删除标识 | 福昕编辑器

Transform Cold-Start Users into Warm via Fused Behaviors in Large-Scale Recommendation

Pengyang Li
pyli@zju.edu.cn
Zhejiang University

Rong Chen
xingshu.cr@alibaba-inc.com
Alibaba Group

Quan Liu*
lq204691@alibaba-inc.com
Alibaba Group

Jian Xu
xiyu.xj@alibaba-inc.com
Alibaba Group

Bo Zheng
bozheng@alibaba-inc.com
Alibaba Group

ABSTRACT

Recommendation for cold-start users who have very limited data is a canonical challenge in recommender systems. Existing deep recommender systems utilize user content features and behaviors to produce personalized recommendations, yet often face significant performance degradation on cold-start users compared to existing ones due to the following challenges: (1) Cold-start users may have a quite different distribution of features from existing users. (2) The few behaviors of cold-start users are hard to be exploited. In this paper, we propose a recommender system called *Cold-Transformer* to alleviate these problems. Specifically, we design context-based *Embedding Adaption* to offset the differences in feature distribution. It transforms the embedding of cold-start users into a warm state that is more like existing ones to represent corresponding user preferences. Furthermore, to exploit the few behaviors of cold-start users and characterize the user context, we propose to simultaneously model *Fused Behaviors* of positive and negative feedback with *Label Encoding*, which encodes more behavior information. Last, to perform large-scale industrial recommendations, we keep the two-tower architecture that de-couples user and target item. Extensive experiments on public and industrial datasets show that Cold-Transformer significantly outperforms state-of-the-art methods, including those that are deep coupled and less scalable.

CCS CONCEPTS

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

KEYWORDS

Recommender Systems; Cold-Start Users; User Behaviors

ACM Reference Format:
Pengyang Li, Rong Chen, Quan Liu, Jian Xu, and Bo Zheng. 2022. Transform Cold-Start Users into Warm via Fused Behaviors in Large-Scale Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22).* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3477495.3531797

*Corresponding author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
SIGIR '22, July 11–15, 2022, Madrid, Spain
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-8732-3/22/07…$15.00
https://doi.org/10.1145/3477495.3531797

1 INTRODUCTION

Most recommendation models' effectiveness is achieved through adopting a great number of user interactions. However, it is common for a small percentage of users to account for most data, while many cold-start users only have very limited data. These cold-start users, including newcomers and long-tail ones, are crucial to the platforms' ecosystem, yet their recommendation performance often faces a significant decrease, which has been widely known as the user cold-start problem.

Conventional deep recommendation models do not explicitly optimize for cold-start users and thus are dominated by users in training sets, i.e., existing users. The gap between cold-start users and existing users often leads to the model's unsatisfactory performance on cold-start users, which is non-trivial to resolve for the following reasons:

- **The distribution of features could be quite different for cold-start users and existing users.** As shown in [2], the embedding of cold-start users is distinctive, which results from the different distribution of features. For instance, when recommending for merchants, statistic features of marketing history play a critical role in recommendation quality. While for cold-start users, these features are significantly different from existing users, making deep models hard to generalize.
- **The few behaviors of cold-start users are hard to be exploited.** A direct method to utilize user behaviors is regarding the behavior sequence as a special feature like sequential models. However, when user behaviors are few, the positive feedback that most sequential models focus on will be extremely limited. Thus these models don't perform well.

A straightforward idea to address the user cold-start problem is generating a good user ID embedding since cold ID embedding is randomly initialized and significantly different from well-trained ID embedding. Previous methods utilize attributes [16], graphs [15] and a few behaviors [22] to initialize the cold ID embedding. Although a good ID embedding alleviates the problem of different distribution, other features still face the problem. Ignoring user and item ID features, MeLU [13] utilizes MAML [7] to learn a global parameter, based on which the personalized model parameters will be locally updated with the few behaviors of cold-start users. Some other methods [5, 14] share a similar idea to MeLU. Their meta-learning-based local updating alleviates the problem of over-fitting

on the few samples. However, it ignores the sequential information of interactions, which limits its capability to understand users comprehensively and needs fine-tuning in inference so that it is not easy to deploy.

To attack the aforementioned problems, we propose three key contributions summarized as follows: Firstly, we propose context-based *Embedding Adaption* to transform the embedding of cold-start and existing users into the same space, which offsets the differences of feature distribution and makes the cold-start users' embedding more in line with their preferences. Specifically, we characterize the context of users with behavior sequences, and the user embedding is then adapted by aggregating the corresponding contextual information with Transformer [18]. Secondly, we propose to exploit the few behaviors of cold-start users further to characterize their context. Since the volume of users' negative feedback is relatively sufficient while that of positive feedback is much more limited, we keep the users' positive and negative feedback in the sequence of *Fused Behavior*. This way, it maintains the integrity of user behaviors and alleviates the heterogeneity of different kinds of feedback through proposed *Label Encoding*. Besides, we propose a globally learned embedding as cold-start users' ID embedding to avoid the repercussion of randomly initialized ID embedding. In order to perform large-scale industrial recommendations, we adopt the two-tower architecture [11] to de-couple the user and target item. Extensive experiments on various datasets demonstrate that our proposed system has significantly improved the recommendation effect for cold-start users and eliminates the performance gap between cold-start and existing users.

2 METHODOLOGY

2.1 User Cold-Start in Recommendation

We model the user cold-start problem based on the binary classification recommendation tasks, e.g., click-through rate (CTR) prediction and user preference estimation. Following [19], we split users into two parts to simulate the real-world recommendation where cold-start users are coming daily and need to be recommended together with existing users. Concretely, users that have been seen in training are called *existing users*, and the unseen users are called *cold-start users*. **Note, cold-start users may accumulate a few behaviors before their interactions are absorbed for training the model.** Cold-start users are usually the main concern in user cold-start recommendation, while some existing users may also have insufficient interactions (e.g., long-tail users). Thus, we measure recommendation performance both on existing and cold-start users.

2.2 Adapting User Embedding

The main contribution of our method is that we de-couple the complete user embedding from deep-coupled networks and transform it into a context-aware warm state, which offsets the differences in features and makes the cold-start users' embedding more in line with their preferences. We argue that warming up only the ID embedding [15, 16, 22] is not optimal. In particular, they warm up the ID embedding for cold-start users or items and then feed them into a model trained on existing users. However, as indicated in [2], the distribution of features may be quite different between existing users and cold-start users. Hence, the model straightforwardly trained on existing users may have challenges generalizing to cold-start users.

To this end, we propose to adapt the embedding representing all the user features, denoted as Embedding Adaption. As shown in Figure 1, the proposed model is based on the two-tower architecture [11] to de-couple the user embedding from the deep networks. As a result, the user embedding can be explicitly adapted in user-specific dynamic contexts represented by fused behaviors. That behavior sequence is of both positive and negative user feedback and sorted according to timestamp, thus can accumulate more information over time. Intuitively, users with limited behaviors are in a cold state and can not be well learned. Their embedding is raw and thus needs further and real-time adaption. The context-aware Embedding Adaption enables the model to utilize the few behaviors to perform user-specific adaption, which is expected to learn to transform the raw user embedding into the context-aware warm state.

The sequence aggregation of Embedding Adaption is implemented with the Transformer [18] architecture since it naturally models the sequential information and can refine the heterogenous fused behaviors with self-attention. Given the embedding sequence of fused behaviors $[e_{v^{(1)}}, ..., e_{v^{(L)}}]$, we first inject the information about the position to make use of the sequential information by *cosine positional encoding* [18]. Then, the raw user embedding e_u and positional encoded $[e_{v^{(1)}}, ..., e_{v^{(L)}}]$ are concatenated as the initial hidden state $\mathbf{H}^{(0)}$, where e_u is not positional encoded since it is heterogeneous from the items sequence. After that, in the MultiHead *(multi-head self-attention)* mechanism of Transformer, the hidden state $\mathbf{H}^{(l)}$ at layer l is linearly projected into h subspaces, and next is applied to the *Scaled Dot-Product Attention* in parallel. Finally, these heads are concatenated and once again projected, resulting in the final values:

$$\text{MultiHead}(\mathbf{H}^{(l)}) = \text{Concat}(\text{head}_1, ..., \text{head}_\text{h})\mathbf{W}^O, \tag{1}$$

where

$$\text{head}_\text{i} = \text{Attn}(\mathbf{H}^{(l)}\mathbf{W}_i^Q, \mathbf{H}^{(l)}\mathbf{W}_i^K, \mathbf{H}^{(l)}\mathbf{W}_i^V). \tag{2}$$

The projections are learnable parameter matrices $\mathbf{W}_i^Q \in \mathbb{R}^{d\times d}$, $\mathbf{W}_i^K \in \mathbb{R}^{d\times d}$, $\mathbf{W}_i^V \in \mathbb{R}^{d\times d}$ and $\mathbf{W}^O \in \mathbb{R}^{hd\times d}$. The Scaled Dot-Product Attention is as follows:

$$\text{Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}}\right)\mathbf{V}, \tag{3}$$

where the temperature $\sqrt{d}$ is to scale the attention distribution for avoiding extremely small gradients [18]. The transformation of hidden state is formulated as follows:

$$\mathbf{H}^{(l+1)} = \text{LN}(\mathbf{H}^{(l)} + \text{Dropout}(\text{MultiHead}(\mathbf{H}^{(l)}))), \tag{4}$$

where LN is Layer Normalization [1] and Dropout [17] is a regularization technique.

In the last layer, since we only want to obtain the transformed context-aware user embedding, the multi-head self-attention is modified as follows:

$$\text{head}_\text{i}^{(\text{last})} = \text{Attn}(\mathbf{H}_0^{(l)}\mathbf{W}_i^Q, \mathbf{H}^{(l)}\mathbf{W}_i^K, \mathbf{H}^{(l)}\mathbf{W}_i^V), \tag{5}$$

where $\mathbf{H}_0^{(l)} \in \mathbb{R}^{1\times d}$ is the corresponding hidden state of raw user embedding.

删除标识 | 福昕编辑器

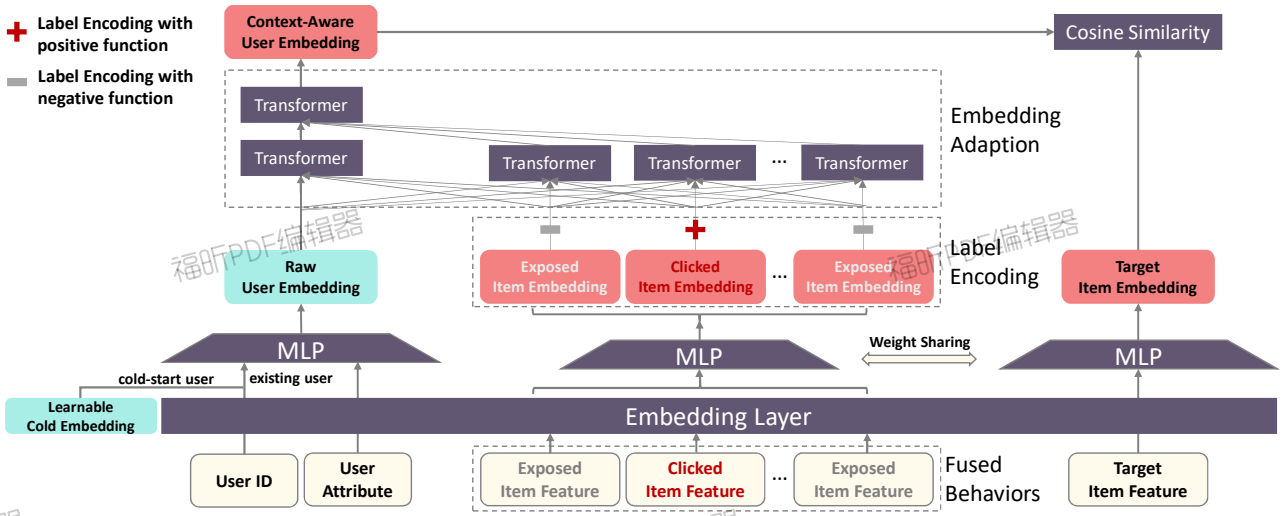

Figure 1: Illustration of our proposed method based on the two-tower framework. Raw user embedding is first extracted from non-sequential features. Then, the embedding adaption layer aggregates the contextual information on raw user embedding and outputs the adapted context-aware user embedding. Finally, the adapted user is scored for a given item to predict CTR.

2.3 Exploiting Fused User Behaviors

A key challenge in the user cold-start problem is that users only have limited behaviors, among which the volume of users' positive feedback is even more scarce. For example, on e-commerce platforms, the number of clicked items of cold-start users can be very few and thus does not characterize user context well. Fortunately, the exposure sequence (i.e., negative feedback) has a relatively larger volume yet is often overlooked. To sufficiently exploit the short history of user interactions, we argue that exploiting the fused behavior sequence of exposure and click is beneficial to representing user context.

As discussed in EdgeRec [8], most of previous sequence modeling works [4, 21] only consider user's positive feedback. EdgeRec realizes the significance of both positive (i.e., clicked items) and negative (i.e., exposed items) feedback, and thus proposed to model exposure sequence and click sequence separately because of the heterogeneity of these two kinds of sequences. Nevertheless, it ignores the sequential relationship between the clicked items and exposed items. For instance, if the user clicks A when exposed to (A, B), it may imply that the user prefers A to B instead of disliking B. Therefore, we propose simultaneously modeling the exposure and click sequence as one fused sequence.

Moreover, consider e_v^* as the historical interacted item v's optimal embedding that dismisses the heterogeneity of different kinds of interactions and embeds the sequential relationship. The vanilla embedding e_v of item v has a large gap with e_v^*. As a result, we approximate e_v^* in two steps.

In the first step, we propose *Label Encoding* to alleviate the problem of heterogeneity. We estimate $\hat{e}_v$ that reduce the gap of heterogeneity for different kinds of interactions. Inspired by [10], we use deep residual learning to encode residual vectors r_v^*, which is more effective than encoding original vectors. It can be formulated as $r_v^* = \hat{e}_v - e_v$, and is approximated by linear mapping $r_v^* \approx r(e_v) = \mathbf{W}^r e_v$ in this paper, where $\mathbf{W}^r \in \mathbb{R}^{d \times d}$ is the parameter matrix and d is the dimension of embedding. Given the embedding sequence of users' interacted items $[e_{v^{(1)}}, ..., e_{v^{(L)}}]$ and corresponding feedback categories $[y^{(1)}, ..., y^{(L)}]$, we estimate the i-th item's embedding $\hat{e}_{v^{(i)}}$ that dismisses the heterogeneity of different user feedback as:

$$\hat{e}_{v^{(i)}} = e_{v^{(i)}} + y^{(i)} r_{pos}(e_{v^{(i)}}) + (1 - y^{(i)}) r_{neg}(e_{v^{(i)}}). \quad (6)$$

Here, to better approximate r_v^* for positive and negative interactions, r_{pos} and r_{neg} are learned separately.

In the second step, we embed the sequential relationship between the clicked items and exposed items. As shown in Figure 1, it is integrated into Embedding Adaption. The positional encoding and self-attention mechanism enable the embedding of each interacted item to be adapted according to related interactions.

2.4 Denoising with Learnable Cold Embedding

For cold-start users that are unseen in the training stage, the ID embedding is usually randomly initialized. We propose a learnable cold embedding that replaces the random ID embedding to eliminate the randomness. By randomly replacing ID embedding with cold embedding in the training stage, we can globally learn the common features of users without well-trained ID embedding. It helps align the distribution of ID embedding between existing users and cold-start users on the inference stage, which also facilitates the Embedding Adaptation for cold-start users.

3 EXPERIMENTS

3.1 Dataset

In this paper, we use two popular and challenging public datasets (MovieLens-1M and Taobao Display Ad) to verify the proposed method. Moreover, we also conduct experiments to evaluate the cold-start performance on an industrial exposure/click dataset with

Methods	MovieLens-1M Existing AUC	MovieLens-1M Existing RelaImpr	MovieLens-1M Cold-Start AUC	MovieLens-1M Cold-Start RelaImpr	Taobao Display AD Existing AUC	Taobao Display AD Existing RelaImpr	Taobao Display AD Cold-Start AUC	Taobao Display AD Cold-Start RelaImpr	Industrial Dataset Existing AUC	Industrial Dataset Existing RelaImpr	Industrial Dataset Cold-Start AUC	Industrial Dataset Cold-Start RelaImpr
BaseModel^m[11]	0.7643	0.00%	0.7228	0.00%	0.6178	0.00%	0.5787	0.00%	0.7688	0.00%	0.7665	0.00%
Wide&Deep^m[3]	0.7616	-1.02%	0.7246	0.81%	0.6253	6.37%	0.5923	17.28%	0.7675	-0.48%	0.7642	-0.86%
DeepFM^m[9]	0.7660	0.64%	0.7384	7.00%	0.6425	20.97%	0.5915	16.26%	0.7701	0.48%	0.7669	0.15%
DIN^p[21]	0.7671	1.06%	0.7433	9.20%	0.6223	3.82%	0.5963	22.36%	0.7737	1.82%	0.7748	3.11%
DIEN^{p+n}[20]	0.7703	2.27%	0.7476	11.13%	0.6235	4.84%	0.5989	25.67%	0.7721	1.23%	0.7789	4.65%
EdgeRec^{p+n}[8]	0.7793	5.68%	0.7539	13.96%	0.6305	10.78%	0.6042	32.40%	0.7812	4.61%	0.7795	4.88%
DropoutNet^{p*}[19]	0.7624	-0.72%	0.7279	2.30%	0.6142	-3.06%	0.5861	9.40%	0.7693	0.19%	0.7682	0.64%
MWUF^{p*}[22]	0.7643	0.00%	0.7334	4.76%	0.6178	6.28%	0.5974	23.76%	0.7688	0.00%	0.7712	1.76%
MAML^{m*}[7]	0.7693	1.89%	0.7395	7.50%	0.6252	3.40%	0.5791	0.51%	0.7742	2.01%	0.7691	0.98%
Ours^{p+n}	**0.7998**	**13.43%**	**0.7775**	**24.55%**	**0.6477**	**25.38%**	**0.6422**	**80.69%**	**0.7816**	**4.76%**	**0.7951**	**10.73%**

Table 1: CTR prediction performance for both existing and cold-start users. User behaviors are generated according to the timestamp. '^{m}': We use the mean of clicked items' embedding as a user feature for the method. '^{p}': The method models the positive feedback. '^{n}': The method models the negative feedback. '*': The method is re-implemented based on BaseModel.

5M records from the Alibaba e-commerce platform. We split each dataset into training and testing sets based on a specific timestamp to simulate real-world recommendation scenarios. Samples before this timestamp are regarded as the training set, and the rest are regarded as the testing set. The testing set is further divided into existing and cold-start users sets, where the users who are unseen in the training set will be regarded as cold-start users. Cold-start users usually have only a few behaviors (both pos and neg feedback). For instance, 94.20% and 77.98% cold-start users in Taobao and Industrial datasets have no larger than 10 interactions, respectively. Yet cold-start users in MovieLens have much more behaviors because of MovieLens' low sparsity (95.53%). As a result, we limit the length of behaviors of cold-start users to 10 in evaluating for MovieLens. In the real-world scenario, the model needs to predict both existing users and cold-start users. Hence, we conduct experiments on both existing and cold-start users to evaluate the comprehensive performance of the models. For all datasets, we generate the user behaviors for each sample according to the users' interactions before the corresponding timestamp.

3.2 Experimental Setup

Evaluation metrics. To evaluate the performance of binary classification tasks (e.g., recommendation and advertising), AUC is a widely used metric [6]. It measures the goodness of order by ranking all the items with the predicted score. Following the cold-start works [16, 22], we adapt AUC as the main metric in our experiments. Besides, as [21, 22], we use the RelaImpr metric to measure the relative improvement over different methods.

Implementation details. We optimize the model using Adam [12] with a learning rate of 0.001. For a fair comparison, we use the same size of embedding layer and the same MLP in all the deep models compared in Section 3. Specifically, the dimensionality of the embedding layer is set to 32, and the MLP contains two hidden layers with 64 units. The mini-batch size for MovieLens and the industrial dataset is set to 200, and 2000 for the Taobao Display Ad dataset. The proposed Embedding Adaption layer is implemented with 2 layers of Transformer with 2 heads, and the dropout rate is set to 0.5. The max length of the behavior sequence is limited to 50.

Methods	De-coupled Model	Existing AUC	Existing RelaImpr	Cold-Start AUC	Cold-Start RelaImpr
FE w/ p		0.7625	0.00%	0.7250	0.00%
FE w/ p//n		0.7793	6.40%	0.7539	12.84%
FE w/ p&n		0.7936	11.85%	0.7702	20.09%
EA w/ p	√	0.7566	-2.25%	0.7164	-3.82%
EA w/ p&n	√	**0.7998**	**14.21%**	**0.7775**	**23.33%**

Table 2: Comparison of different sequence modeling methods on MovieLens based on the same sequence aggregation layer. 'p//n': Separated positive and negative feedback. 'p&n': Fused positive and negative feedback.

3.3 Comparison with State-of-the-Arts

We conduct experiments on both public and industrial datasets. Table 1 reports results for existing and cold-start users. Here we denote the two-tower model DSSM [11] as *BaseModel* since our model is based on it. For methods that do not explicitly model user behaviors, we use the simple but effective sequence utilizing approach, i.e., the mean of clicked items' embedding. From the experimental results, we highlight the following observations:

The effectiveness for cold-start users. Firstly, we note that our model performs best in AUC for cold-start users among all the competitors, although these methods are competitive. DIN, DIEN, and EdgeRec are popular sequence modeling methods. Their personalized interest extraction utilizes behaviors to model users, but when the number of behaviors is limited for cold-start users, they can't perform as well as ours. The cold-start methods DropoutNet and MWUF also utilize behavior sequences and improve the prediction performance for cold-start users. However, they only generate a good user ID embedding with behaviors instead of adapting the whole user embedding. MAML fine-tunes for cold-start users during

inference with behaviors, which ignores the sequential information of behaviors. As a result, our model outperforms these SOTAs and almost eliminates the gap of prediction performance between existing and cold-start users.

The effectiveness for existing users. We can find that our model consistently outperforms all the competitors on existing users as well. It is because some existing users also have limited behaviors (e.g., long-tail users) and thus can be further optimized. For instance, about half of the existing users in the industrial dataset have few behaviors (less than 10) during training and thus have similar performance as cold-start users. Besides, about 10% and 30% of the existing users have less than 10 behaviors in MovieLens and TaobaoAd, respectively. Therefore, our cold-start method also improves existing users' prediction performance, although the improvement is not as obvious as on cold-start users.

3.4 Ablation Study

To study the effects of Embedding Adaption and different sequence modeling methods, we conduct a series of experiments on MovieLens-1M. We propose to utilize the fused sequence of users' positive and negative behaviors with Label Encoding to adapt user embedding (denoted as Embedding Adaption, **EA**), which is different from typical methods that regard the behavior sequence as a feature and extract one or several embedding to represent this feature (denoted as Feature Extraction, **FE**). Besides, typical methods like DIN [21] only focus on positive feedback to exploit the users' behavior sequence, and some methods like EdgeRec [8] realize the significance of both positive and negative feedback and model them separately. They also need to be fairly compared with our sequence modeling method. Hence, as shown in Table 2, we conduct a performance comparison between our sequence modeling method and other methods based on the same sequence aggregation layer, i.e., Transformer. We can find that utilizing the Fused Behaviors with Label Encoding generally and significantly improves the comprehensive performance, and Embedding Adaption further improves the cold-start users' performance. Since the distinct positive feedback can not fully represent users' context when positive feedback is extremely scarce, it is reasonable that Embedding Adaption obtains lower performance with the positive feedback only.

4 CONCLUSION

In this paper, we delve into the realistic problem, *cold-start recommendation* for users, which affects both consumers and merchants on many e-commerce platforms. We propose an effective method called Cold-Transformer that can exploit limited behaviors via adapting user embedding through fused behavior sequences. It transforms user embedding into a warm state, which is closer to the embedding of preferred items. Extensive experimental results on diverse datasets demonstrate the effectiveness of Cold-Transformer. For future work, we'd like to further extend our work to cold-start users without any behavior (i.e., zero-shot).

REFERENCES

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

[2] Zhihong Chen, Rong Xiao, Chenliang Li, Gangfeng Ye, Haochuan Sun, and Hongbo Deng. 2020. Esam: Discriminative domain adaptation with non-displayed items to improve long-tail performance. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 579–588.

[3] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, Rohan Anil, Zakaria Haque, Lichan Hong, Vihan Jain, Xiaobing Liu, and Hemal Shah. 2016. Wide & Deep Learning for Recommender Systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*. 7–10.

[4] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM conference on recommender systems*. 191–198.

[5] Manqing Dong, Feng Yuan, Lina Yao, Xiwei Xu, and Liming Zhu. 2020. Mamo: Memory-augmented meta-optimization for cold-start recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 688–697.

[6] Tom Fawcett. 2006. An introduction to ROC analysis. *Pattern recognition letters* 27, 8 (2006), 861–874.

[7] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*. PMLR, 1126–1135.

[8] Yu Gong, Ziwen Jiang, Yufei Feng, Binbin Hu, Kaiqi Zhao, Qingwen Liu, and Wenwu Ou. 2020. EdgeRec: Recommender System on Edge in Mobile Taobao. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2477–2484.

[9] Huifeng Guo, Ruiming TANG, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: A Factorization-Machine based Neural Network for CTR Prediction. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*. 1725–1731.

[10] 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.

[11] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning Deep Structured Semantic Models for Web Search Using Clickthrough Data. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. 2333–2338.

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

[13] Hoyeop Lee, Jinbae Im, Seongwon Jang, Hyunsouk Cho, and Sehee Chung. 2019. MeLU: Meta-Learned User Preference Estimator for Cold-Start Recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1073–1082.

[14] Yuanfu Lu, Yuan Fang, and Chuan Shi. 2020. Meta-learning on heterogeneous information networks for cold-start recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1563–1573.

[15] Wentao Ouyang, Xiuwu Zhang, Shukui Ren, Li Li, Kun Zhang, Jinmei Luo, Zhaojie Liu, and Yanlong Du. 2021. Learning Graph Meta Embeddings for Cold-Start Ads in Click-Through Rate Prediction. *arXiv preprint arXiv:2105.08909* (2021).

[16] Feiyang Pan, Shuokai Li, Xiang Ao, Pingzhong Tang, and Qing He. 2019. Warm Up Cold-Start Advertisements: Improving CTR Predictions via Learning to Learn ID Embeddings. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 695–704.

[17] Nitish Srivastava, Geoffrey 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* (2014), 1929–1958.

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

[19] Maksims Volkovs, Guang Wei Yu, and Tomi Poutanen. 2017. DropoutNet: Addressing Cold Start in Recommender Systems.. In *NIPS*. 4957–4966.

[20] 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 AAAI Conference on Artificial Intelligence*, Vol. 33. 5941–5948.

[21] Guorui Zhou, Xiaoqiang Zhu, Chenru 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.

[22] Yongchun Zhu, Ruobing Xie, Fuzhen Zhuang, Kaikai Ge, Ying Sun, Xu Zhang, Leyu Lin, and Juan Cao. 2021. Learning to Warm Up Cold Item Embeddings for Cold-start Recommendation with Meta Scaling and Shifting Networks. *arXiv preprint arXiv:2105.04790* (2021).