Leveraging Historical Interaction Data for Improving Conversational Recommender System

Kun Zhou[1], Wayne Xin Zhao[2,3*], Hui Wang[1], Sirui Wang[4], Fuzheng Zhang[4], Zhongyuan Wang[4] and Ji-Rong Wen[2,3]
[1]School of Information, Renmin University of China
[2]Gaoling School of Artificial Intelligence, Renmin University of China
[3]Beijing Key Laboratory of Big Data Management and Analysis Methods
[4]Meituan-Dianping Group
francis_kun_zhou@163.com, {batmanfly, hui.wang, jrwen}@ruc.edu.cn,
wangsirui@meituan.com, zhfzhkris@outlook.com

ABSTRACT

Recently, conversational recommender system (CRS) has become an emerging and practical research topic. Most of the existing CRS methods focus on learning effective preference representations for users from conversation data alone. While, we take a new perspective to leverage historical interaction data for improving CRS. For this purpose, we propose a novel pre-training approach to integrating both item-based preference sequence (from historical interaction data) and attribute-based preference sequence (from conversation data) via pre-training methods. We carefully design two pre-training tasks to enhance information fusion between item- and attribute-based preference. To improve the learning performance, we further develop an effective negative sample generator which can produce high-quality negative samples. Experiment results on two real-world datasets have demonstrated the effectiveness of our approach for improving CRS.

CCS CONCEPTS

• **Information systems** → **Personalization**; **Recommender systems**.

KEYWORDS

Conversational Recommender System, Pre-training Approach

ACM Reference Format:
Kun Zhou[1], Wayne Xin Zhao[2,3*], Hui Wang[1], Sirui Wang[4], Fuzheng Zhang[4], and Zhongyuan Wang[4] and Ji-Rong Wen[2,3]. 2020. Leveraging Historical Interaction Data for Improving Conversational Recommender System. In *The 29th ACM International Conference on Information and Knowledge Management (CIKM '20), October 19–23, 2020, Virtual Event, Ireland.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3340531.3412098

*Corresponding author.

CIKM '20, October 19–23, 2020, Virtual Event, Ireland

ACM ISBN 978-1-4503-6859-9/20/10…$15.00
https://doi.org/10.1145/3340531.3412098

1 INTRODUCTION

With the rapid development of intelligent agents in e-commerce platforms, conversational recommender system (CRS) [5, 6, 9] has become an emerging research topic in seeking to provide high-quality recommendations to users through conversations. Generally, a CRS consists of a conversation module and a recommendation module. The conversation module focuses on acquiring users' preference via multi-turn interaction, and the recommendation module focuses on how to utilize the inferred preference information to recommend suitable items for users.

Most of the existing CRSs are designed in a "*system asks-user responds*" mode [5, 6]. At each round, CRS issues a query about user preference and the user replies to the system with personalized feedback. Typically, a system query is generated according to some attributes of items (*e.g., what is your favorite movie genre*), and the user feedback reflects the specific preference of a user on that attribute (*e.g., I like Action movies*). A mainstream approach [5, 6] is to construct a belief tracker module that infers the attribute-based preference of a user from such a multi-turn conversation. In this way, the inferred preference can be presented as a sequence of inferred attributes (*e.g.,* "Genre=*Action* → Director=*James Cameron*" in a movie CRS). Furthermore, Factorization Machine [5, 6] or KG-based models [9] can be applied to construct the user preference representation and make the final recommendation.

However, these existing studies for CRS suffer from two major issues. First, the information of a conversation itself is quite limited. Many CRSs are further optimized to reduce the number of rounds that the system interacts with users [5, 6]. Thus, some useful attributes are likely to be missed in the inferred attribute-based preference. Second, it may be not sufficient to utilize attribute-based preference alone for making the recommendation. For example, the candidate item set can be still large even after the filter of several attributes.

To address the two issues, we observe that a CRS is usually deployed within an application platform. When an existing user from the platform enters into its deployed CRS, we can obtain his/her historical interaction data, *i.e.,* a chronologically-ordered item sequence of the user. Intuitively, historical interaction data provides another kind of useful data signal to infer user preference. However, it is not easy to integrate the two kinds of information for CRS, which are different in essence. Indeed, historical interaction reflects item-level user preference in long run, while conversation

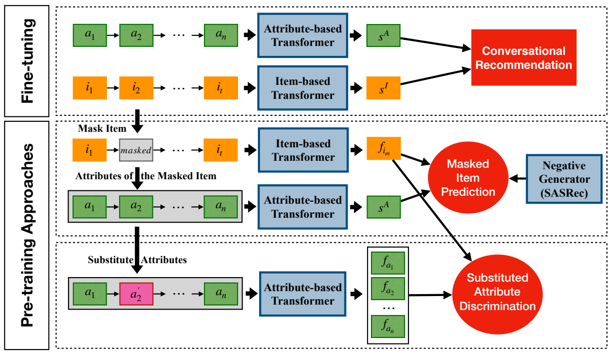

Figure 1: The overview of our approach. During pre-training, we incorporate MIP and SAD tasks to optimize the parameters. During fine-tuning, we optimize our pre-trained model by conversational recommendation task.

data reflects attribute-level user preference at present. Therefore, it is important to develop an effective model that can fuse item-level and attribute-level user preference for CRS.

Inspired by the success of pre-training methods like BERT [2], we propose a novel pre-training approach to leveraging historical interaction data of users for improving conversational recommendation. Our main idea is to integrate both item-based preference sequence (from historical interaction data) and attribute-based preference sequence (from conversation data) via pre-training methods. To model the two kinds of preference sequences, we develop an approach based on a self-attentive architecture [4], containing an item-based Transformer and an attribute-based Transformer. Specially, we design two auxiliary tasks for enhancing the data fusion, namely *Masked Item Prediction (MIP)* and *Substituted Attribute Discrimination (SAD)*. The MIP strategy adapts the idea of the Masked Language Model (MLM) in BERT [2] to conversational recommendation; the SAD strategy emphasizes the capacity of discriminating between actual or substituted attributes based on contextualized item representations. To improve the pre-training performance, we further propose to generate high-quality negative samples with a pre-training model by following IR-GAN [8] and ELECTRA [1].

To our knowledge, it is the first time that historical interaction data has been integrated and utilized in a setting of CRS by a pre-training approach. We believe such an idea is promising to improve existing CRS methods. To demonstrate the effectiveness of our approach, we construct experiments on real-world datasets that are tailored to the CRS task. Experiments show that our approach is more effective than a number of competitive CRS methods.

2 PROBLEM STATEMENT

When a user u enters into a CRS, a multi-round conversation will be initiated for making an accurate recommendation in a "*system asks-user responds*" mode. As discussed before, most of existing CRSs [5, 6] focus on learning attribute-level user preference for user preference. Typically, the queries from CRS are about the user preference over the possible item attributes, which is denoted by an attribute set $\mathcal{A}$.

As the conversation goes on, the system becomes increasingly clear about user preference, since it has acquired an *attribute-based preference sequence* from the target user, denoted by $\mathcal{P}^{(A)} = a_1 \rightarrow \cdots a_j \cdots \rightarrow a_n$, where n is the sequence length and a_j belongs to the attribute set $\mathcal{A}$. For a movie CRS, an example for the obtained $\mathcal{P}^{(A)}$ can be given as: GENRE=*Action* $\longrightarrow$ DIRECTOR=*James Cameron*.

Besides, we assume that the interaction history of user u is also available in our setting. It is given as an *item-based preference sequence*, denoted by $\mathcal{P}^{(I)} = i_1 \rightarrow \cdots i_k \cdots \rightarrow i_m$, where each $i_k \in \mathcal{I}$ is an item that user u has interacted with at the kth step before the conversation, and $\mathcal{I}$ is the item set. We further assume that each item i_k is also associated with a set of attribute values, denoted by $\mathcal{A}_{i_k}$, which is a subset of the entire attribute set $\mathcal{A}$.

Based on these notations, the task in our paper is defined as: given both attribute- and item-based preference sequences $\mathcal{P}^{(A)}$ and $\mathcal{P}^{(I)}$, we aim to accurately predict the item i^* that well matches the needs of user u in a conversation.

Note that our task setting is slightly different from existing CRS studies [5, 6]. Here, our focus is not to ask good questions or form suitable text-based responses. We aim to improve the recommendation task using historical interaction data after already acquiring some attribute preference information from the user. Therefore, we assume that the attribute-based preference $\mathcal{P}^{(A)}$ can be obtained through existing methods (*i.e.,* belief tracker [5, 6]). Indeed, such a task setting can be considered as a sub-task or sub-module for existing CRSs, *i.e.,* how to effectively generate the item recommendations based on *interaction history* and *conversation data*. In practice, CRS can run such a module for our task multiple times once the acquired attribute preference is updated.

3 APPROACH

In this section, we present the proposed approach to the item recommendation task in CRS, which is inspired by the recently proposed pre-trained language models [2]. Our approach contains two major stages, namely the pre-training and fine-tuning stages. We aim to learn effective representations through information fusion in the pre-training stage. Then we fine-tune the model according to the CRS task. The major contribution of our work lies in the pre-training stage, where we carefully design several auxiliary pre-training tasks to fuse item- and attribute-based preference. We present an overview of our approach in Fig. 1.

3.1 Base Model

We adopt Transformer [7] as our base model, which consists of embedding layer, self-attention layer, and prediction layer.

In the embedding layer, we maintain an item embedding matrix $\mathbf{M}_I \in \mathbb{R}^{|\mathcal{I}| \times d}$ and an attribute embedding matrix $\mathbf{M}_A \in \mathbb{R}^{|\mathcal{A}| \times d}$. The two matrices project the high-dimensional one-hot representation of an item or attribute to low-dimensional dense representation. Furthermore, we incorporate a learnable position encoding matrix $\mathbf{P} \in \mathbb{R}^{n \times d}$ to enhance the input representations of item sequence.

Based on the embedding layer, we build item and attribute encoders by stacking multiple self-attention blocks, and the two encoders are implemented with the same architecture but with different parameters. A self-attention block generally consists of two sub-layers: a multi-head self-attention layer and a point-wise feed-forward network. More details can be found in [7].

In the prediction layer of our model, we compute the preference score of user u for each candidate item i based on the interaction history $\mathcal{P}^{(I)}$ and conversation data $\mathcal{P}^{(A)}$ as:

$$P(i|u, \mathcal{P}^{(A)}, \mathcal{P}^{(I)}) \propto [\boldsymbol{s^I}; \boldsymbol{s^A}]^\top \mathbf{W}_M \boldsymbol{e}_i, \tag{1}$$

where $\boldsymbol{e}_i$ is the representation of item i from item embedding matrix $\mathbf{M}_I$, $\boldsymbol{s^I}$ and $\boldsymbol{s^A}$ are the learned state representations of the last position from the item and attribute encoders respectively, and $\mathbf{W}_M \in \mathbb{R}^{2d\times d}$ is the trainable parameter matrix.

3.2 Improvement with Pre-training Strategies

Based on the above model architecture, we propose two pre-training tasks, the *Masked Item Prediction* task (MIP) and the *Substituted Attribute Discrimination* task (SAD), to enhance the data representations via effective fusion of item- and attribute-based preference.

3.2.1 Masked Item Prediction. Inspired by BERT [2], we construct a *Cloze* task based on the item sequence. Given an item sequence $C = i_1 \rightarrow \cdots i_k \cdots \rightarrow i_m$, we randomly mask a proportion of items in the sequence, *e.g.,* replacing with special token "[*mask*]". For each masked item, we predict its original ID based on its contextual information, consisting of contextual items $C_{\neg i_k}$ and attributes $\mathcal{A}_{i_k}$. Following BERT [2], we leverage the bidirectional contextual information in item sequence for predicting the masked item as:

$$P(i_k|C_{\neg i_k}, \mathcal{A}_{i_k}) = \sigma\big([\boldsymbol{f_{i_k}}; \boldsymbol{s^A}]^\top \mathbf{W}_M \boldsymbol{e}_{i_k}\big) \tag{2}$$

where $\boldsymbol{f}_{i_k}$ is the representations for $C_{\neg i_k}$, which is the representation for the k-th position using the bidirectional Transformer. $\boldsymbol{s^A}$ is the representation of $\mathcal{A}_{i_k}$ as in Eq. 1, and $\sigma(.)$ is the sigmoid function to obtain the probability. We adopt the pairwise ranking loss with negative sampling to learn this pre-training task.

3.2.2 Substituted Attribute Discrimination. Furthermore, we design another task that enhances the fusion between item- and attribute-level information. In an item sequence, let $\mathcal{A}_{i_k}$ denote the associated attributes for the item i_k. Following ELECTRA [1] which replaces word in sentence by another irrelevant word, we randomly substitute some of its attributes in $\mathcal{A}_{i_k}$ with negative attributes that are randomly sampled, and obtain a corrupted $\tilde{\mathcal{A}}_{i_k}$. In this way, our second *Cloze* task to predict if the attribute a_j in $\tilde{\mathcal{A}}_{i_k}$ has been replaced or real:

$$P(y_{a_j} = 1|C, \tilde{\mathcal{A}}_{i_k}) = \sigma\big(f_{i_k}^\top \mathbf{W}_P f_{a_j}\big), \tag{3}$$

where y_{a_j} is a discrimination label for a_j, $\boldsymbol{f}_{i_k}$ and $\boldsymbol{f}_{a_j}$ are the representations of item i_k and attribute a_j through our architecture, respectively, and $W_P \in \mathcal{R}^{d\times d}$ is the trainable parameter matrix. We adopt the cross-entropy loss to learn this pre-training task.

3.3 Learning with Enhanced Negative Samples

A key procedure to learn with the MIP task is to sample irrelevant items as *negative samples.* In recommender systems [3, 4, 6], the most commonly adopted way is to randomly sample negative items. However, it has been widely recognized that the quality of negative samples directly affects the models [1, 5, 8]: too easy or too difficult negative samples are likely to lead to worse performance.

Table 1: Statistics of the datasets after preprocessing.

Datasets	#Users	#Items	#Interactions	#Attributes
Meituan	14,290	30,839	727,954	331
LastFM	2,100	1,921	39,828	71

Inspired by IR-GAN [8] and ELECTRA [1], we propose a new strategy to derive negative samples for the masked item prediction task, in which we set up a special module for generating high-quality negative samples. Since our setting is in a sequential manner, we utilize the state-of-the-art sequential recommendation model (*i.e.,* SASRec [4]) as a negative sample generator. We first pre-train the generator with traditional pairwise ranking loss using random sampling. Then, we sample negative samples according to the probability distribution that it assigns to each candidate item. Since SASRec has achieved very good performance in sequential recommendation, the top-ranked items with high probabilities are likely to be "*close-to-real*" ones, which is helpful to improve the learning of our approach. Note that although we can update the generator as that in standard GAN, we train its parameters only once. We have empirically found that the improvement with iterative updating is limited in our task.

Finally, the entire training procedure of our approach consists of the pre-training and fine-tuning stages. In the pre-training stage, we apply the two proposed pre-training strategies (with the improved negative sampling method) to enhance the learning of the data representations. In the fine-tuning stage, we adopt the pairwise rank loss to re-optimize the parameters according to our task:

$$L = \sum_{c\in\mathcal{D}} \log \sigma\big(P(i^+|u, c) - P(i^-|u, c)\big), \tag{4}$$

where c is a conversation involving user u from training data $\mathcal{D}$, and i^+ and i^- are the actual or negative items in this conversation.

4 EXPERIMENT

4.1 Experimental Setup

4.1.1 Datasets and Setup. We conduct experiments on two datasets: Meituan and LastFM. Meituan dataset contains 6-year (2014-2020) shopping transactions in Beijing on the food subcategory of the Meituan platform[1]. LastFM dataset is adopted by EAR [5], which is a music artist recommendation dataset and shared in HetRec 2011 tracks[2]. We rebuild these datasets for our task following [5]. The statistics of the two datasets are summarized in Table 1.

4.1.2 Evaluation metrics. To evaluate the CRS models, we adopted the *leave-one-out* evaluation [3, 4]. Each user has a sequence of conversational recommendation records, we hold out the last record in user history as the test data, treat the record just before the last as the validation set, and utilize the remaining data for training. Besides, each conversational recommendation record contains a targeted item. We pair the targeted item in the test set with 100 randomly sampled negative items. Hence, the task becomes to rank these items for each user. We adopt NDCG@10 and MRR to evaluate the performance of the ranking list for all the models.

[1] https://www.meituan.com
[2] https://grouplens.org/datasets/hetrec-2011/

Table 2: Performance comparisons of different methods on this task. The number marked with "*" indicates the improvement is statistically significant compared with the best baseline (t-test with p-value < 0.05).

Datasets	Meituan		LastFM	
Models	MRR	NDCG@10	MRR	NDCG@10
CRM	0.0942	0.1009	0.0567	0.0547
EAR	0.0838	0.0869	0.0483	0.0512
GRU_I	0.0964	0.1036	0.0692	0.0734
SASRec_I	0.1001	0.1083	0.0783	0.0778
GRU_{I+A}	0.0825	0.0847	0.1034	0.1170
SASRec_{I+A}	0.1327	0.1496	0.1231	0.1295
Ours	**0.2026***	**0.2354***	**0.1800***	**0.2032***
-w/o MIP	0.1495	0.1722	0.0828	0.0841
-w/o SAD	0.0627	0.0604	0.1091	0.1216
-w/o NG	0.1760	0.2022	0.1708	0.1926

4.1.3 Models. We consider the following baselines for comparisons: (1) CRM [6] utilizes factorization machines to learn the interaction between user, item, and attribute; (2) EAR [5] utilizes linear interactions among user, item, and attribute; (3) GRU_I [3] adopts Gated Recurrent Units to model user history; (4) SASRec_I [4] adopts self-attentive model to leverage user history; (5) GRU_{I+A} uses a GRU module to model user history, and a self-attention module to model the conversation data, in which the two types of information are fused by a linear layer; (6) SASRec_{I+A} adopts two self-attentive modules to model user history and conversation data, respectively, in which the two types of information are fused by a linear layer.

Among all the above methods, CRM and EAR are CRS models, which do not model the sequential pattern in user history. Then, GRU_I and SASRec_I only utilize the sequence information in user history, while GRU_{I+A} and SASRec_{I+A} utilize both user history and conversation data. Our approach adopts MIP and SAD tasks to pre-train the parameters in our model, in which we further improve MIP task by a negative sample generator (NG). The details of our model and dataset are available at this link: https://github.com/RUCAIBox/Pre-CRS. To evaluate the effectiveness of the two pre-training tasks and NG, we conduct an ablation study by removing one component from our approach at each time.

4.2 Results and Analysis

Table 2 presents the performance of different methods on the conversational recommendation task. First, we can observe that CRM and EAR do not perform very well in our task setting, since they do not utilize the user history effectively. Second, GRU_{I+A} and SASRec_{I+A} perform better than GRU_I and SASRec_I, which indicates that it is useful to leverage the user history and conversation data jointly. However, the result of GRU_{I+A} is worse than GRU_I in Meituan dataset. One possible reason is that the RNN architecture limits the information fusion. Third, the self-attentive models SASRec_I and SASRec_{I+A} perform better than GRU-based models, it indicates that the self-attentive architecture is particularly suitable for the sequential data in this task. Furthermore, our approach consistently outperforms all the baselines, which indicates the effectiveness of our pre-training method and negative sample generator. Our approach fuses the item- and attribute-level user preference, so that it can improve the performance on conversational recommendation task. Compared with SASRec_{I+A} which has the same architecture as ours, the pre-training method brings large improvement on both datasets. It indicates the effectiveness of our pre-training approach.

Finally, comparing our approach with its ablation variants, we can see that the three components all contribute to the final performance. After removing the Substituted Attributes Discrimination or Masked Item Prediction task, the performance significantly drops. It indicates the importance of the two components.

5 CONCLUSION

This paper presented a novel pre-training approach for conversational recommendation task, which focused on leveraging the item sequence from user history and attribute sequence from conversation data effectively. Based on a self-attentive architecture, our approach designed two pre-training tasks, namely Masked Item Prediction (MIP) and the Substituted Attributes Discrimination (SAD). We further improved our pre-training method by introducing a negative generator to produce high-quality negative samples. Experimental results on two datasets demonstrated the effectiveness of our approach for conversational recommendation task. As future work, we will investigate how to apply our approch to other related recommendation tasks, especially context-aware sequential recommendation and faceted search tasks.

ACKNOWLEDGEMENT

This work was partially supported by the National Natural Science Foundation of China under Grant No. 61872369 and 61832017, Beijing Academy of Artificial Intelligence (BAAI) under Grant No. BAAI2020ZJ0301, and Beijing Outstanding Young Scientist Program under Grant No. BJJWZYJH012019100020098, the Fundamental Research Funds for the Central Universities, the Research Funds of Renmin University of China under Grant No.18XNLG22 and 19XNQ047. Xin Zhao is the corresponding author.

REFERENCES

[1] Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. In *ICLR 2020*.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT 2019*. 4171–4186.

[3] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk. 2016. Session-based Recommendations with Recurrent Neural Networks. In *ICLR 2016*.

[4] Wang-Cheng Kang and Julian J. McAuley. 2018. Self-Attentive Sequential Recommendation. In *ICDM 2018*. 197–206.

[5] Wenqiang Lei, Xiangnan He, Yisong Miao, Qingyun Wu, Richang Hong, Min-Yen Kan, and Tat-Seng Chua. 2020. Estimation-Action-Reflection: Towards Deep Interaction Between Conversational and Recommender Systems. In *WSDM 20*. 304–312.

[6] Yueming Sun and Yi Zhang. 2018. Conversational Recommender System. In *SIGIR 2018*. 235–244.

[7] 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 *NeurIPS 2017*. 5998–6008.

[8] Jun Wang, Lantao Yu, Weinan Zhang, Yu Gong, Yinghui Xu, Benyou Wang, Peng Zhang, and Dell Zhang. 2017. IRGAN: A Minimax Game for Unifying Generative and Discriminative Information Retrieval Models. In *SIGIR 2017*. 515–524.

[9] Kun Zhou, Wayne Xin Zhao, Shuqing Bian, Yuanhang Zhou, Ji-Rong Wen, and Jingsong Yu. 2020. Improving Conversational Recommender Systems via Knowledge Graph based Semantic Fusion. In *KDD 2020*.