Movie recommendation algorithm based on knowledge graph

Weizhuang Han
Beijing University of Technology
Beijing, China
18811323416@163.com

Quanming Wang
Beijing University of Technology
Beijing, China
wangqm@bjut.edu.cn

Abstract—**The traditional collaborative filtering algorithm only uses the item-user scoring matrix, but does not consider the semantics of the item. The recommendation effect is often not ideal. The connotation knowledge of the movie and the interest preference of the user are added to the knowledge graph, wherein the connotation knowledge is represented by the relationship between the movie entity and its characteristic entity, and the user's interest preference is to use the like relationship and the like relationship between the movie entity and the user entity. Said. After adding the intension knowledge and the user's interest preferences in the knowledge graph, the complex and diverse relationship between the entities is represented in the form of a triple. The knowledge graph is used to represent the learning method, the movie knowledge graph is embedded in a low-dimensional semantic space, the movie entity is vectorized, and then the semantic similarity between the movie entities is calculated, similar to the item-based collaborative filtering. Sexual combination, the film's own connotation information and user preferences in the knowledge graph are integrated into the collaborative filtering for recommendation, which can not only make up for the problem of semantic sparsity, but also can solve the lack of user subjectivity of the knowledge graph because the knowledge graph contains the user's subjective interest preferences. In the knowledge graph, both the objectivity of the connotation knowledge of the film itself and the subjectivity of the user's interest are considered. Compared with the connotation knowledge of the film itself in the knowledge graph, user preferences and object-based items are added to the knowledge graph. The synergistic filtering fusion enhances the recommended effect.**

Keywords—recommendation algorithm, knowledge graph, representation learning, collaborative filtering

I. Preface

In recent years, with the vigorous development of knowledge graph technology and recommendation technology [7], the integration of related technologies in two fields has become a trend. Noia et al. used DBpedia [3], Freebase [4] and other open link data (Linked Open Data) in the recommendation technology, and verified the effectiveness of this method by experimentally calculating its accuracy and recall rate [5]. Literature [6] attempts to use the structural features of the knowledge graph to integrate the ontology into the collaborative filtering algorithm. Literature [7] attempts to combine the knowledge graph representation learning algorithm with the implicit feedback-based collaborative filtering, transforming the original data into a preference sequence for parameter learning, and strengthens the performance of the collaborative filtering recommendation algorithm. Literature [5] uses heterogeneous information networks to represent project and project attribute relationships in knowledge graphs, and uses Bayesian-based collaborative filtering to solve entity recommendation problems. Existing research shows that the knowledge graph representation learning method can embed the knowledge graph into a low-dimensional semantic space, and can use the continuous numerical vector to reflect the structural features of the knowledge graph.

II. Related Theory

Item-based collaborative filtering algorithm [4] (referred to as ItemCF): Recommends users to items that are similar to the items they liked before.

The knowledge graph TransH represents learning: the TransE model proposed by Bordes et al. [1], which is based on the distributed vector representation of entities and relationships in the knowledge graph. TransE is well applied to handle one-to-one relationships, but there are deficiencies in dealing with complex relationships (reflexive, one-to-many, many-to-one, many-to-many). In order to solve the shortcomings of TransE in dealing with complex relationships, Wang Z, Zhang J, Feng J, et al. proposed the TransH [6] model. This paper adopts the TransH model. The core idea is to define a hyperplane W_r and a relation vector d_r for each relationship. $h_\perp, t_\perp$ is the projection of h,t on W_r, where the correct triple is required to satisfy $h_\perp + d_\perp = t_\perp$. This can make the same entity have different meanings in different relationships, and different entities can have the same meaning in the same relationship.

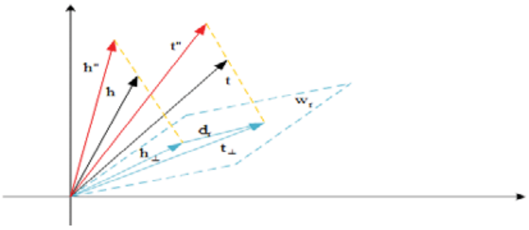

Figure 1. Geometric meaning of the TransH model

As shown in the figure above, for the correct triple (h, r, t) Δ (h, r, t) (Δ (h, r, t) represents the correct set of triples, Δ' (h', r, t) and Δ' (h, r, t') represent incorrect triples), and the relationships that need to be satisfied are shown in the figure. Then for an entity h" if (h", r, t) Δ is satisfied, h"=h is required in TransE, and the constraint is relaxed to h, h" in the TransH algorithm. It is also possible to distinguish h", h, and thus have different representations. The loss function defined in TransH is:

$$f_r(h,t) = \| h - w_r^T h w_r + d_r - t + w_r^T t w_r \|_2^2 \quad (1)$$

w_r is the normal vector of the plane W_r, $w_r^T\ h\ w_r$ which is the projection on w_r. This is because $w_r^T\ h = | w_r^T \| h | \cos\theta$ represents the length of the projection of h in the w_r direction (with sign), multiplied by w_r, that is, the projection of h on w_r , $h -\ w_r^T\ h\ w_r$ is the projection of h on the plane W_r, , $t -\ w_r^T\ t\ w_r$ is the projection of t on the plane W_r. So get the objective function:

$$\min \sum_{(h,r,t)\in S} \sum_{(h',r,t')\in S'} [\gamma + f_r(h,t) - f_r(h',t')]_+ \quad (2)$$

γ is a marginal parameter. S is a positive triple, and S' is a negative triple. The negative triplet S' is obtained by randomly replacing the header entity or the tail entity with any other entity, which can be expressed as

$$S'(h,r,t) = \{(h',r,t \mid h' \in E)\} \mathrm{Y} \{(h,r,t' \mid t' \in E)\} \quad (3)$$

Where: h' and t' represent the replaced head and tail entities, respectively.

Training: The model calculates the corresponding objective function according to formula (2). The positive examples are from the original triples in the knowledge graph, and the negative examples are from the triples generated by the negative sampling algorithm. When updating the parameters in the objective function, this paper uses the gradient descent algorithm to minimize the objective function. The algorithm updates the parameters in the model by iteratively until it converges. After the training is completed, a distributed representation of the entities and relationships in the knowledge graph is obtained, and the semantically similar entities in the knowledge graph are mapped to corresponding positions in the vector space.

The similarity measure between the entity vectors representing the learning embedded in the low-dimensional semantic space is mainly achieved by cosine similarity. Assuming that A and B are two vectors, and n is their dimension, their cosine similarity is expressed as:

$$\mathrm{sim}(A,B) = \frac{A \cdot B}{\| A \| \| B \|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (A_i)^2}} \quad (4)$$

III. Collaborative Filtering Recommendation Algorithm Based On Representation Learning

The basic idea of collaborative filtering recommendation algorithm based on TransH representation learning: using the TransH representation learning algorithm which is better than many complex relationships, embedding the objective connotation knowledge and subjective user preferences of the movie knowledge map into a low-dimensional space to generate the semantics of the item. Matrix, the semantic similarity matrix of the movie is calculated by cosine similarity (4), then the similarity matrix of the item is calculated by the collaborative filtering algorithm, and the semantic similarity matrix of the movie is weighted and combined, and the fusion ratio is adjusted to generate the final list of recommendations. Compared with the connotation knowledge of only the articles in the knowledge map, and ignoring the user interest preferences, the effect of the recommendation is significantly improved.

A. Building a Knowledge graph

Adding film connotation knowledge to the knowledge graph: For a typical knowledge graph, a triplet group represented by the directed graph and a mutual link between the triples form a network of knowledge sets, this triplet carries the semantic information of the entity itself. The entity acts as a node, and the relationship between the entities acts as an edge. The film entity mainly contains the main features of actors, types, directors, screenwriters, etc. These features are also entities, which construct the film knowledge graph through the relationship between the film entity and the feature entity. The relationship between the movie entity and the feature entity mainly includes act_in, director, write, belong_to and their inverse relationship. Using the relationship between the feature entity and the movie entity, a triple of a movie knowledge graph similar to that shown in Figure 2 can be obtained. Thus, in the movie knowledge graph, movie entities with the same characteristics have certain similarities.

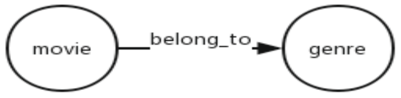

Figure 2 Connotation knowledge triples in the knowledge graph

user
like
movie

Figure 3 User preferences triples in the knowledge graph

Knowledge graphs are added to user hobbies: the user's interest preferences are extracted from the user-score matrix. A movie in which each user scores greater than the average score of the movie that he has evaluated is defined as a favorite, and the user is regarded as an entity in the knowledge graph, and the relationship between the user entity and the movie entity that the user likes is like and liked (Is_liked), the relationship between the user and the movie is also represented as a triple in the knowledge graph, as shown in Figure 3, so that the user's interest in the movie is added to the knowledge graph. Then, the movies that the same user likes have certain similarities. Users who like the same movie have certain similarities, and similar movies that users like have similar similarities. In this way, in the film knowledge graph, there are rich knowledge of the objective connotation of the item, and it has a variety of user interest preferences.

B. Algorithm steps

1) TransH is used to learn to embed the relationship between entities and entities in the knowledge graph in a low-dimensional vector space to generate an entity vector matrix;

2) The item-user rating matrix used to calculate the similarity between the item and the item, and generate the

item-item similarity matrix $R_{m\times n}$.

3) According to the entity vector matrix in 1), the cosine similarity degree (such as Equation 4) is used to calculate the semantic similarity matrix $I_{m\times n}$ of the knowledge graph movie entity and the item-item similarity matrix in 2 $R_{m\times n}$ weighted fusion, the fusion method is as in equation (5);

4) Adjust the fusion ratio, generate a fusion similarity matrix, perform user score prediction, and generate a recommendation list.

Item fusion similarity calculation: Based on the embedded vector of the items in the knowledge graph, the knowledge similarity $sim_{knowledge\_graph}$ based on the knowledge graph is obtained. Based on the user's scoring matrix for the item, the item similarity $sim_{users\_behavior}$ based on the user behavior is obtained. By combining these two similarities, the final similarity of the articles is obtained. The method of fusion is weighting, and the specific calculation method is as follows:

$$sim(A,B)=\alpha\cdot sim_{knowlwdge\_graph}(A,B)+(1-\alpha)sim_{users\_behavior}(A,B) \quad (5)$$

Where is the fusion factor, the value range is [0,1].

IV. Experimental Results and Analysis

A. Data set

The data set of this paper is divided into two parts. One is the movie data crawled from Baidu Encyclopedia. After the data is structured, the knowledge is extracted. The extracted data is saved in the form of triples to construct the knowledge graph. The data includes movie entities and feature entities such as actors, directors, scriptwriters, genres, and the relationship between movie entities and feature entities. The second is MovieLens-1M. The data set mainly includes 1,040 209 scores of 3,900 movies from 6,040 MovieLens users. The data set has two uses. On one hand, user preferences are extracted from the knowledge graph, and the other is used. Aspects calculate the similarity of a movie based on user behavior. After processing, the knowledge graph finally obtained 55,235 entities, 10 relationship attributes and 666,882 triple data.

Data processing: The movie entities built in the Baidu Encyclopedia movie database and the movies in the MovieLens-1M dataset cannot all match. For example, the movie Toy Story (1995) and the movie Toy Story are the same movie, but the Baidu Encyclopedia movie entity and the MovieLens movie name cannot be completely matched due to the release year. In addition, the same movie could not be matched because MovieLens added a foreign name, and the version number was inconsistent. In order to match the movie entities extracted by Baidu Encyclopedia movie database with MovieLens-1M movies, this paper uses the edit distance and string matching's method [3] to map each movie in the MovieLens-1M data set to the knowledge map. After mapping, it finally matched 3,223 movie entity data, which is enough for subsequent work.

B. Analysis of experimental results

The knowledge graph contains the connotation knowledge of the film and the subjective interest preference of the user and the recommendation result of the knowledge graph only contains the connotation knowledge of the film. At the same time, the TransH-CF algorithm proposed in this paper is compared with other algorithms under the two methods. The algorithm includes: an item-based collaborative filtering algorithm (ItemCF), which incorporates TransE to represent a collaborative filtering algorithm (TransE-CF).

The parameters of the experiment are set as follows: the number of common neighbors of Top-k is k=20, indicating that the commonly used learning rate of learning is set to 0.01, and the embedded dimensions are 100, 150, 200, 250, 300. For the fusion similarity weights, the recommendation (10:0) from the full use of collaborative filtering is recommended (10:10) to the full use of the knowledge graph semantic similarity (10:0). For each set of experiments, it was cycled 10 times and averaged.

1) The item-based collaborative filtering algorithm (ItemCF) experimental results are: accuracy rate of 26.01%, recall rate of 17.01%, F value of 0.2056.

2) In the knowledge graph, only the objective factors of the content connotation knowledge are added, and the experimental results of the user's preference subjective factors are not added: the horizontal axis is the embedded dimension. The results show that the experimental results are best when the fusion ratio is 0.6. Therefore, the experimental results with a fusion ratio of 0.6 were selected for comparison of TransH-CF and TransE-CF. As can be seen from the figure, compared with collaborative filtering, the experimental effect is the best when the embedding dimension is 200. When the learning is TransE, the accuracy rate is increased by 1.28%, and the recall rate is increased by 0.58%, indicating that when learning to use TransH, Accuracy increased by 2% and recall rate increased by 0.71%. In comparison, TransH means that learning has a better experimental effect than TransE, but the improvement is still limited.

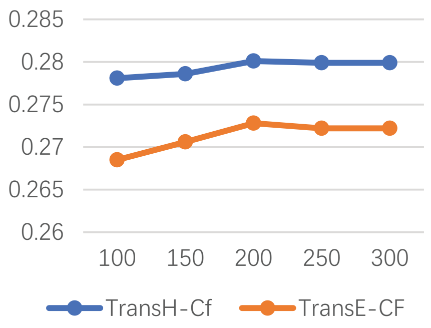

Figure 4 accuracy comparison

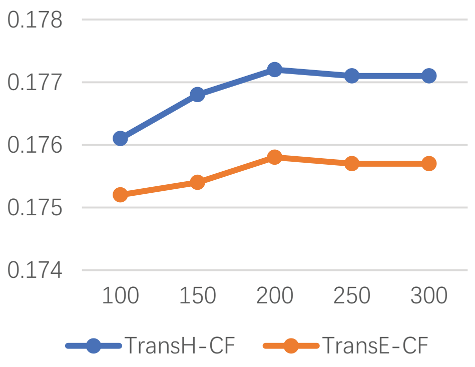

Figure 5 recall ratio comparison

3) The experimental results of adding objective factors of item connotation knowledge and subjective factors of user preference in the knowledge graph: the horizontal axis is the fusion ratio. The experimental results with the best embedding dimension of 200 were selected for comparison of TransE-CF and TransH-CF.

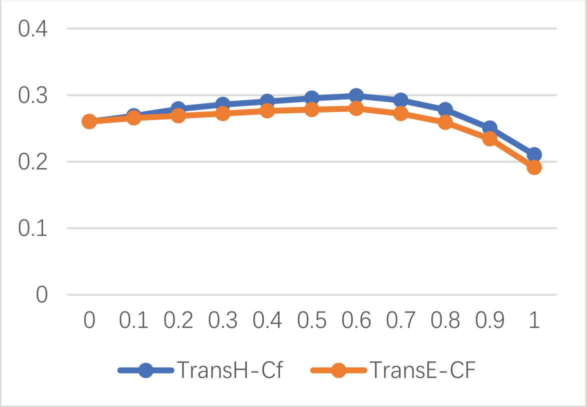

Figure 6 Comparison of accuracy

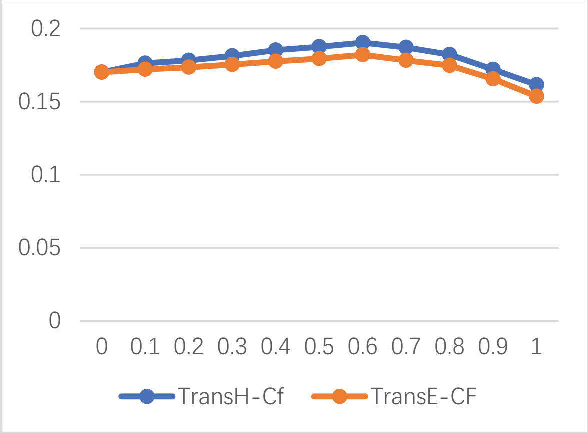

Figure 7 Comparison of recall ratio

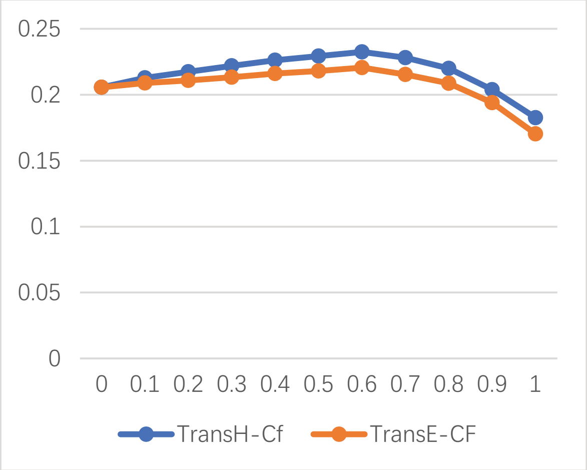

Figure 8 Comparison of F values

It can be seen from the graph analysis that the experimental effect is optimal when the fusion ratio is 0.6. When the learning is TransE, the accuracy rate is increased by 1.98%, the recall rate is increased by 1.19%, and the F value is increased by 0.0149. This indicates that the accuracy of the TransH is increased by 3.87%, the recall rate is increased by 2.02%, and the F value is increased by 0.0269. It can be known that compared with the knowledge only in the knowledge graph, the objective factors of the film connotation knowledge and the subjective factors of the user's preference are added, and the improvement of the recommendation effect is very obvious. However, the experimental results of TransE-CF are still far from the experimental results of TransH-CF. It shows that TransH has obvious advantages over TransE in representing many-to-many complex relationships.

V. Conclusion

The film recommendation algorithm based on knowledge graph proposed in this paper considers both the objective connotation knowledge of the film and the subjective interest preference of the user in the knowledge graph. The TransH representation is used to embed the film entity into the low-dimensional semantic space to generate the semantic similarity matrix. Recommendations are made in conjunction with an item similarity matrix calculated based on user behavior. Compared with the knowledge graph, only the film connotation knowledge and the knowledge graph representation are used to learn to use the TransE model, which significantly improves the recommendation effect. At the same time, users have potential relationships in addition to the behavior of movies. Therefore, how to add potential relationships between users to the knowledge graph is the next step.

References

[1] Yangyong Zhu, Sun Wei. Progress in Recommendation System Research[J]. Journal of Frontiers of Computer Science and Technology. Vol.9, No.5, 2015, p.513-525.

[2] Guoxia Wang, Liu Heping. Overview of personalized recommendation system[J]. Computer Engineering and Applications. Vol.48, No.7, 2012, p.66-76.

[3] Noia TD, Mirizzi R, Ostuni VC. Linked open data to support content-based recommender systems [C]. International Conference on Semantic Systems. ACM, 2012, p.1-8.

[4] Zhang Zijian, Gong Lin, Xie Jian.Ontology-based Collaborative Filtering Recommendation Algorithm[C]. Proceedings of International Conference on Brain Inspired Cognitive Systems.Berlin, Germany, 2013,p.172-181.

[5] Zhang Fuzheng, Yuan NJ, Lian Defu. Collaborative Know ledge Base Embedding for Recommender Systems[C].Proceedings of the 22nd ACM SIGKDD International Conference on Know ledge Discovery and Data M ining.New York, USA, 2016, p.353-362.

[6] Sarwar B, Kartpis G, Konstan J. Itembased Collaborative Filtering Recommendation Algorithms [C].Proceedings of International Conference on World Wide Web. New York, USA, 2001, p.285-295.

[7] Bordes A, Usunier N, Garcia-Duran A. Translating Embeddings for M odeling M ulti-relational Data [C].Proceedings of International Conference on Neural Information Processing Systems. New York, USA, 2013, p.2787-2795.

[8] Wang Zhen, Zhang Jianwen, Feng Jianlin. Know ledge Graph Embedding by Translating on Hyperplanes [C].Proceedings of the 28th AAAI Conference on Artificial Intelligence. Berlin, Germany, 2014, p.1112-1119.

[9] Yu X, Ren X, Sun Y. Personalized entity recommendation: A heterogeneous information network approach [C].Proceedings of the 7th ACM International Conference on Web Search and Data Mining, 2014, p.283-292.