Improving Conversational Recommender Systems via Knowledge-Enhanced Temporal Embedding

Chen Ji, Jilu Wang, Jie Xu, Wenxiao Liu, Zihong Yang, Feiran Huang and Chaozhuo Li

September 11, 2023

Improving Conversational Recommender Systems via Knowledge-enhanced Temporal Embedding

Chen Ji[1], Jilu Wang[1], Jie Xu[1⋆], Wenxiao Liu[2], Zihong Yang[2], Feiran Huang[2], Chaozhuo Li[3]

[1] School of Information Science and Technology, Beijing Foreign Studies University, China
[2] College of Cyber Security/College of Information Science and Technology, Jinan University, China
[3] Microsoft Research Asia, Beijing, China

Abstract. Conversational recommender systems are becoming increasingly popular due to their potential to facilitate personalized interactions between users. However, one major challenge lies in accurately representing the semantic meaning of the conversational history to make relevant recommendations. In this paper, we propose a knowledge-enhanced model KMTE to enhance conversational recommender systems. To achieve a more nuanced understanding of users' evolving interests and behaviors over time, a knowledge-enhanced temporal embedding is integrated into KMTE to facilitate the encoding of temporal aspects into the representation of user dialogues. Our proposal is extensively evaluated on a real conversational dataset, and the experimental results demonstrate the effectiveness and superiority of our proposals in improving the accuracy and relevance of conversational recommender systems. Our work sheds light on the potential of leveraging advanced language models to enhance the performance of conversational recommender systems.

Keywords: Conversational Recommender Systems, Pre-trained Language Models, Temporal Embedding.

1 Introduction

Conversational recommender systems aim to tap into users' preferences and recommend items that they might like through multiple rounds of real-time interaction based on natural language. By capturing the user's dynamic preferences through multiple rounds of conversations, the intrinsic interests of users could be gradually modeled and refined, leading to the desirable recommendations. Conversational recommender systems have been widely employed in a myriad of real-life applications, which are capable of improving the user experience and driving revenue for the merchants.

Existing conversational recommender systems could be roughly classified into two categories: attribute-based conversational recommender systems and

⋆ Corresponding author (jxu@bfsu.edu.cn).

generation-based conversational recommender systems. Attribute-based conversational recommender systems are usually trained using reinforcement learning methods, which mainly focus on the design of strategy modules. The primary objective of these models is to achieve highly accurate recommendation outcomes within the fewest conversation rounds feasible. Consequently, they often employ a predetermined conversation template to present the recommendation results. Generative-based conversational recommender systems typically use a sequence-to-sequence model to construct dialogues, with a primary focus on the user interaction module. This category of methods places greater emphasis on delivering a seamless conversational experience to users and enhancing the interpretability of the generated results.

Despite the promising performance of existing conversational recommender models, they are still encountering numerous challenges in natural language understanding and generation. Due to the fluid and dynamic nature of natural language conversations and the inherent leaps in human thinking, it is challenging for dialogue-based systems to promptly capture the evolving user interests during a conversation. For instance, if a user's preference shifts from drama movies to comedy movies within a span of one minute, the system might still be focused on retrieving and recommending drama movies, failing to capture the transition in user interests. While humans are capable of comprehending and accommodating such shifts effortlessly, conversational recommender systems may become confused by this sudden change, ultimately leading to suboptimal recommendation quality.

In this paper, we propose a novel **K**nowledge-enhanced **M**odel via **T**emporal **E**mbedding based on pre-trained language models to enhance the ability of successfully recommending items, dubbed KMTE. To effectively capture the real-time shift of user interest, KMTE is integrated with a novel temporal embedding module, which contributes to capturing dynamic user preferences while incorporating domain-specific knowledge from downstream tasks to enhance the representation capability of sentence embeddings. Specifically, we assign time indices based on the user's historical interests in the conversation, organizing their timestamps into a sequential order that is incorporated into the input layer of the recurrent neural network. Additionally, we continuously pre-train the language models, allowing it to learn more relevant information and discriminative representations. Experimental results on the open dataset demonstrate the effectiveness of our approach in improving the performance of downstream tasks.

Our major contributions are summarized as follows:

- To capture the temporal dynamics of user-item interactions, we introduce a temporal embedding module into KMTE, which facilitates the capability of attending to the dynamic interests of users.
- By replacing the traditional word2vec with the SOTA pre-trained models, we propose a novel approach to incorporate the knowledge of downstream tasks into word embeddings. The experiments demonstrate that our approach significantly outperforms the baseline method on the downstream tasks.

– Through the incorporation of temporal embedding and integration of downstream task knowledge, KMTE significantly boosts the recommendation quality of dialogue recommendation systems. Experimental results demonstrate that our proposal is capable of better understanding users' dynamic behavior patterns and generating more accurate recommendations.

2 Related Work

The conversational recommendation system is composed of two main modules: the recommendation component and the conversation component.

Recommender systems are intended to identify a subset of items from the item pool that satisfy the user's interests. Traditional approaches are highly reliable on historical user-item interactions, such as purchasing records [1, 7]. To address problems of sparse data, conversational recommender systems pay more attention to information from conversations rather than historical interaction. In particular, it is well acknowledged that knowledge graphs can be used to improve recommendation performance and interpretability [5, 15, 20, 16, 21, 14].

Conversation systems aim to generate appropriate responses based on multiround contextual situations. The existing works can be divided into retrieval-based [6, 23, 17, 19, 12] and generation-based approaches [8, 13, 18, 10].

Early conversational recommender systems mainly used predefined actions to interact with users [4]. Nowadays some research has started to integrate these two components for the purpose of better understanding the user's needs. In addition, follow-up studies [2, 11, 22] have adopted external KGs to improve CRS, with a focus mainly on enhanced item representation.

Based on previous studies, we design a novel conversational recommendation method that incorporates user preference, entity and temporal information, with downstream task knowledge injected.

3 Problem Definition

In our proposed approach, the user's historical information is transformed into a time sequence represented as $X = \{x_1, x_2, ..., x_N\}$ (x_i indicates a single token). This time sequence is then combined with other relevant features within the model. Moreover, we aim to generate word embeddings based on BERT, incorporating downstream task knowledge to enhance attention towards specific items during the recommendation process.

4 Methodology

4.1 Framework

Figure 3 demonstrates the framework of the proposed KMTE model. We integrate user interests, entity features, and time information to enhance the model's

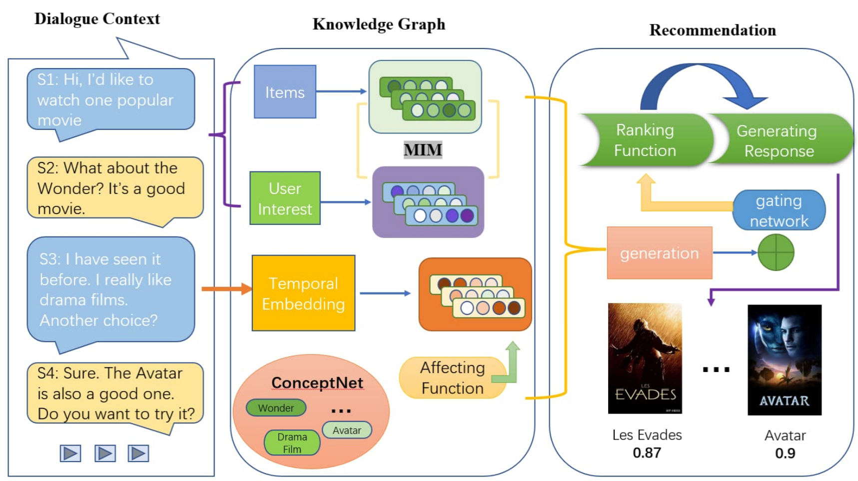

Fig. 1. Framework of the proposed KMTE model.

understanding of users' dynamic interests, filter out less relevant long-term hobbies, and generate personalized real-time recommendations for users, thereby improving the performance and effectiveness of the recommender system.

4.2 Knowledge Graph Encoder

We build a knowledge graph by mapping user interests and mentioned entities (movies) in the conversation to embedding vectors. Specifically, we define a user interest embedding vector $IU(u)$ and an entity embedding vector $IE(e)$, where u represents the user and e represents the movie entity:

$$IU(u) = Embedding(u), IE(e) = Embedding(e) \tag{1}$$

Embedding(u) maps the user to their interest embedding vector while Embedding(e) maps a movie entity to its corresponding embedding vector.

By concatenating the user interest and movie entity embedding vectors, we construct a knowledge graph. This knowledge graph can be represented as a graph, where user interest vectors and entity embedding vectors serve as nodes, and the associations between them are represented by edges.

$$A(u, e) = Similarity(IU(u), IE(e)) \tag{2}$$

In this graph, the association between users and movies can be measured by calculating the similarity between the user interest embedding vector $IU(u)$ and the entity embedding vector $IE(e)$. In this paper, we use cosine similarity to achieve that goal.

By building such a knowledge graph, we can leverage the graph structure and the associations between nodes to infer user interests and provide more accurate and personalized recommendations.

4.3 Dynamic Interest Module

We utilize temporal embedding to represent the relative time intervals between messages in a conversation. By calculating the difference in time offsets between messages and mapping it to the corresponding temporal embedding vector, we can capture the dynamic characteristics of the messages. Additionally, we combine user interest features with movie entity features to enable the model to better understand the evolving interests of users. Suppose the time offset of the i-th message is t_i, and the time offset of the previous message is t_{i-1}.

$$TE(t_i - t_{i-1}) = Embedding(t_i - t_{i-1}) \tag{3}$$

Here, Embedding is a function that maps the relative time interval to the corresponding time embedding vector. By incorporating time information in this way, we can capture the temporal dynamics between messages and enhance the modeling capabilities of the recommendation system.

The temporal embedding vector $TE(t)$ is adjusted through the impact function g_time. t represents the given time. α is a tuning parameter that controls the rate of time decay. The exponential function $exp(-\alpha * t)$ represents the weight of time decay, indicating that the impact of time embedding vectors decreases as the distance from the current time increases:

$$g_{time}(TE(t)) = exp(-\alpha * t) \tag{4}$$

By using such an exponential function, the time relevance calculation is further defined as:

$$S(u, e, t) = Similarity(IU(u), IE(e)) * g_{time}(TE(t)) \tag{5}$$

$Similarity(IU(u), IE(e))$ represents the similarity calculation function between the user interest embedding vector $IU(u)$ and the entity embedding vector $IE(e)$. $exp(-\alpha * t)$ adjusts the time relevance based on the given time t. Obviously, the closer the time is to the present, the greater the influence it will have, while the impact of past events gradually diminishes. This approach allows the model to focus more on the user's recent interests and activities, improving the modeling capability for dynamic user interests.

4.4 Recommender Module

Unlike traditional recommender systems, which typically rely on previous interaction records, we assume that no prior interaction data is accessible. Our

Table 1. Experimental results on different models.

Models	Metrics		
	R@1	**R@10**	**R@50**
Popularity	0.012	0.061	0.179
TextCNN	0.013	0.068	0.191
ReDial	0.024	0.140	0.320
KBRD	0.031	0.150	0.336
KGSF	0.037	0.181	0.372
KMTE	**0.044**	**0.236**	**0.456**

recommendation generation formula is as follows:

$$R(u) = weightedsum(S(u, e, t) * A(u, e))/weightedsum(S(u, e, t)) \tag{6}$$

In this formula, we utilize a weighted sum operation *weightedsum* to calculate the recommendation result $R(u)$. We multiply $S(u, e, t)$ and $A(u, e)$, which represents the weighted combination of the user's dynamic interest and the relevance between the user and the entity. Then, we normalize the result by dividing it with the weighted sum of the time relevance.

This approach allows us to accurately capture the user's real-time interests rather than relying solely on historical preferences, thereby enabling the system to generate more precise recommendations.

4.5 Parameter Learning

Parameter learning plays a crucial role in the proposed model. To optimize the performance, we begin by calculating the loss function for the recommendation task, taking into account potential regularization terms. The resulting loss function is denoted as L, which encompasses the overall objective of the model:

$$L = L_{rec} + \lambda * L_{reg} \tag{7}$$

Next, we compute the gradients of the loss function with respect to the model parameters θ to determine the direction of parameter updates. The learning rate α determines the step size of parameter updates, controlling the magnitude of parameter changes during each update.

$$\theta_{new} = \theta_{old} - \alpha * \Delta L \tag{8}$$

By selecting an appropriate learning rate and balancing the weights of the regularization terms, we can effectively train the model and enhance its recommendation performance.

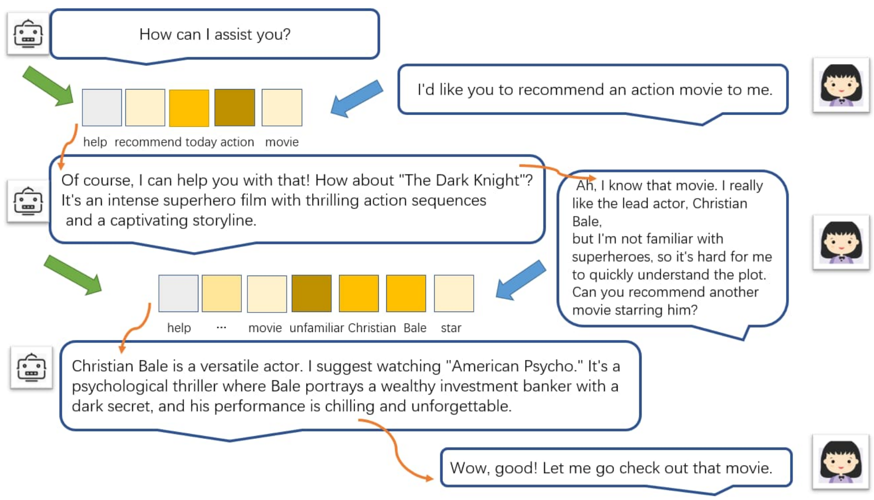

Fig. 2. A sampled conversation case for quantitative analysis.

5 Experiment

5.1 Experimental Settings

Dataset We evaluate our model on DIALog (REDIAL) dataset, which is a conversational recommendation dataset published by Towards Deep Conversational Recommendations(2018). We filter out textual information in the dataset and replace the movie numbers with their real names to improve the completeness and readability of the text.

Baselines We compare the proposed knowledge-intensive model with popular baselines for verifying the performance. Among these baselines, Popularity and TextCNN [3] are recommendation methods. We do not include other recommendation models, since there are no historical user-item interaction records except the text of a single conversation. Besides, REDIAL [9], KBRD [2] and KGSF [22] are all conversation recommendation methods.

5.2 Quantitative Analysis

Table 1 presents the performance of different proposed model. As we can see, Popularity and TextCNN perform evenly,but pale in comparison to the three CRS models Redial, KBRD and KGSF which are more independent by using entities only to make recommendations. Furthermore, KGSF performs more outstandingly because it achieves semantic fusion via knowledge graphs. It suggests that richer information is useful for enhancing data representation. Finally, our model KMTE outperforms the baseline by a large margin. KMTE fuses KG and valid temporal information while injecting downstream task knowledge to improve the system's understanding of conversational information, thus greatly improving recommendation quality.

5.3 Qualitative Analysis

In this multi-turn dialogue shown in Figure 2, the user's interest transition occurs naturally through the association with the actor. This recommendation not only caters to the user's preference for Bale but also addresses their desire for a movie outside the superhero genre. To conclude, the dialogue recommendation system accurately captures the user's dynamic interest transition by leveraging the information from the user's timestamps.

5.4 Ablation Study

We conduct the ablation study based on three variations of our complete model, including: (1) using Word2Vec to generate word embeddings; (2) using pre-trained BERT instead; and (3) injecting downstream task knowledge into the trained model. We evaluate the performance by calculating recall@k (k=1, 10, 50) and visualize the results (See Figure 3). As shown in the graph, the performance of using Word2Vec-generated embeddings is relatively poor in the recommendation system, while using pre-trained BERT embeddings shows performance enhancement. However, the best results are achieved when injecting downstream task knowledge into the model, further boosting the performance of the recommendation system.

6 Conclusion

In this study, we propose a new knowledge-intensive model based on BERT's masked language model and inject knowledge of downstream tasks to enable a better understanding of natural language. Our approach fully considers the characteristics of the conversational dataset and leverages the power of MLM to effectively train a model suitable for the conversational dataset and generate the corresponding embedding vector. Our experiments show that our approach outperforms state-of-the-art models, highlighting the effectiveness of incorporating temporal information into the model and generating text for the conversational dataset.

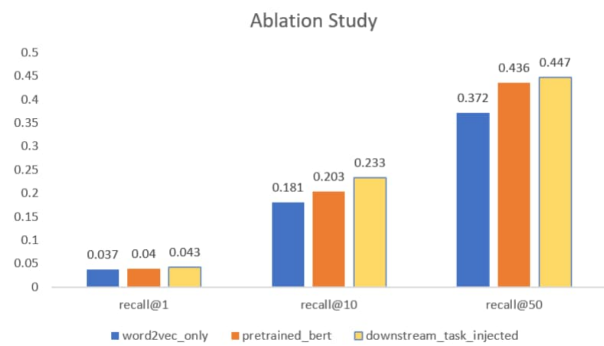

Fig. 3. Ablation Study.

Acknowledgement

This work was supported by the Fundamental Research Funds for the Central Universities (No. 2021QD014) and in part by the National Natural Science Foundation of China (No. 62272200, 61932010, U22A2095).

References

1. Bobadilla, J., Ortega, F., Hernando, A., Gutiérrez, A.: Recommender systems survey. Knowledge-based systems **46**, 109–132 (2013)
2. Chen, Q., Lin, J., Zhang, Y., Ding, M., Cen, Y., Yang, H., Tang, J.: Towards knowledge-based recommender dialog system. arXiv preprint arXiv:1908.05391 (2019)
3. Chen, Y.: Convolutional neural network for sentence classification. Master's thesis, University of Waterloo (2015)
4. Christakopoulou, K., Radlinski, F., Hofmann, K.: Towards conversational recommender systems. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. pp. 815–824 (2016)
5. Huang, J., Zhao, W.X., Dou, H., Wen, J.R., Chang, E.Y.: Improving sequential recommendation with knowledge-enhanced memory networks. In: The 41st international ACM SIGIR conference on research & development in information retrieval. pp. 505–514 (2018)
6. Ji, Z., Lu, Z., Li, H.: An information retrieval approach to short text conversation. arXiv preprint arXiv:1408.6988 (2014)
7. Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 426–434 (2008)
8. Li, J., Galley, M., Brockett, C., Gao, J., Dolan, B.: A diversity-promoting objective function for neural conversation models. arXiv preprint arXiv:1510.03055 (2015)

9. Li, R., Ebrahimi Kahou, S., Schulz, H., Michalski, V., Charlin, L., Pal, C.: Towards deep conversational recommendations. Advances in neural information processing systems **31** (2018)
10. Li, R., Zhao, J., Li, C., He, D., Wang, Y., Liu, Y., Sun, H., Wang, S., Deng, W., Shen, Y., et al.: House: Knowledge graph embedding with householder parameterization. In: International Conference on Machine Learning. pp. 13209–13224. PMLR (2022)
11. Liao, L., Takanobu, R., Ma, Y., Yang, X., Huang, M., Chua, T.S.: Deep conversational recommender in travel. arXiv preprint arXiv:1907.00710 (2019)
12. Pang, B., Li, C., Liu, Y., Lian, J., Zhao, J., Sun, H., Deng, W., Xie, X., Zhang, Q.: Improving relevance modeling via heterogeneous behavior graph learning in bing ads. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 3713–3721 (2022)
13. Serban, I., Sordoni, A., Bengio, Y., Courville, A., Pineau, J.: Building end-to-end dialogue systems using generative hierarchical neural network models. In: Proceedings of the AAAI conference on artificial intelligence. vol. 30 (2016)
14. Tian, Z., Li, C., Zuo, Z., Wen, Z., Hu, X., Han, X., Huang, H., Wang, S., Deng, W., Xie, X., et al.: Multi-grained topological pre-training of language models in sponsored search. In: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 2189–2193 (2023)
15. Wang, X., Wang, D., Xu, C., He, X., Cao, Y., Chua, T.S.: Explainable reasoning over knowledge graphs for recommendation. In: Proceedings of the AAAI conference on artificial intelligence. vol. 33, pp. 5329–5336 (2019)
16. Wang, Y., Li, C., Liu, Z., Li, M., Tang, J., Xie, X., Chen, L., Yu, P.S.: An adaptive graph pre-training framework for localized collaborative filtering. ACM Transactions on Information Systems **41**(2), 1–27 (2022)
17. Zhang, P., Yan, Y., Li, C., Wang, S., Xie, X., Song, G., Kim, S.: Continual learning on dynamic graphs via parameter isolation. arXiv preprint arXiv:2305.13825 (2023)
18. Zhang, Y., Li, C., Xie, X., Wang, X., Shi, C., Liu, Y., Sun, H., Zhang, L., Deng, W., Zhang, Q.: Geometric disentangled collaborative filtering. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 80–90 (2022)
19. Zhao, J., Qu, M., Li, C., Yan, H., Liu, Q., Li, R., Xie, X., Tang, J.: Learning on large-scale text-attributed graphs via variational inference. arXiv preprint arXiv:2210.14709 (2022)
20. Zhao, W.X., He, G., Yang, K., Dou, H., Huang, J., Ouyang, S., Wen, J.R.: Kb4rec: A data set for linking knowledge bases with recommender systems. Data Intelligence **1**(2), 121–136 (2019)
21. Zhao, Y., Li, C., Peng, J., Fang, X., Huang, F., Wang, S., Xie, X., Gong, J.: Beyond the overlapping users: Cross-domain recommendation via adaptive anchor link learning. In: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 1488–1497 (2023)
22. Zhou, K., Zhao, W.X., Bian, S., Zhou, Y., Wen, J.R., Yu, J.: Improving conversational recommender systems via knowledge graph based semantic fusion. In: Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining. pp. 1006–1014 (2020)
23. Zhou, X., Dong, D., Wu, H., Zhao, S., Yu, D., Tian, H., Liu, X., Yan, R.: Multi-view response selection for human-computer conversation. In: Proceedings of the 2016 conference on empirical methods in natural language processing. pp. 372–381 (2016)