Proceedings of the 6th International Conference on Computing and Data Science
DOI: 10.54254/2755-2721/71/20241656

Research on movie recommendation algorithms based on machine learning

Kai Wang

Taiyuan University of Technology, School of Software, Software Engineering, Taiyuan, 030000, China

kaiwang_tyut@outlook.com

Abstract. In recent years, the number of movies released worldwide has grown exponentially. Due to the large number of movies, it is difficult for users to find movies that match their preferences. Therefore, with the development of Internet technology, it has become an important research direction how to filter out the movies that users are interested in from the massive movie data. This paper mainly focuses on the film recommendation algorithms based on machine learning, including the traditional collaborative filtering algorithm, rating-based sorting recommendation algorithm and content-based recommendation algorithm. By conducting a detailed analysis of the principles, advantages and disadvantages, as well as application scenarios of these recommendation algorithms, this paper aims to identify methods that best fit current movie recommendation systems. The objective is to improve the real-time and personalized recommendation, while providing users with better film recommendation services.

Keywords: Movie recommendations, machine learning, collaborative filtering, recommendation algorithm.

1. Introduction

With the continuous advancement of network technology, the problem of information overload has become increasingly serious. Especially in the entertainment industry, movies play a crucial role and their quantity is also increasing day by day. According to statistics, the number of new movies released globally is rapidly increasing every year, and users often find it difficult to quickly find films that meet their interests when facing such a huge film resource. Therefore, how to help users find suitable films in a short time has become an urgent problem. The movie recommendation algorithm came into being, using artificial intelligence and big data technology to provide users with personalized movie recommendations [1].

Movie recommendation algorithm as a tool for information filtering. At present, collaborative filtering algorithm, rating-based sorting recommendation algorithm and content-based recommendation algorithm are the mainstream technologies. Among them, the collaborative filtering algorithm has become a common choice in film recommendation systems due to its high recommendation accuracy and user acceptance [2]. Through this study, this paper hopes to promote the technological progress of the film recommendation system, improve its practical application effect on major platforms, and provide basic data and theoretical support for subsequent research.

DOI: 10.54254/2755-2721/71/20241656

2. Literature review

2.1. Machine learning

As the core technology of artificial intelligence, machine learning is widely used in film recommendation systems, which greatly improves the accuracy of recommendations and user satisfaction. Machine learning algorithms construct accurate recommendation models by mining users' historical behavior data and movie content features, thereby achieving personalized recommendations. Commonly used machine learning algorithms include the collaborative filtering algorithm, the rating-based sorting recommendation algorithm and the content-based recommendation algorithm.

2.2. Recommendation algorithm

2.2.1. Collaborative filtering. Collaborative Filtering is one of the most classical and widely used algorithms in recommendation systems. This algorithm predicts users' preferences for unknown items by mining the similarity between users and items [3]. Collaborative filtering algorithms are mainly divided into user-based collaborative filtering(UCF) and item-based collaborative filtering(ICF). The UCF first calculates the similarity between users, finds other users with similar interests to the target user, and then uses the historical behavior of these similar users to predict the target user's interest in a project. The ICF predicts by calculating the similarity between items. That is, to find other items similar to the target item and then predict the user's score on the target item according to the user's score on these similar items. The common similarity calculation methods include cosine similarity, Pearson correlation coefficient, and Euclidean distance [4-6]. These three methods for calculating similarity are described below.

Euclidean distance is used to calculate the plane distance between two vectors. However, in multi-dimensional space, with the increase in dimension, the distance between two vectors will become larger and larger, which will lead to a deviation from the Euclidean distance calculation results. The closer the distance, the higher the vector similarity.

$$\text{Distance}(a, b) = \frac{1}{1+\sqrt{(x_a - x_b)^2 (y_a - y_b)^2}} \tag{1}$$

The closer the Distance(a, b) approaches to 1, the greater similarity between vectors a and b.

Cosine similarity is used to calculate the similarity between two vectors. It is usually used in text classification, information retrieval and other fields.

$$\text{Cos}(x, y) = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \sqrt{\sum_{i=1}^{n} y_i^2}} \tag{2}$$

The similarity range is [-1, 1], x_i represents the score of vector x on vector i , y_i represents the score of vector y on vector i .When the result of the calculation is closer to 1, the similarity between the vectors is higher. When the calculation result is closer to -1, the reverse is true.

Pearson correlation coefficient is mainly used to measure the linear correlation between two vectors.

$$\text{Pearson}(x, y) = \frac{\sum_{i \in M_{x,y}} (P_{xi} - \overline{P_x})(P_{yi} - \overline{P_y})}{\sqrt{\sum_{i \in M_{x,y}} (P_{xi} - \overline{P_x})^2} \sqrt{\sum_{i \in M_{x,y}} (P_{yi} - \overline{P_y})^2}} \tag{3}$$

2.2.2. Content-Based recommendation algorithm. Content-based recommendation algorithm is an important branch in the field of recommendation systems. It mainly recommends the most suitable content for users by analyzing users' past behaviors and preferences and combining the characteristics of the content. The algorithm focuses on the processing of content information, and can better meet the personalized needs of users. Usually, it mainly includes the following steps: feature extraction, user portrait generation, similarity calculation, and result recommendation.

DOI: 10.54254/2755-2721/71/20241656

The first is feature extraction. The goal of feature extraction is to extract the core information from the content and express it as a feature vector. Taking a film recommendation as an example, the characteristics of a film can include various metadata such as director, actor, type, release year, score, etc. These features are extracted by text analysis technology, natural language processingtechnology (NLP) , and information retrieval methods. For example, Bag of Words (BoW), TF-IDF, Word2Vec, and other methods can be used for feature extraction of movie description text. Image processing technology can also be used to extract visual features from movie posters. Through these feature extraction methods, the film can be comprehensively described and analyzed.

The process of generating user portraits is to transform users' interests and preferences into feature expressions. By analyzing users' historical behaviors (such as browsing, clicking, collecting, scoring, etc.), user portraits can be generated. For example, users who like sci-fi movies can generate user portraits with sci-fi tags by analyzing the historical data of their viewing and rating. The generation of user portraits helps to accurately grasp the interests and needs of users.

Similarity calculation is at the core of the content-based recommendation algorithm. Common similarity calculation methods include Cosine Similarity, Euclidean Distance and Pearson Correlation Coefficient.

When generating the recommendation results, the algorithm first calculates the similarity between the user portrait and the content to be recommended, sorts according to the similarity, and selects the content with the highest similarity to recommend to the user. This method can avoid the cold start problem to some extent, because the detailed content features help provide reasonable suggestions for new users and new content.

However, the content-based recommendation algorithm also has its limitations. The first is the limitation of recommendation scope, because the algorithm only recommends items similar to the content that users have expressed preferences and lacks diversity. The algorithm relies on high-quality descriptions of content features. When the content description is incomplete or noisy, the recommendation effect will be affected. The content-based recommendation algorithm cannot capture the dynamic changes in users' interests and can only recommend according to the existing data. To improve the above problems, content-based recommendation algorithms can be combined with other recommendation algorithms, such as CF. The CF is based on the historical behavior data of users and recommends by finding other users who are similar to the target user or have similar content, so as to overcome the limitations of the content-based recommendation algorithm [7].

2.2.3. Rating-Based sorting recommendation algorithm. The core idea of rating-based sorting recommendation algorithm is to sort the items according to the user's rating of the items, and then recommend the items that are most likely to meet the user's interest. Compared with CF and content-based recommendation algorithms, rating-based sorting recommendation algorithm are more direct and efficient and can play an important role in large-scale real-time recommendation systems [8]. The recommendation algorithm usually includes the basic steps of data collection, rating prediction, ranking, and recommendation. Among them, rating prediction is the key step. The data collection stage is mainly to collect users' rating data on items, including explicit rating (such as 1-5 points explicitly selected by users) and implicit rating (such as user click, collection, purchase, and other behavior data). When processing these data, the problem of data sparsity is often encountered; that is, most users do not rate most items. In order to deal with data sparsity, the common solution is to use matrix decomposition techniques, such as SVD (singular value decomposition) and MF (matrix decomposition). These decomposition methods can not only perform well in processing sparse data but also capture the potential relationship between users and items.

The rating prediction stage is used to predict the ratings of items that users have not yet rated based on existing rating data. Rating prediction includes a model-based method and a memory-based method. The model-based method, such as the matrix decomposition method, reconstructs the original score matrix by decomposing the score matrix into two low-dimensional matrices. This method can not only deal with sparse data but also capture the potential relationship between items and users. Memory- based

DOI: 10.54254/2755-2721/71/20241656

methods mainly include user item collaborative filtering algorithms, especially the k-nearest neighbor algorithm. This method predicts the score based on the similarity between users or between items, which is simple, direct, and interpretable. However, compared with the model-based method, it has poor scalability and high computational overhead on large-scale data sets [9]. The ranking and recommendation stage is to sort the items according to the prediction score and recommend the items that best meet the user's interests. The commonly used method is top-N recommendation, which is to recommend the n items with the highest prediction score. In the sorting process, a series of evaluation indicators, such as click-through rate (CTR), conversion rate (CR), etc., need to be considered to ensure the quality of the recommendation and user satisfaction.

Rating-based sorting recommendation algorithm utilizes users' historical rating data to provide personalized recommendation services, achieving a dual improvement in the accuracy and efficiency of the recommendation system. Despite facing challenges such as data sparsity and cold start, the rating-based sorting recommendation algorithm will play a greater role in the application of recommendation systems in the future through the combination of model improvement and a variety of technologies.

3. Methodology

3.1. Dataset

The experimental data set used in this study is mainly from movielens. The movielens dataset is a classic dataset for the study of film recommendation systems provided by GroupLens research group at Minnesota University. This paper selected the movielens' data set, which contains about 1,00,000 scoring records of users on about 9700 movies. Specifically, each rating record includes a user ID, a movie ID, a rating (a value of 1-5) and a timestamp. The metadata of the movie includes the movie name, movie ID, and related type information [10].

3.2. Procedure

In order to ensure the scientificity and accuracy of the test data, this paper has carried out detailed design and practice in the process and method of the experiment. The experimental process is mainly divided into five steps: data preprocessing, feature engineering, model selection and training, model evaluation and optimization, and performance evaluation [11].

3.2.1. Data preprocessing. Data preprocessing is an indispensable part of machine learning, which aims to clean up and standardize data to improve model performance. Remove records containing missing values to ensure data integrity. Then, the time stamp is converted to date format, and the scoring data is processed in time segments according to the release year of the film. For the metadata of users and movies, One-Hot encoding is used to convert category data into numerical features. In order to deal with the problem of data sparsity, this study uses standardization technology to normalize the score data to the [0,1] interval by Min-Max standardization.

3.2.2. Feature engineering. In the recommendation system, feature engineering aims to extract and construct as many effective features as possible from the original data so as to improve the prediction ability of the model. User characteristics include user ID, user age, gender, occupation, and other information. Film features include film ID, film type, release year, etc. In the process of feature construction, the average score, score variance, and score preference of popular movies for each user are calculated by considering the user's viewing history. At the same time, interactive features are introduced, including the number of interactions between users and movie types and the number of ratings between users and specific movies. Through the feature selection method, important features are retained, while redundant and irrelevant features are removed.

3.2.3. Model selection and training. This study examined the performance of a variety of machine learning algorithms in film recommendation, including collaborative filtering, content-based

DOI: 10.54254/2755-2721/71/20241656

recommendation algorithm and recommendation algorithm based on rating sorting. In the collaborative filtering algorithm, user-user collaborative filtering and item-item collaborative filtering are implemented, respectively. The content-based algorithm uses TF-IDF technology to calculate the similarity of movie description text and generates recommendation results combined with the user's rating history. During the model training process, the scoring data is divided into a training set and a testing set, accounting for 70% and 30% respectively. In order to prevent the overfitting problem, the k-fold cross-validation technique is applied to train and validate multiple subsets.

3.2.4. Model evaluation and optimization. Validate the model. During the model validation process, cross validation methods are used to deeply integrate and optimize each model. Cross validation methods can effectively avoid the problem of model overfitting. According to the data from the validation set, it is clear that the collaborative filtering algorithm has a prominent lead in the accuracy and effect of recommendations.

3.2.5. Performance evaluation. In the recommendation systems, precision is one of the most commonly used and fundamental measures. It is used to evaluate how many items the recommendation system has in a given recommendation list that users are really interested in. The calculation of precision is based on the ratio of the number of items correctly recommended in the recommendation list to the total number of items recommended.

$$\text{precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{4}$$

TP is the number of items correctly recommended by the system, TN is the number of items correctly not recommended by the system, FP is the number of items incorrectly recommended by the system, and FN is the number of items incorrectly not recommended by the system. Among the evaluation indexes of the recommendation system, recall is a key evaluation index, which evaluates the effectiveness of the model by measuring the proportion of successful recommendations of the recommendation system in the actual positive samples.

$$\text{recall} = \frac{\text{TP}}{\text{TP+FN}} \tag{5}$$

4. Result and analysis

Validate the designed machine learning based movie recommendation algorithm through a series of experiments, and analyze it from several indicators including precision, recall, and coverage. In terms of precision, the UCF has achieved high performance. Specifically, the Top-10 recommendation precision of the algorithm reaches 0.312. The recall rate is 0.0626, which also shows the advantages of the UCF. In terms of coverage, the ICF shows a wider range of recommendations. Figure 1 shows the experimental results of the UCF, after recommending 10 movies, the precision, recall and coverage of the algorithm are calculated, respectively.

This paper continues to change the number of recommendation indicators for further exploration. The following Figure 2-4 are the comparison charts of the precision, recall rate, and the coverage of UCF, the rating-based sorting recommendation algorithm, the ICF, and the content-based recommendation algorithm. The simulation results show that the precision and recall rate of the UCF are better than those of other recommendation algorithms, and has obvious advantages in coverage index. And the recall rate will increase with the increase of user recommendation indicators, and the advantages of the algorithm will become more obvious, which can provide users with more accurate preference indicator recommendations.

DOI: 10.54254/2755-2721/71/20241656

```
Recommending 10 movies...
Splitng dataset...
Done!
train: 70445 test: 30391
Building movie metadata...
Done!
Calculating user similarity matrix...
Calculation done!
For the userId = 3, 10 movies are recommended as follows:
   movieId                                          title  year
0      318                       The Shawshank Redemption  1994
1      858                                  The Godfather  1972
2      593                       The Silence of the Lambs  1991
3      457                                The Fugitive  1993
4     2571                                  The Matrix  1999
5     2858                               American Beauty  1999
6      912                                   Casablanca  1942
7     1225                                     Amadeus  1984
8      110                                   Braveheart  1995
9     1196  Star Wars: Episode V - The Empire Strikes Back  1980
Evaluation start...
Precision = 0.3120
Recall = 0.0626
Coverage = 0.0487
```

Figure 1. Various indicators of user-based collaborative filtering recommendation algorithm

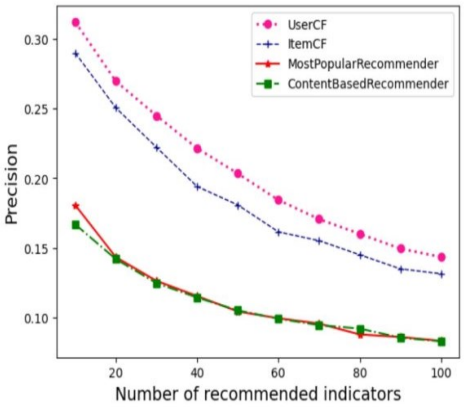

Figure 2. Comparison chart of precision of recommendation algorithms under different recommendation metrics

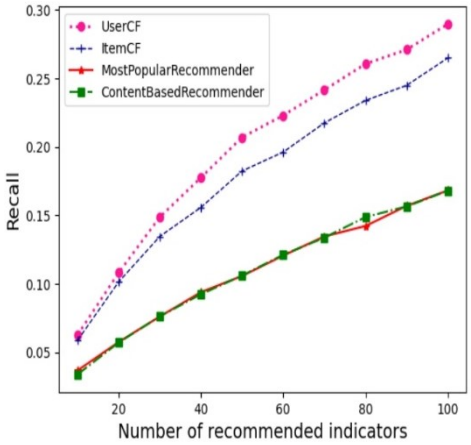

Figure 3. Comparison chart of recall of recommendation algorithms under different recommendation metrics

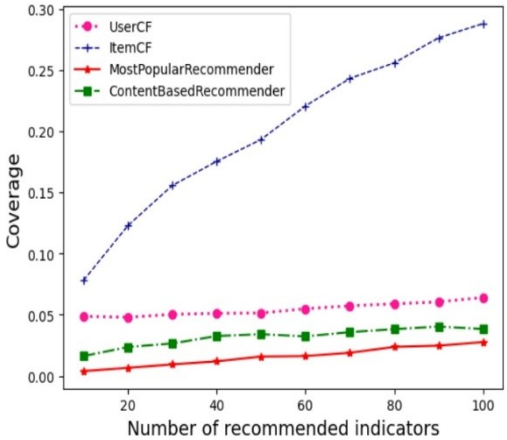

Figure 4. Comparison chart of coverage of recommendation algorithms under different recommendation metrics

The experimental results indicate that when the recommendation index increases, the recall rate, precision, and coverage of the recommendation system will change. The precision has decreased, while the recall rate has increased. Moreover, ICF have obvious advantages in terms of coverage. The results of the experiment also reveal the various performances of the film recommendation system based on machine learning. It shows the advantages and disadvantages of different recommendation algorithms in detail through a variety of evaluation criteria, such as precision, recall, and coverage. In general, collaborative filtering algorithm performs well in many indicators, especially in applications requiring high recall and precision.

5. Conclusion

This paper explored the precision, recall, and coverage of different recommendation algorithms. Through comparison, it is concluded that the three indicators of the collaborative filtering algorithm are higher than those of other recommended algorithms. Although the collaborative filtering algorithm is widely used in major recommendation systems, its problems, such as cold start and sparsity, often lead to poor accuracy of recommended content and ultimately reduce the performance of the system. In the actual use scenario, different recommendation algorithms are widely used with their own advantages, but they also have different problems. The algorithms adopted will be determined according to the

specific application scenario and the actual needs of users. Based on experimental results, collaborative filtering algorithm may be a better solution to achieve better recommendation accuracy and user satisfaction.

In summary, from all the data obtained in this experiment, different recommendation algorithms have their own characteristics for each evaluation standard. User based collaborative filtering methods are suitable for situations with large user groups and relatively stable user interests, while project-based collaborative filtering is suitable for situations with variable product types and relatively small user groups. Therefore, in a real movie recommendation environment, it may be necessary to mix recommendation algorithms according to specific requirements to achieve the best recommendation performance. Hybrid recommendation algorithms can not only solve the shortcomings of single algorithms, but also provide users with more accurate and efficient recommendation services. This discovery provides a key direction for the future development of recommendation systems.

References

[1] Wang Z. A content-based collaborative filtering algorithm for movies and TVS recommendation. Proceedings of the 5th International Conference on Computing and Data Science. 2023:10. DOI:10.26914/c.cnkihy.2023.108612.

[2] Zhang Qi, Yu Shuangyuan, Yin Hong-feng and Xu Baomin. Neural Collaborative Filtering for Social Recommendation Algorithm. Computer Science 2023(02): 115-122.

[3] Zhao J. Research and improvement of movie recommendation based on a collaborative filtering algorithm. Proceedings of the 5th International Conference on Computing and Data Science. 2023:13. DOI:10.26914/c.cnkihy.2023.108744.

[4] Myers L, Sirois M J. Spearman correlation coefficients, differences between[J]. Wiley StatsRef: Statistics Reference Online, 2014.

[5] Benesty J, Chen J, Huang Y. On the importance of the Pearson correlation coefficient in noise reduction[J]. IEEE Transactions on Audio, Speech, and Language Processing, 2008, 16(4): 757-765.

[6] Sedgwick P. Pearson's correlation coefficient[J]. Bmj, 2012, 345.

[7] Kim S, Oh B, Kim M, et al. A movie recommendation algorithm combining collaborative filtering and content information[J]. J Korea Inform Sci Soc Softw Appl, 2012, 39(4): 261-268.

[8] Wang C, Wang H, Pi J, et al. Park recommendation algorithm based on user reviews and ratings[J]. International Journal of Performability Engineering, 2019, 15(3): 803.

[9] Ze W, Dengwen Z. Optimization collaborative filtering recommendation algorithm based on ratings consistent[C]//2016 7th IEEE International Conference on Software Engineering and Service Science (ICSESS). IEEE, 2016: 1055-1058.

[10] Zhang J, Peng Q, Sun S, et al. Collaborative filtering recommendation algorithm based on user preference derived from item domain features[J]. Physica A: Statistical Mechanics and its Applications, 2014(396): 66-76.

[11] Zhang B, Zhang Y, Shen J, et al. Research on differential pulse voltammetry detection method for low concentration glucose based on machine learning model[J]. International Journal of Electrochemical Science, 2024, 19(2): 100479.