2020 19th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)

Research on items Recommendation Algorithm Based on Knowledge Graph

Pei Liu, HongXing Liu, ChuanLong Li
School of Computer Science and Technology
Wuhan University of Technology
Wuhan, China
1960544079@qq.com

***Abstract*—Traditional recommendation systems mostly use collaborative filtering algorithm, which has problems with cold start and data sparseness. A common idea to solve these problems is to introduce some auxiliary information as input in the recommendation algorithm. The knowledge graph contains rich semantic information, which can provide potential assistance for the recommendation system. The research on the existing recommendation methods based on knowledge graphs found that these methods lacked the consideration of entity attributes information. Therefore, this paper considers attribute factors and proposes the interest modeling method the entity attributes-based in the knowledge graph, and fusions with traditional collaborative filtering algorithm to improve the recommended effect. The results show that the proposed recommended algorithm has better property than other commonly used benchmark algorithms.**

Keywords-knowledge graph; entity attributes; user interest modeling; integration recommendation;

I. INTRODUCTION

The knowledge mapping is a semantic web that reveals this connection between entity and entity, and describes the concepts or entities in reality and their relationships through symbolic forms. Many scholars have already applied knowledge graphs to recommendation fields such as news and music. Reference [1] first used the rich semantic information of knowledge graphs to enhance music recommendation. Reference [2] made music recommendations through the matching of neighbor items based on entities and paths. Reference [3] through combined Node2VEC methods and incremental methods to achieve news recommendation.

Currently, the recommended methods knowledge graph-based include embedding methods representation-based learning and methods meta-based paths. The embedding methods representation-based learning use knowledge representation learning to vectorize entities relationships to obtain low-dimensional dense vector representations of entities and relationships. Then fusion calculation with traditional collaborative filtering to get recommendation list. Reference[4] puts forward a collaborative filtering algorithm clustering-based and singular value decomposition to improve the algorithm effect. The embeddable method shows great flexibility in using the KG-assisted recommendation system, but the Knowledge Graph Embedding algorithms used in methods are generally more applicable to in-graph applications such as link prediction, rather than recommendation. These methods based on meta path provides additional guidance for recommendations by exploring the connection patterns between items in the knowledge graph. Based on meta-graphs,KG is recommended as a heterogeneous information web[5].It contains a variety of node types and edge types. By constructing a meta-path between users or items to make recommendations.

This paper considers that the embedding method based on representation learning usually introducing entity vectors as supplementary information, while users often pay attention to attribute information in entities, so attribute factors will be considered in the recommendation process to improve the recommendation effect. Through the TransR model, the extracted entity triples are vectorized, and the obtained vectors are used to calculate user interest, and knowledge mapping is fused with the traditional collaborative filtering algorithm to complete the recommend.

II. RELATED THEORY

Knowledge representation is to describe knowledge with computer acceptable symbols, that the symbolization process of knowledge. Through learning, the semantic information of entity is converted into low-dimensional vectors to replace the triple calculation with calculating the vector relationship between entities and relationships, which improves the calculation efficiency and results.

The TransE is a model that come up with Bordes et al.[6], using the "translation invariant phenomenon" of word vectors to solve the multi-relational data processing model. (h,r,t) is a triad, r is the relationship between entity h and t .The TransE model regards the relationship r as the distance between entity h and t . Through continuous adjustments h , r and t , make $h+r$ as equal as possible to t , As shown in (1).

$$h+r \approx t \tag{1}$$

Afterwards, the TransR model [7] is proposed as an extension of TransE model and can handle complex relationships. It believes that each entity has multiple attributes, and different relationships correspond to different attributes of the entity. Therefore, different relationships should be embedded in different semantic spaces. Entities should be mapped to the corresponding relationship space before establishing their translation relationship. Where h and t are vector representations in the entity space, h_r and t_r are vector representations of the entities in the relationship space, M_r is a mapping matrix that maps entities from the

2473-3636/20/$31.00 ©2020 IEEE
DOI 10.1109/DCABES50732.2020.00061

entity space to the relationship space r. Their relationship is shown in (2).

$$h_r = hM_r, t_r = tM_r \tag{2}$$

The loss function definition of the TransR model is shown in (3).

$$f_r(h,t) = \|h_r + r - t_r\|_2 \tag{3}$$

III. RECOMMENDATION ALGORITHM BASED ON KNOWLEDGE GRAPH

Text put forward a recommended algorithm knowledge-based graph, which combines semantic information of knowledge graph and user evaluation information to make more accurate recommendation to users. The algorithm model can be seen from Fig. 1.

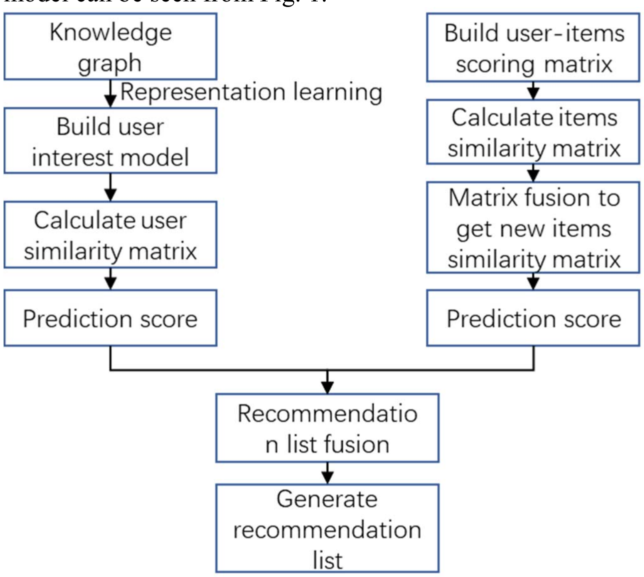

Figure 1. Recommendation algorithm model knowledge-based graph

A. Build interest model

For problems that entity attributes are not considered in the recommendation knowledge-based graph, this text put forward a consumer interest modeling approach entity-based attributes in knowledge graph. The TransR knowledge representation is used to vectorize the attribute triplet of entity to get the vector representation of the triplet, and then the weight of each attribute in the triplet set is calculated and weighted summation to build the consumer-interest models. In this paper, this vector representation of entities, relationships and attributes are obtained by using TransR algorithm, and then calculate the user interest vector. The set of attribute triples is shown in (4).

$$T_u = \{(h,r,a) \mid h \in V_u\} \tag{4}$$

Where V_u is the entity set of user u's historical visit, and T_u is the triplet information contained in the entity set accessed by user u. (h,r,a) is attribute triples that the user has evaluated, h is the entity, r is the relationship, and a is the attribute value of the entity. Through vector representation of entities and attributes, the weight of attribute a in entity h is calculated using (5).

$$w_a = \frac{\exp(h^T r_a)}{\sum_{(h,r,t)\in T_a} \exp(h^T r)} \tag{5}$$

Where h and r_a are vector representations of entities and relationships obtained through representation learning. The user interest model is obtained by weighting and summing all attribute values in the user history evaluation set. The computed process is shown in (6).

$$C_u = \sum_{(h,r,a)\in T_u} w_a a \tag{6}$$

B. Score prediction based on user similarity

The similarity of user's preference for attributes is calculated by the similarity of vector C_u .This text, Euclidean distance is used to calculate user preference similarity, as shown in (7). Equation (8) is used to constrain the calculated value between (0,1] to obtain the similarity between user u 、 v .

$$d(c_u,c_v) = \sqrt{\sum_{i=1}^{k}(e_{ui}-e_{vi})^2} \tag{7}$$

$$sim_{property}(u,v) = \frac{1}{d(c_u,c_v)+1} = \frac{1}{1+\sqrt{\sum_{i=1}^{k}(e_{ui}-e_{vi})^2}} \tag{8}$$

Considering that the preference between users with low similarity has little affect on the recommendation result, but it will affect the calculation efficiency. So a threshold δ is set, when the preference similarity between users is higher than the threshold value, the preference between users is considered to be similar; otherwise, the preference of users is considered different, and the user similarity is set to 0.The user similarity matrix algorithm is shown in Alg.1.

Algorithm 1. The user similarity matrix algorithm

Input: Similarity threshold δ , User interest model Uc ,
User evaluation set U

Output: User preference similarity matrix $\boldsymbol{S}$

Algorithm 1. The user similarity matrix algorithm

for u **in** U **do**
    for v **in** U **do**
        User u's interest vector $c_u \leftarrow U_c$
        User v's interest vector $c_v \leftarrow U_c$
        $Euclidean\ Distance \leftarrow$ Equation (7)
        Preference similarity $sim_{property}(u,v) \leftarrow$ Equation (8)
        if $sim_{property}(u,v) < \delta$ **then**
            $s_{uv} = 0$
        else
            $s_{uv} = sim_{property}(u,v)$
        end
    end **for 循环**
end **for 循环**
return S

Through the obtained user similarity matrix, Equation (9) is used to compute users score for the forecast of all items.

$$p_{ui} = \overline{R_u} + \frac{\sum_{v \in s(u)} sim(u,v) \cdot (R_{vi} - \overline{R_v})}{\sum_{v \in s(u)} |sim(u,v)|} \tag{9}$$

Where p_{ui} represents the predicted score of user u for items i, $S(u)$ represents the user's neighbor set, $sim(u,v)$ represents the user's preference similarity, $\overline{R_u}$ denotes the average score of user u, and $\overline{R_v}$ denotes the average score of user v. Then sort according to the scoring results, giving priority to generate the first N project recommended list.

C. Score prediction based on item similarity

This paper uses item-based collaborative filtering algorithms. Use the user's historical score record to make score predictions for the items not evaluated by users. The item similarity calculation is shown in (10).

$$\mathrm{sim}_{\mathrm{cf}}\left(\mathrm{I_i}, \mathrm{I_j}\right) = \frac{\mathrm{I_i} \cdot \mathrm{I_j}}{\|\mathrm{I_i}\| \|\mathrm{I_j}\|} = \frac{\sum_{k=1}^{m} R_{ki} \times R_{kj}}{\sqrt{\sum_{k=1}^{m} R_{\mathrm{ki}}^2} \times \sqrt{\sum_{k=\mathrm{i}}^{m} R_{\mathrm{kj}}^2}} \tag{10}$$

where R_{ki} represents user k 's evaluation of item j .Through the obtained item similarity matrix, the user's predicted score for the item is calculated. The calculation process is shown in (11).

$$\mathrm{p}_{\mathrm{ui}} = \frac{\sum_{\mathrm{j} \in N(u) \cap S(i,k)} sim(i,j) \times R_{ij}}{\sum_{\mathrm{j} \in N(u) \cap S(i,k)} sim(i,j)} \tag{11}$$

Among them, p_{ui} represents the prediction score of item i by user u, $N(u)$ means the set of items that user u has scored, and $S(i,k)$ means the set of items most similar to item i. By sorting the calculated predicted scores, the first N items are preferentially selected to generate a list of recommendations.

D. Recommendation list fusion

The two parts of the recommendation results are fused to generate the final recommendation list. The fusion process is shown in Alg.2. The main idea of the algorithm is to put the items in the two sets L and E into the Top-N recommended set in turn by traversing. During this process, it is necessary to ensure that the placed object does not exist in the recommended set, to ensure the uniqueness of the objects in the recommendation collection.

Algorithm 2. Fusion algorithm

Input: Collaborative filtering neighbor sets based on knowledge graph and item similarity: $\mathrm{Set\ L} = \{\mathrm{L}_0, \ldots, L_n\}$;
Collaborative filtering neighbor sets based on user preference similarity: $\mathrm{Set\ E} = \{\mathrm{E}_0, \ldots, \mathrm{E_n}\}$

Output: Recommended set $\mathrm{C} = \{\mathrm{C}_0, \ldots, C_k\}$

for i<N **do**
    If $\mathrm{L_i} \notin \mathrm{C}$
        C **append**($\mathrm{L_i}$)
    If Len(C)==k; **break**;
    If $\mathrm{E_i} \notin \mathrm{C}$
        C **append**($\mathrm{E_i}$)
    If Len(C)==k; **break**;
Return Top-N Recommended set C

IV. EVALUATION

A. Data Set

The data set in this paper contains two parts: knowledge graph and user comments. The knowledge graph uses the BMKG published by the Knowledge Engineering Laboratory of Tsinghua University in 2016. The knowledge graph contains more than 720,000 video-related entities, 91 attributes, and more than 13 million triples. The user reviews in this paper are obtained by crawling Douban movie reviews, and got 5,262 users, 1,320 movies, and 152,540 reviews.

B. Evaluation index

This paper uses the correct rate (Pre@k) and recall rate (Rec@k) as evaluation indicators, and k indicates that the user recommends the top k items. The accuracy rate is the proportion of correctly recommended items in the actual recommended in the project; the recall rate is the proportion of correctly recommended items in the actually visited items L_u. L_r represents the list of items recommended by the system. The higher the accuracy and recall rate, the better the performance of the recommendation algorithm. Accuracy eand recall rates are defined as follows:

$$\text{Pre@k} = \frac{|L_u \cap L_r|}{k} \tag{12}$$

$$\text{Rec@k} = \frac{|L_u \cap L_r|}{|L_u|} \tag{13}$$

C. Parameter settings

The similarity threshold δ is the main parameter when calculating the user's preference similarity. This paper uses the change trend of the Pre@5 value to illustrate the influence of parameter δ on the system. It can be seen from Fig. 2 that the recommendation system has the best effect when the similarity threshold δ is set to 0.75.

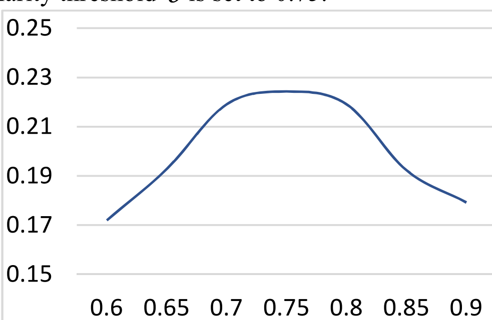

Figure 2. The influence of similarity threshold on Pre@5

D. Experimental results and analysis

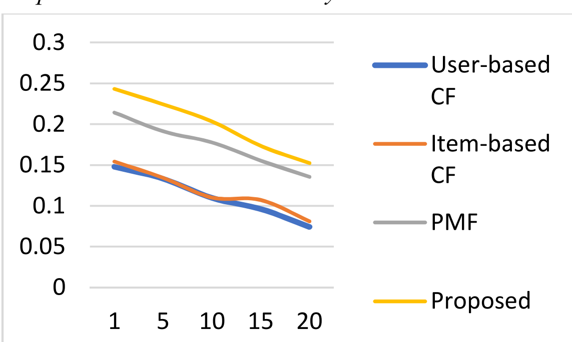

Figure 3. Accuracy rate

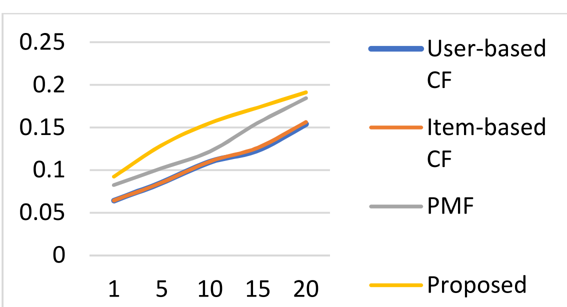

Figure 4. Recall rate

Through iterative experiments on the data set, selecting the parameter values that optimize the experimental results, set the calculation user similarity threshold $\delta = 0.75$, embed dimension $K = 50$. Fig.3 and Fig.4 show the experimental results of relevant algorithms on the data set. The abscissa is the number k of items, ordinate is the accuracy and recall rate.

In Fig.3 and Fig.4, the results of User-based CF and Item-based CF are less effective than the PMF algorithm based on matrix decomposition, indicating that the performance of the matrix decomposition model is generally better than the memory-based collaborative filtering recommendation algorithm in the case of relatively sparse data. Compared with the PMF algorithm matrix-based decomposition, this article puts forward the recommended algorithm in both accuracy and recall rate has increased significantly, indicating that the addition of auxiliary data such as user preference data and attribute semantic information improves the recommendation performance and has a better recommendation effect.

V. CONCLUSION

The main advantages of the recommended algorithm put forward in text are that it introduces the attributes and entity semantics of knowledge mapping in the recommendation process. Using this extra information, a user interest model is established, and then a user preference similarity matrix is constructed to facilitate the discovery of associations between users. At the same time, the article similarity matrix is obtained by fusing the deep semantics of the knowledge mapping. The users and items similarity are used to calculate the recommendation list to complete the personalized recommendation. This results reveal that performance of correct and recall rate of the recommendation algorithm are improved compared to other algorithms, indicating that methods are effective in the recommendation system.

REFERENCES

[1] Sergio Oramas,Vito Claudio Ostuni, Tommaso Di Noia. Sound and Music Recommendation with Knowledge Graphs[J]. ACM Transactions on Intelligent Systems & Technology, 2016, 8(2):21.

[2] W. Hong-xia, "An Improved Collaborative Filtering Recommendation Algorithm," 2019 IEEE 4th ICBDA, Suzhou, China, 2019, pp. 431-435.

[3] Jiangtao Ren, Jiawei Long, Zhikang Xu. Financial news recommendation based on graph embeddings[J]. Decision Support Systems,2019,125.

[4] Z. Wang, Y. Li, L. Fang and P. Chen, "Joint Knowledge Graph and User Preference for Explainable Recommendation," 2019 IEEE 5th ICCC, Chengdu, China, 2019, pp. 1338-1342.

[5] J. Tang and R. Lin, "Ranking-based Collaborative Clustering for Heterogeneous Information Network,"2018 IEEE 18th International Conference on Communication Technology (ICCT), Chongqing, 2018, pp. 246-250.

[6] Bordes A, Usunier N, Garcia-Duran A. Translating embeddings for modeling multi-relational data[C]. Advances in neural information processing systems, 2013: 2787-2795.

[7] Lin Y, Liu Z, Sun M. Learning Entity and Relation Embeddings for Knowledge Graph Completion[C]. Proc of AAAI Conference on Artificial Intelligence. Menlo Park. 2015:2181-2187.