wise module) in our proposed MIRACLE, we conduct the ablation study in a cross-domain dataset Office and a cross-platform dataset Online Retail to analyze the impact of each component. We adopt NDCG@10 for evaluation. We prepare three variants of the proposed MIRACLE for comparisons, including **(1)** without position-wise Module (w/o PW) **(2)** without interest-level contrastive learning (w/o IC), **(3)** without sparse capsule activation module (w/o SS), and we set the number of interest capsule to 4. Figure 5 reports the experiment results of each variant. Here, it is obvious that MIRACLE obtains the best performance across different datasets, all the components are useful to improve the recommendation performance.

4.5 Hyper-parameter Sensitivity (RQ4)

We continue analyzing the model performance with respect to the parameter sensitivity. Specially, we investigate the number of transformer layer L in the sequence encoder and the number of total interest capsule C in the sparse interest capsule network. Table 3 and 4 illustrates the performance of MIRACLE when L and C take different values.

For the number of transformer layers, when $L = 1$ and $L = 2$, the performance of the model is not much different, but when $L = 3$, the performance of the model will decline. This is inconsistent with the general fact that the larger the pre-training model, the better the effect. The possible reason is that: **(1)** the amount of pre-training data is not large, an overly large model will overfit the pre-training data. **(2)** we freeze the parameters of the sequence encoder during the fine-tune stage, so larger models struggle to adapt to downstream datasets.

For the number of interest capsule C, we can observe that the model will achieve better results overall when C=32. As C becomes smaller, the performance of MIRACLE of the Online Retail and Office datasets will gradually decline, while C gets larger, the performance in the Scientific dataset will decline, the possible reason is that scientific dataset, too many capsules will lead to too fine-grained interests and reduce the performance.

4.6 Visualization

In Figure 6, we sample twelve users from the Office dataset to visualize the sparse capsule activation module. The x-axis represents the capsule index, the y-axis represents the last five digits of the user ID, and n means the sequence length. The darker the color in the heatmap, the more items activate that capsule. We can find that the activation of interest capsules is dynamic: sequences of different lengths may have the same number of interests, and sequences of the same length may also have different number of interests.

5 CONCLUSION

In this paper, we propose a novel multi-interest pre-training framework for sequential recommendation. We utilize MoE structure and bidirectional transformer to fuse cross-domain knowledge and model context information respectively. Moreover, we solve the restriction that the number of user's interest count should be predefined in the previous multi-interest learning methods and model the temporal information in the dynamic routing. Empirical results on several cross-domain datasets and one cross-platform dataset demonstrate that our model performs better than state-of-the-art baselines. Further, visualization of sparse capsule activation validates that our model can adaptively determine the number of user interests.

ACKNOWLEDGMENTS

This work was supported by National Natural Science Foundation of China (No. 62272349); Young Top-notch Talent Cultivation Program of Hubei Province; and CCF-AFSG Research Fund. Lixin Zou and Chenliang Li are the corresponding authors.

REFERENCES

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450* (2016).

[2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[3] Yukuo Cen, Jianwei Zhang, Xu Zou, Chang Zhou, Hongxia Yang, and Jie Tang. 2020. Controllable multi-interest framework for recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2942–2951.

[4] Zheng Chai, Zhihong Chen, Chenliang Li, Rong Xiao, Houyi Li, Jiawei Wu, Jingxu Chen, and Haihong Tang. 2022. User-Aware Multi-Interest Learning for Candidate Matching in Recommenders. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1326–1335.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[7] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939* (2015).

[8] Yupeng Hou, Shanlei Mu, Wayne Xin Zhao, Yaliang Li, Bolin Ding, and Ji-Rong Wen. 2022. Towards Universal Sequence Representation Learning for Recommender Systems. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 585–593.

[9] Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*. IEEE, 197–206.

[10] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

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

[12] Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the sentence embeddings from pre-trained language models. *arXiv preprint arXiv:2011.05864* (2020).

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

[14] Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*. 188–197.

[15] Zhaopeng Qiu, Xian Wu, Jingyue Gao, and Wei Fan. 2021. U-BERT: Pre-training user representations for improved recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 4320–4327.

[16] 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.

[17] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. 2017. Dynamic routing between capsules. *Advances in neural information processing systems* 30 (2017).

[18] J Ben Schafer, Dan Frankowski, Jon Herlocker, and Shilad Sen. 2007. Collaborative filtering recommender systems. In *The adaptive web*. Springer, 291–324.

[19] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538* (2017).

[20] 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.

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

[22] 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*. 565–573.

[23] Yu Tian, Jianxin Chang, Yanan Niu, Yang Song, and Chenliang Li. 2022. When Multi-Level Meets Multi-Interest: A Multi-Grained Neural Model for Sequential Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1632–1641.

[24] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[25] Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan. 2019. Session-based recommendation with graph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33. 346–353.

[26] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 974–983.

[27] 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 *IJCAI*. 4320–4326.

[28] 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*. 1893–1902.