ISSN: 2664-9640

# A hybrid recommendation model for multi-objective feature learning based on knowledge graphs

Yimao Liu[1], Yan Peng[2,] * and Lushuai Niu[1]

[1] School of Automation and Information Engineering, Sichuan University of Science and Engineering, Zigong 643000, China;

[2] School of Engineering Practice Center, Sichuan University of Science and Engineering, Zigong 643000, China.

* Corresponding Author

## Abstract

**Personalized recommendation systems appear more and more frequently in people's lives, however, the traditional partial recommendation system model input vectors are sparse, and a large number of user and item features do not interact, which makes the final recommendation result biased and eventually affects the system performance. Therefore, in this paper, we propose a multi-objective hybrid recommendation algorithm model based on knowledge graph, MOHR-KG. Firstly, we use convolutional network to extract item features and combine them with the item's own features to transform them into low-dimensional vectors, and then obtain the auxiliary item information in the knowledge graph, and then introduce the attention mechanism, and then cross-train to mine the feature information to output results. The experimental results show that; compared with the current traditional recommendation methods, MOHR-KG has good performance in evaluation indexes such as accuracy and improves the system performance.**

## Keywords

**Knowledge graph; attention mechanism; recommendation algorithm; multi-objective.**

## 1. Introduction

With the development of information technology and the Internet, the problem people face is not the lack of information, but how to filter information, thus, the recommendation system was born. Recommendation systems aim to provide users with personalized recommendation services [1] and help them get the information they want faster and more accurately.

After continuous development, traditional recommendation algorithms are no longer able to complete the recommendation task well. Currently, researchers have achieved good results by combining deep learning and traditional recommendation algorithms to complete the recommendation task, but still face many problems such as cold start [2], so we propose that the recommendation can be enhanced with the help of knowledge graphs as an auxiliary information source.

Therefore, in this paper, we propose a multi-objective hybrid recommendation model based on knowledge graph to fully exploit the user and item information so that the model enables the interaction of features between users and items, and finally, we improve the performance of the whole recommendation system through training and optimization of the model to make effective recommendations.

ISSN: 2664-9640

## 2. Related Work

Receiving inspiration from the successful application to graph embedding in various tasks, researchers have recently tried to use graphs to improve the performance of recommendation algorithms. personalized entity recommendation (PER) [3] treats KG as a heterogeneous information network and extracts potential features based on meta-paths to represent different types of relational paths of connectivity between users and items, but PER relies heavily on manually designed meta-paths, which limits its use in generic schemes. Collaborative knowledge base embedding (CKE) [4] combines collaborative filtering algorithm (CF) [5] with structural knowledge and textual knowledge in a unified framework, which is more suitable for in-graph applications than recommendation applications, and the loose connection between CF and KGE modules is less effective for recommendation models.

Because, to address the limitations of previous work, we propose a multi-objective hybrid recommendation model based on knowledge graphs, MOHR-KG, which is a general end-to-end deep recommendation framework [6], aiming to use knowledge graph embeddings to assist in recommendation tasks, by cross-training the recommendation module and graph embedding module to complete the recommendation task, in which we introduce a cross-compression unit that achieve a high-level interaction between the two modules with enough higher-order features interacting with each other between entities. We also add an attention mechanism to the cross-compression unit to further improve the system performance.

## 3. MOHR-KG models

### 3.1. Framework of MOHR-KG

The framework of MOHR-KG is shown in Figure 1; it is mainly composed of four parts: recommendation module, knowledge graph embedding module, cross-compression unit and attention mechanism.

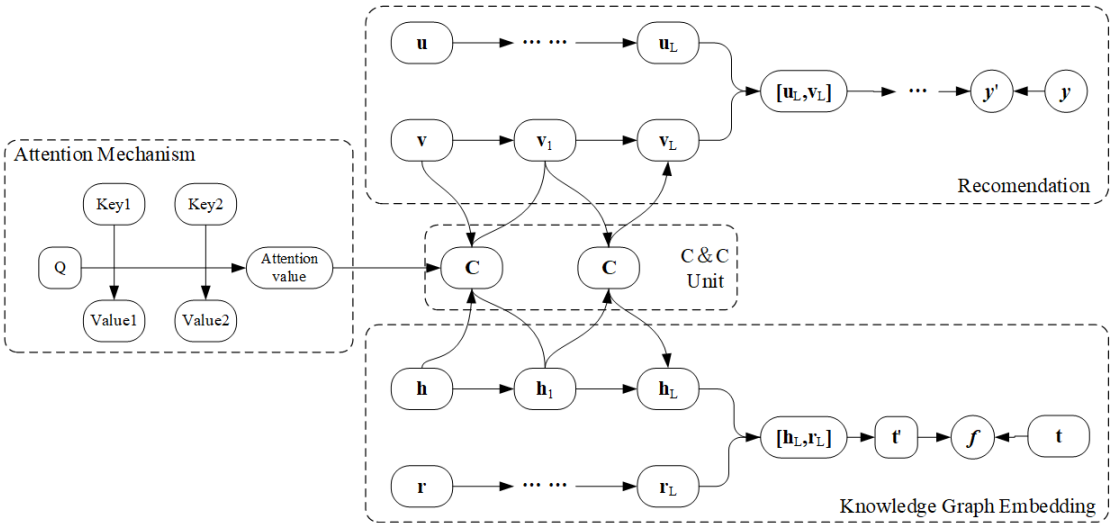


Figure 1: Framework of MOHR-KG

### 3.2. Recommended Modul

The input in the recommendation module is composed of two original feature vectors u and v [7], whose potential compression features we extract using the MLP of the L-layer [8].

$$u_L = M(M(\ldots M(u))) = M^L(u) \tag{1}$$

Where M(x) is a fully connected neural network layer with weights W, deviations b and nonlinear activation function σ(y), and for item v, we extract its features using L cross-compression units.

$$v_L = E_{e \sim S(v)}\left[C^L(\mathbf{v},\mathbf{e})[\mathbf{v}]\right] \tag{2}$$

Where S(v) is the set v of item-related entities.

After obtaining the potential features of user u and item v, we combine the two by means of a prediction function to obtain the prediction results.

$$\hat{y}_{uv} = \sigma\left(f_{RS}\left(u_L, v_L\right)\right) \tag{3}$$

## 3.3. Knowledge graph embedding module

In order to effectively embed the auxiliary information of the knowledge graph into the recommender system, we need to transform the semantic information into low-latitude vectors and embed multi-relational data in the recommender model [4]. We first use multiple cross-compression units and nonlinear layers for the head h and relation r [9,10], and then connect their potential features together to act on the K-layer MLP for predicting the tail t.

$$h_L = E_{v \sim S(h)}\left[C^L(v,h)[e]\right] \tag{4}$$

$$r_L = M^L(r) \tag{5}$$

$$\mathrm{t}' = M^K\left(\begin{bmatrix} h_L \\ r_L \end{bmatrix}\right) \tag{6}$$

Where S(h) is the set of entity association terms, t' is the prediction vector of t, and finally the function f score is calculated [11].

$$\mathrm{score}(h,r,t) = f_{KG}(t,t') \tag{7}$$

## 3.4. Cross-compression unit

The intersection operation consists of each possible feature interaction of the item v and the entity e associated with it.

$$C_l = v_l e_l^T = \begin{bmatrix} v_l^{(1)} e_l^{(1)} & \dots & v_l^{(1)} e_l^{(d)} \\ \dots & & \dots \\ v_l^{(d)} e_l^{(1)} & \dots & v_l^{(d)} e_l^{(d)} \end{bmatrix} \tag{8}$$

Where $v_l$ and $e_l$ denote the items and entities of the Lth layer, respectively, the relationship between item v and its entity e is explicitly modeled in the cross-feature matrix. Then, the item and entity feature vectors of the next layer are output by projecting into their potential representation space through the cross-feature matrix as follows.

$$\begin{aligned} v_{l+1} &= C_l w_l^{VV} + C_l^T w_l^{EV} + b_l^V = v_l e_l^T w_l^{VV} + e_l v_l^T w_l^{EV} + b_l^V \\ e_{l+1} &= C_l w_l^{VE} + C_l^T w_l^{EE} + b_l^E = v_l e_l^T w_l^{VE} + e_l v_l^T w_l^{EE} + b_l^E \end{aligned} \tag{9}$$

where $w_i$ and $b_i$ are the weights and biases of the network, respectively.

## 3.5. Attentional Mechanisms

The attention mechanism to some extent simulates the characteristics of human vision, which tends to focus on the target when observing things. Using the attention mechanism can help the neural network focus on feature information that is more relevant to the target task. The data s to be processed are constituted into a series of key-value pairs <Key, Value>, Key is the attribute and Value denotes the corresponding value. Given a certain attribute Q in the target, the weight coefficient corresponding to Value is obtained by calculating the correlation between Q and Key, and then the Value is weighted to obtain the final Attention value, which is calculated as follows.

$$a_i = \text{softmax}\left(\text{sim}\left(Q, \text{Key}_i\right)\right)$$

$$\text{softmax}\left(x_i\right) = e^{x_i} / \sum_{i=1}^{h} e^{x_i}$$

$$\text{Atten} = \sum_{i=1}^{n} a_i \text{ Value}_i \tag{10}$$

Where sim(-) denotes the correlation function, ai denotes the weight coefficient corresponding for each Value

### 3.6. Loss function

The complete loss function of the model is.

$$L = L_{RS} + L_{KG} + L_{REG} = \sum_{u \in U, v \in V} J\left(\hat{y}_{uv}, y_{uv}\right)$$

$$-\lambda_1 \left( \sum_{(h,r,t) \in G} \text{score}(h,r,t) - \sum_{\left(h',r',t'\right) \notin G} \text{score}\left(h',r',t'\right) \right) + \lambda_2 \| W \|_2^2 \tag{11}$$

where $L_{RS}$ denotes the loss of the recommendation module, $L_{KG}$ denotes the loss of the knowledge graph embedding module, and $L_{REG}$ denotes the canonical term.

## 4. Experiment and Evaluation

### 4.1. Dataset

The MovieLens dataset is a set of movie rating data provided by MovieLens users from the late 1990s to the beginning of the 21st century. The data mainly includes movie titles, genres and eras, users' ages, genders, occupations, and zip codes, and users' ratings of movies.

This experiment uses one of the MovieLens-1M datasets, which mainly includes 6040 users, 3882 movies, and 1 million records of users' ratings of movies.

### 4.2. Evaluation metrics

In this paper, two metrics, ACC (accuracy) and AUC (area under curve) of ROC curve, are used to evaluate the MOHR-KG model, aiming to test the performance of the model on the public dataset.

The ACC metric is used to describe the proportion of classifications that are correctly predicted for the overall, i.e.

$$\text{ACC} = \frac{n_{\text{correct}}}{n_{\text{total}}} \tag{12}$$

Where $n_{correct}$ indicates the number of records correctly predicted and $n_{total}$ indicates the number of all test data.

The AUC metric is a quantification of the ROC curve. In some cases, the ROC curve is not always smooth due to the threshold value taking, and it is difficult to determine the performance of the model. Therefore, AUC is chosen to evaluate the model, and the value of AUC is the area formed by the ROC curve and the FPR axis.

### 4.3. Experimental results and analysis

In order to verify the effectiveness of the MOHR-KG model, this paper compares and analyzes this model with some current representative and mainstream recommendation algorithms

(PER [3], LibFM [12], CKE [4]), etc., and the final results are averaged three times as shown in Table 1.

Table 1: Performance comparison of different algorithms in common dataset

| Model | ACC | AUC |
|---|---|---|
| PER | 0.656 | 0.710 |
| CKE | 0.732 | 0.798 |
| KGCN | 0.787 | 0.811 |
| LibFM | 0.822 | 0.845 |
| Wide & Deep | 0.811 | 0.869 |
| MOHR-KG | 0.844 | 0.902 |

## 5. Conclusion

In this paper, we propose a multi-objective feature learning [14] hybrid recommendation model based on knowledge graphs, which mainly addresses the problems of vector initialization and feature extraction in recommendation systems [13]. The experimental results show that this paper's method has significant performance improvement on general-purpose datasets. However, due to the complexity of the recommender system, so it is in the practical application there are still many problems that need to be discussed in depth to solve, for example, we can build more accurate user portraits so as to enrich the information [15,16], which will be more helpful to our model as a way to improve the system performance.

## References

[1] PAULC,JAY A,EMRES. Deep neural networks for youtube recommendations[C]// /Proceedings of the10th ACM Conference on Recommender Systems. Boston: ACM,2016:191-198.

[2] BENGIO Y, COURVILLE A,VINCENTP. Representation learning: a review and new perspectives[J]. IEEE Trans action on Pattern Analysis and MachinIntelligence,2013,35(8):1798-1828.

[3] Xiao Yu, Xiang Ren, Yizhou Sun, Quanquan Gu, Bradley Sturt, Urvashi Khandelwal, Brandon Norick, and Jiawei Han. 2014. Personalized entity recommendation: A heterogeneous information network approach. In Proceedings of the 7th ACM International Conference on Web Search and Data Mining. 283–292.

[4] Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. 2016. Collaborative knowledge base embedding for recommender systems. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 353–362.

[5] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. Computer 42, 8 (2009).

[6] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and S Yu Philip. 2017. Learning Multiple Tasks with Multilinear Relationship Networks. In Advances in Neural Information Processing Systems. 1593–1602.

[7] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In Proceedings of the 26th International Conference on World Wide Web. 173–182.

[8] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. 2015. Collaborative deep learning for recommender systems. In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 1235–1244.

[9] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning Entity and Relation Embeddings for Knowledge Graph Completion.. In The 29th AAAI Conference on Artificial Intelligence. 2181–2187.

[10] Ruobing Xie, Zhiyuan Liu, and Maosong Sun. 2016. Representation Learning of Knowledge Graphs with Hierarchical Types.. In IJCAI. 2965–2971.

[11] Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. 2016. Cross-stitch networks for multi-task learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 3994–4003.

[12] Steffen Rendle. 2012. Factorization machines with libfm. ACM Transactions on Intelligent Systems and Technology (TIST) 3, 3 (2012), 57.

[13] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge graph embedding: A survey of approaches and applications. IEEE Transactions on Knowledge and Data Engineering 29, 12 (2017), 2724–2743.

[14] Yu Zhang and Qiang Yang. 2017. A survey on multi-task learning. arXiv preprint arXiv:1707.08114 (2017).

[15] Wenlu Zhang, Rongjian Li, Tao Zeng, Qian Sun, Sudhir Kumar, Jieping Ye, and Shuiwang Ji. 2015. Deep model based transfer and multi-task learning for biological image analysis. In 21st ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2015. Association for Computing Machinery.

[16] Qiang Zhou and Qi Zhao. 2016. Flexible Clustered Multi-Task Learning by Learning Representative Tasks. IEEE Trans. Pattern Anal. Mach. Intell. 38, 2 (2016), 266–278.