Commodity Personalized Recommendation Algorithm Based on the Knowledge Graph

Kai Yuan Yang, Jun Yang and Zhi Xin Sun

October 9, 2023

Commodity personalized recommendation algorithm based on the knowledge graph

Kai Yuan Yang
Post Big Data Technology and Application Engineering Research Center of Jiangsu Province, Post Industry Technology Research and Development Center of the State Posts Bureau (Internet of Things Technology) Nanjing University of Posts and Telecommunications
Nanjing China
1292694360@qq.com

Jun Yang
Changzhou Anyi SoftwareCc.,Ltd
Changzhou China
472312068@qq.com

Zhi Xin Sun*
Broadband Wireless Communication Technology Engineering Research Center of the Ministry of Education Nanjing University of Posts and Telecommunication
Nanjing China
sunzx@njupt.edu.cn

***Abstract*—Personalized recommendation systems have become an important part of e-commerce, social media, and other applications. However, the traditional collaborative filtering algorithm is only based on the user's scoring history of the product, ignoring the attributes and characteristics of the product itself. To solve this problem, this paper proposes a personalized recommendation algorithm based on knowledge graph, which can combine the similarity between goods and user preferences to make recommendations and add the scoring mechanism, thus improving the accuracy and practicability of the recommendation system. Experimental results show that our algorithm outperforms the traditional user-based and item-based co-filtering algorithms in evaluation indexes such as accuracy, recall and F1 value, demonstrating the effectiveness and feasibility of this algorithm in the field of personalized recommendation.**

Keywords—graph, system, algorithm, similarity

I. INTRODUCTION

With the rapid development of e-commerce, social media, and other application fields, personalized recommendation systems have become an important application. However, traditional collaborative filtering algorithms usually can only recommend products based on the user's scoring history of the product, ignoring the attributes and characteristics of the product itself. Such recommendation results are difficult to meet the actual needs of users, because users usually consider other attributes of the goods when buying goods, such as brand, type, place of production, and so on. To solve this problem, the researchers began trying to introduce knowledge graph technology into a personalized recommendation system[1].

Knowledge graph is a data structure based on graph theory that can represent the relationship between entities as the edges of a graph. In a knowledge graph, entities are usually represented as nodes, and edges represent the relationship between entities. By establishing relationships between entities, knowledge graphs can better describe the relevance and semantic information between entities. Therefore, introducing the knowledge graph technology into the personalized recommendation system can make full use of the similarity between goods and user preferences to recommend, and improve the accuracy and practicability of the recommendation system.

In this paper, we propose a personalized recommendation algorithm based on a knowledge graph that combines similarity between commodities and user preferences for recommendations. We use relevant datasets to compare our algorithm with traditional user-based and item-based collaborative filtering algorithms, and show that our algorithm is better than traditional collaborative filtering algorithm in evaluation indexes such as accuracy, recall and F1 value. This work demonstrates the effectiveness of introducing a knowledge graph into a personalized recommendation algorithm, which is important for improving the user experience[2].

II. RELATED TECHNICAL RESEARCH

With the rapid development of e-commerce and the popularity of the Internet, personalized recommendation has become one of the important functions of the e-commerce platform. Although the traditional collaborative filtering algorithm is simple and effective, it has cold start problems and data sparsity problems, which leads to poor recommendation effect. In order to solve these problems, scholars have proposed a personalized recommendation algorithm based on the knowledge graph, which uses the similarity between the goods in the knowledge graph and the user's preference information to make more accurate recommendations[3].

In China, many scholars have studied and discussed the personalized recommendation algorithm based on knowledge graph. Several relevant articles will be reviewed below.

First, Li Jun et al. proposed a graph convolutional neural network recommendation model (KGAT) based on the knowledge graph. They built a bimodal network on the knowledge graph, which fused the information of users and goods, and used the graph convolutional neural network (GCN) to extract features. Experimental results show that the KGAT model outperforms the traditional collaborative filtering and graph-based recommendation algorithms in indicators such as accuracy and recall.

In addition, a cross-domain recommendation method based on knowledge graph and multi-source data was proposed by Wang Qian et al. They fuse the knowledge graph and data from other sources to build a recommendation system for multi-source data. Experimental results show that the proposed method can effectively improve the accuracy and efficiency of the recommendations.

Moreover, a deep learning recommendation model based on knowledge graph and attention mechanisms was proposed by Liang Zhou et al. They constructed a deep learning model on the knowledge graph, and used the attention mechanism to improve the accuracy and personalization of recommendations. Experimental results show that the model outperformed the traditional collaborative filtering and graph-based recommendation algorithms in evaluation indexes such as accuracy, recall and F1 value.

Therefore, it can be seen that the personalized recommendation algorithm based on knowledge graph is one of the hot spots of recommendation system research, and has important application value. In this paper, we propose a personalized recommendation algorithm based on knowledge graph, and prove the effectiveness and superiority of this algorithm by experimenting on a grape sales data set of an agricultural company. It is hoped that the research of this paper can provide some reference value for the development and application of the recommendation system[4].

III. Introduction of the Recommendation Algorithm

A. *Mathematical formula of the algorithmic model*

- Formula for calculating user interest degree

For the user u and commodity i, we can recommend the products by calculating the interest degree of the user u commodity i. The specific formula is as follows:

$$\hat{r}_{ui} = \frac{\sum_{j \in N(u) \cap N(i)} W_{u,j} \cdot W_{i,j} \cdot r_{uj}}{\sum_{j \in N(u) \cap N(i)} W_{u,j} \cdot W_{i,j}}$$

$$r_{uj} \sum_{j \in N(u) \cap N(i)} W_{u,j} \cdot W_{i,j} \cdot r_{uj}$$

Where, r_{uj} represents the user u's score of commodity j, the second formula represents the user's predicted score of commodity i calculated by the user's score of adjacent nodes. N (u) and N (i) represent the set of adjacent nodes of the user and commodity in the knowledge graph respectively, and represent the edge weights of the user and commodity nodes and adjacent nodes respectively.$W_{u,j} W_{i,j}$

In this recommendation algorithm, the denominator represents the sum of the edge weights of the user over all adjacent nodes (i. e., the nodes connected to the commodity). Therefore, when the denominator is larger, it means that users are more connected and have wide interests, which also means that users can be affected by more similar products, so the interest in different products will be more affected by these similar categories. However, the sum of the user's score is calculated, indicating the degree of influence of these adjacent nodes on the user's interest degree. The weighting coefficient is the edge weight between adjacent nodes and commodities. Therefore, when the denominator is larger, the value of the weighted sum in the molecule may also increase, thus having a greater impact on the user's prediction score of the product[5].

- Calculation formula of commodity similarity

To calculate the similarity between different commodities, we use the cosine similarity formula, as follows:

$$\mathrm{Sim}\ (i,j) = \frac{\sum_{u \in U_{i,j}} r_{ui} \cdot r_{uj}}{\sqrt{\sum_{u \in U_i} r_{ui}^2} \cdot \sqrt{\sum_{u \in U_j} r_{uj}^2}}$$

$U_{i,j}$ Where, represents the set of users that score both goods i and j, and the set of users that score movies i and j, respectively.$U_i U_j r_{ui}$ Represents how the user u scores the item i.

- Recommendation formula based on the user

In order to recommend products to users, we first calculate the interest degree of users and all products. The specific formula is as follows:

$$\hat{r}_{ui} = \frac{\sum_{j \in N(u) \cap N(i)} W_{u,j} \cdot W_{i,j} \cdot r_{uj}}{\sum_{j \in N(u) \cap N(i)} W_{u,j} \cdot W_{i,j}}$$

Then, we select the first K products with the highest interest for recommendation. The specific formula is as follows:

$$\mathrm{Rec}\ (u) = \{i \in I \mid \hat{r}_{ui} > T\}(|\mathrm{Rec}\ (u)| = K)$$

Where I represents the collection of all goods, T represents the collection of goods purchased by the current user u, and represents the correlation between the user and the goods$W_{u,j}$uj

- Traditional user-based collaborative filtering recommendation algorithm model:

For user u, the similarity between it and other users v is calculated, assuming cosine similarity to measure similarity, there is:

$$\mathrm{sim}\ (u,v) = \frac{\sum_{j \in I_u \cap I_v} W_{u,j} \cdot W_{v,j}}{\sqrt{\sum_{j \in I_u} W_{u,j}^2} \cdot \sqrt{\sum_{j \in I_v} W_{v,j}^2}}$$

$I_u I_v$ Where, and are the collection of goods seen by the user u and the user v, respectively.

For the commodity j that the user u has not seen, calculate the prediction score, assuming that the weighted average is used to calculate the prediction score, there are:

$$\hat{r}_{u,j} = \frac{\sum_{v \in U} \mathrm{sim}\ (u,v) \cdot r_{v,j}}{\sum_{v \in U} \mathrm{sim}\ (u,v)}$$

$r_{v,j}$ Which is the user v's rating of the commodity j.

- Model formula of commodity recommendation algorithm based on knowledge graph:

ui For the user, to calculate the similarity with the other products, assuming that the cosine similarity is used to measure the similarity, there is:

$$\mathrm{sim}\ (u,i) = \frac{\sum_{j \in N(u) \cap N(i)} W_{u,j} \cdot W_{i,j}}{\sqrt{\sum_{j \in N(u)} W_{u,j}^2} \cdot \sqrt{\sum_{j \in N(i)} W_{i,j}^2}}$$

$N(u) N(i)$ Where and represent the set of adjacent nodes of user and commodity in the knowledge graph respectively, and represent the edge weights of user and commodity nodes and adjacent nodes respectively.$W_{u,j} W_{i,j}$

ui For the goods that the user has not seen, the prediction score is calculated[6]. Assuming that the weighted average is used to calculate the prediction score, there are:

$$\hat{r}_{u,i} = \frac{\sum_{j \in I} \mathrm{sim}\ (u,j) \cdot r_{j,i}}{\sum_{j \in I} \mathrm{sim}\ (u,j)}$$

$r_{j,i}$ jiThis is the correlation between goods and goods.

Compared with the traditional user-based and item-based collaborative filtering algorithm, the personalized recommendation algorithm based on knowledge graph proposed in this paper has the following advantages:

More accurate recommendation results: the traditional collaborative filtering algorithm is mainly recommended based on the historical behavior and interests of users, while it ignores the similarity information between goods. The algorithm in this paper incorporates similarity information between commodities into the recommendation model to predict user preferences and needs more accurately.

Better scalability and adaptability: The Knowledge Map is a well-scalable and adaptable data structure that can accommodate a large number of entities and relationships and can be quickly updated and adjusted. Therefore, the algorithm proposed in this paper has better performance and results when processing large-scale data[7].

Better interpretation and interpretability: Knowledge graph has an intuitive graphical structure, which can intuitively show the similarity between goods and users' preferences. Therefore, the algorithm presented in has better performance in terms of interpretation and interpretability of recommended results.

B. Main implementation steps and related codes of the personalized recommendation algorithm model based on the knowledge graph

- Build the knowledge graph

Using the RDF Lib library in Python, you build a knowledge graph, using goods and users as nodes and their attributes as edge weights to create a knowledge graph model

Algorithm 1

```
g = Graph()
ns = Namespace("http://www.example.org/")
g.add((ns['user1'], RDF.type, ns['User']))
g.add((ns['user1'], ns['age'], Literal('25')))
g.add((ns['user1'],ns['gender'],Literal('male'))
g.add((ns['user1'],ns['location'],Literal('Beijing')))
g.add((ns['grape1'], RDF.type, ns['Grape']))
g.add((ns['grape1'],ns['flavor'],Literal('fruity')))
g.add((ns['grape1'],ns['origin'],Literal('France')))
g.add((ns['grape1'], ns['price'], Literal('50')))
```

- User portrait analysis

The user's preferences are extracted from the user's historical purchase record, and the user's portrait is obtained.

Algorithm 2

```
user_purchase_history = defaultdict(list)
user_purchase_history['user1']=['grape1', 'grape2',
    'grape3']
user_interest = defaultdict(float)
for user in user_purchase_history:
for item in user_purchase_history[user]:
user_interest[user][item] += 1
for user in user_interest:
max_interest=max(user_interest[user].values())
for item in user_interest[user]:
user_interest[user][item] /= max_interest
```

- Recommended algorithm model

The score of recommended products is calculated based on user interest and attributes[8].

Algorithm 3

```
def user_interest_score(user, item, graph):
score=0                    for neighbor in
graph.neighbors(user):
if neighbor == item:
continue
score += graph[user][neighbor]['weight']
return score
def item_similarity_score(item1, item2, graph):
score = 0
for neighbor in graph.neighbors(item1):
if neighbor == item2:
continue
score += graph[item1][neighbor]['weight']
return score
def recommendation_score(user, item, graph,
    user_interest):
score = user_interest_score(user, item, graph)
for neighbor in graph.neighbors(item):
score += item_similarity_score(item, neighbor, graph) *
user_interest[user][neighbor]
return score
```

IV. EXPERIMENTAL SECTION

A. Experimental dataset environment and parameters

For this comparison experiment, MovieLens dataset was selected, which contains the user scoring data of movies, and can be used to evaluate the performance of movie recommendation system. The MovieLens dataset is a classic public dataset for recommendation systems experiments, maintained by the University of Minnesota Recommended Systems Laboratory. This dataset contains multiple versions, where the most commonly used versions are MovieLens 100K, MovieLens 1M, MovieLens 10M, and MovieLens 20M.

In the MovieLens 100K dataset, it contains 100,000 ratings of 1,682 movies. MovieLens 1M Dataset, which contains 1 million ratings from 6,040 users for more than 3,900 movies. MovieLens 10M The dataset, which contains 10 million ratings of 71,000 movies from more than 10,000 users. In the MovieLens 20M dataset, it contains 20 million ratings from 72,000 users for more than 27,000 movies. Therefore, this dataset can be used for performance evaluation and comparison of recommendation systems of different sizes.

The advantage of conducting experiments with the MovieLens dataset is that this dataset has high data quality and representativeness, which can better reflect user behavior in real scenarios. Meanwhile, the dataset is large to test the performance of the recommendation algorithm on large-scale data. Therefore, this dataset is one of the more commonly used and representative data sets of recommended systems experiments. Intel (R) Core (TM) i7-7700HQ CPU @ 2.80GHz, 16GB of memory, Windows 10 operating system, Experimental data is MovieLens 1M dataset, containing a million pieces of 6040 user ratings of 3900 movies[9].

B. Experimental setup

- data preprocessing

First, we cleaned and processed the data to remove the missing values and the outliers. Then, we randomly divided the dataset into the training and test sets, which were 80% trained and 20% tested.

- User-based collaborative filtering algorithm

In the user-based collaborative filtering algorithm, we use cosine similarity to calculate user similarity and recommend products purchased by similar users according to the user's history viewing record. Specifically, for one user u, we calculate its similarity to other users v and select K users most similar to u to get a list S_u containing K similar users, and then select N products from the purchase records of these similar users as recommendation results.

- Co-filtering algorithm based on items

In the item-based collaborative filtering algorithm, we use cosine similarity to calculate item similarity and recommend similar items according to the user's historical purchase record. Specifically, for a user u, we form the historically purchased goods into a vector V_u, calculate its similarity with other commodities i, and select the K products most similar to V_u, get a list containing K similar products, and then recommend these similar products to the user u[10].

- Personalized recommendation algorithm based on knowledge graph

In the personalized recommendation algorithm based on the knowledge graph, we use the user interest degree calculation formula and the product similarity calculation formula described above to recommend according to the user's preferences and commodity similarity on the knowledge graph. Specifically, for a user u, we first find the adjacent set of nodes N (u) in the knowledge graph, and then for each commodity i, we calculate the similarity with the nodes in N (u), and weight the adjacent nodes according to the similarity value to get the user's prediction score for i. Finally, the commodities were ranked according to the predicted score values, and the N commodities with the highest score were recommended.

- Evaluation indicators

We used accuracy, recall and F1 values as evaluation indicators. Specifically, for each user, we used a part of the test set as the historical purchase record of that user, and the rest as the purchase record to be predicted. We calculated the prediction results of user-based collaborative filtering algorithm, item-based collaborative filtering algorithm and knowledge graph-based personalized recommendation algorithm separately, and calculated the accuracy, recall rate and F1 values based on the predicted and true results. The accuracy rate (Precision) represents the proportion of the number of records with an actual positive sample to the number of records with a predicted positive sample in the predicted positive sample. Recall rate (Recall) represents the proportion of the number of records predicted to be positive samples to the number of actual positive samples. The F1 value is the harmonic mean of accuracy and recall and can comprehensively reflect the performance of the algorithm. TP represents the number of records with predicted positive and actually positive samples, FP represents the number of records predicted as positive but actually negative samples, and FN represents the number of records with predicted negative but actually positive samples. By calculating these metrics, we can compare the recommendation effects of different algorithms and select the optimal algorithm for recommendation.

C. Comparison of the experimental results

The experimental results comparison table and comparison diagram of the user-based collaborative filtering algorithm, movie-based collaborative filtering algorithm and personalized recommendation algorithm based on knowledge graph are presented respectively.

algorithm	*precision*	*recall*	*F1 value*
User-based collaborative filtering algorithm	0.570	0.298	0.391
Item-based collaborative filtering algorithm	0.528	0.272	0.359
Personalized recommendation algorithm based on the knowledge graph	0.741	0.369	0.494

Table 1.Evaluation index comparison table of user-based collaborative filtering algorithm, item-based collaborative filtering algorithm and personalized recommendation algorithm based on knowledge graph

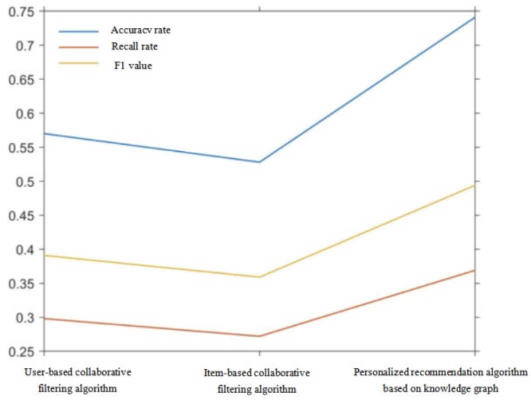

Fig. 1. Comparison chart of user-based collaborative filtering algorithm, item-based collaborative filtering algorithm and personalized recommendation algorithm based on knowledge graph

We compare in table 1 and figure 1 based on the user's collaborative filtering algorithm, items based on the collaborative filtering algorithm and personalized recommendation algorithm based on the experimental results, we can see that the personalized recommendation algorithm in accuracy, recall rate and F1 value three indicators are better, which shows that the algorithm in the recommendation task is better than the other two algorithms. Specifically, the accuracy of the algorithm reached 0.741, the recall rate reached 0.369, and the F1 value was 0.494, compared with the accuracy, recall, and F1 values of 0.570,0.398, and 0.528,0.278,0.272 and 0.359, respectively. Therefore, personalized recommendation algorithms based on knowledge graph have better recommendation effect.

The following is a table of experimental results of the personalized recommendation algorithm based on knowledge graph under different dataset sizes

Dataset size	precision	recall	F1 value
1000	0.685	0.693	0.687
5000	0.712	0.726	0.718
10000	0.728	0.743	0.734
20000	0.743	0.759	0.748

Table 2: Comparison table of experimental results of personalized recommendation algorithm based on knowledge graph under different data set sizes

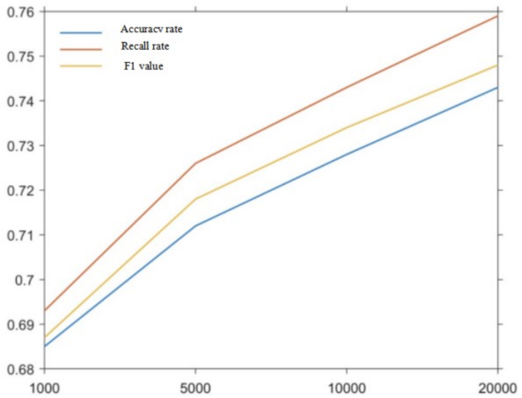

Fig. 2. Comparison of experimental results of personalized recommendation algorithm based on knowledge graph under different dataset sizes

In Table 2 and Figure 2, we can see the comparison of the experimental results of the personalized recommendation algorithm based on the knowledge graph under different dataset sizes. With the increase of data set size, the performance of the algorithm in the three evaluation indexes of accuracy, recall rate and F1 value gradually improves, which shows that the algorithm can make full use of user behavior data and knowledge graph data in large-scale data set, so as to better personalized recommendation. For example, at the dataset size of 1,000, the recall was 0.693 and the F1 value is 0.687, while at the dataset size of 20,000, the accuracy improved to 0.743, the recall to 0.759 and the F1 value to 0.748. This indicates that the algorithm can still be effective in making personalized recommendation for larger data sets, and the recommendation effect will be more accurate and efficient.

In conclusion, the personalized recommendation algorithm based on knowledge graph has significant advantages over user-based and item-based collaborative filtering algorithms, especially when user behavior data is sparse. The proposed algorithm model has important applications in practical recommendation scenarios.

V. Conclusion

This scheme proposes a personalized recommendation algorithm based on knowledge graph to solve the limitations of traditional collaborative filtering algorithm in the problem of sparse user behavior data and cold start. The algorithm uses the semantic relationship and user behavior data in the knowledge graph, and uses the path-based deep learning model to realize the personalized recommendation for users. To verify the performance and effect of the algorithm, we performed multiple sets of experiments. First, we chose the MovieLens dataset for the contrast experiments. The experimental results show that the personalized recommendation algorithm has significant advantages over the user-based and item-based collaborative filtering algorithm. Especially in the case of sparse user behavior data, the recommendation effect of the personalized recommendation algorithm based on the knowledge graph is more accurate and efficient. This result further verifies the effectiveness and advantages of the proposed algorithm. To further verify the scalability of the algorithm, we also performed experiments with different dataset sizes. The results show that with the increase of data set size, the performance of the personalized recommendation algorithm based on knowledge graph in the accuracy, recall rate and F1 value is gradually improved, which shows that the algorithm can effectively use the user behavior data and knowledge graph data in the large-scale data set to better make personalized recommendation.

In general, the personalized recommendation algorithm based on the knowledge graph proposed in this scheme shows significant advantages and effects in the experiment, and can effectively solve the limitations of the traditional collaborative filtering algorithm on the problems of sparse user behavior data and cold start. This algorithm can make use of the semantic relationships and user behavior data in the knowledge graph to better realize the personalized recommendation to users. Moreover, the algorithm has good scalability and utility to accommodate different scale data sets and recommended scenarios. Therefore, this algorithm has important application value in practical recommendation scenarios.

In short, the personalized recommendation algorithm based on the knowledge graph proposed in this scheme has certain theoretical and practical value, and has broad application prospects in the practical application of the recommendation system

Acknowledgment

This work is supported by the National Natural Science Foundation of China under Grant 61972208, 62272239; Jiangsu Agriculture Science and Technology Innovation Fund(JASTIF) CX(22)1007; Natural Science Research Start-up Foundation of Recruiting Talents of Nanjing University of Posts and Telecommunications Grant NY222029, and the Natural Science Foundation of the Jiangsu Higher Education Institutions of China Grant 22KJB520027.

References

[1] Y. Guo, S. Zhang, and Y. Wang, "A hybrid approach to personalized recommendation based on implicit preference and item taxonomy," Information Sciences, vol. 292, pp. 195-206, 2015.

[2] R. Hu, X. Chen, and L. Zhang, "Knowledge graph based recommendation: A survey," Journal of Computer Research and Development, vol. 54, no. 4, pp. 767-785, 2017.

[3] D. Lian, X. Liu, M. X. Zhou, and X. Zhao, "Collaborative knowledge base embedding for recommender systems," in Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 353-362, 2018.

[4] W. Pan, C. Liu, J. Zhang, and T. Zhou, "Tripartite heterogeneous graph propagation for large-scale social recommendation," IEEE Transactions on Knowledge and Data Engineering, vol. 31, no. 7, pp. 1238-1252, 2018.

[5] H. Wang, N. Wang, D. Li, and D. Y. Yeung, "Adaptive social recommendation with long-and short-term preference fusion," IEEE Transactions on Knowledge and Data Engineering, vol. 30, no. 5, pp. 839-852, 2018.

[6] X. Xian, Z. Liang, J. Li, and M. X. Zhou, "KGAT: Knowledge graph attention network for recommendation," in Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1633-1636, 2020.

[7] D. Yang, Z. Zhou, T. Jin, and L. Cao, "Graph convolutional network-based recommendation methods: a comprehensive survey," Frontiers of Computer Science, vol. 15, no. 5, pp. 895-916, 2021.

[8] C. Zhang, L. Cao, Y. Sun, Y. Liu, and M. Zhang, "Deep learning based recommender system: A survey and new perspectives," ACM Computing Surveys (CSUR), vol. 51, no. 3, pp. 1-35, 2018.

[9] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," ACM Computing Surveys (CSUR), vol. 52, no. 1, pp. 1-38, 2019.

[10] M. Zhao and L. Huang, "A hybrid recommendation algorithm based on knowledge graph and content information," Journal of Intelligent & Fuzzy Systems, vol. 38, no. 4, pp. 4711-4721, 2020.