

RS on video games based on item-based collaborative filtering algorithm

Hongyun Zhu

Chengdu University of Technology, Department of Software Engineering, No. 1,
Erxianqiao East 3rd Road, Chenghua District, Chengdu, China

zhu.hongyun@student.zy.cdut.edu.cn

Abstract. With the rapid development of the Internet and e-commerce, recommender systems have received great attention and wide application in this environment. Because it is difficult for people to choose the one that they like in the face of the dazzling array of items on the Internet, and these e-commerce sites also need to consider how to improve efficiency, the recommendation system is an excellent solution. This paper mainly reviewed the development of recommender systems, focusing on the research and experiments of a recommender system based on an item-based collaborative filtering algorithm. According to the experimental results and some previous studies, summarizing the advantages and disadvantages of this method, proposing some solutions, and pointing out some problems that will be faced by future researches on recommendation systems.

Keywords: Recommendation system, Collaborative filtering, Item-based Collaborative filtering.

1. Introduction

With the vigorous development of e-commerce, e-commerce platforms such as Amazon, Taobao and JD are becoming more and more popular in people's lives, especially among some young people. Because online shopping has significantly facilitated people's lives, people can browse a variety of products without leaving home. However, the problem also arises. In the face of the dazzling variety of products, users do not know how to choose the products they want or like. The use of recommendation systems in the field of e-commerce can improve this problem. According to users' purchased records, browsing records, and search history and other information from a huge list of products to recommend to users the products they may prefer.

Although recommendation systems play an increasingly important role in the field of e-commerce, most of the existing e-commerce recommender systems can only make recommendations based on a part of the available information, and it is more difficult to provide accurate and effective recommended service in the face of increasingly large data. The collaborative filtering algorithm is a relatively effective recommendation algorithm at present. The e-commerce recommendation model designed by combining this recommendation algorithm and machine learning can make full use of as much information as possible, and collect various types of data to provide consumers with effective referral services.

To incorporate key components of recommendation system and to facilitate data extraction, we categorized this paper into the following six domains: (1) The background and introduction of

recommendation system; (2) The general knowledge of recommendation system; (3) The principle and methods of the experiment (4) Show experimental results; (5) Summarize the experimental results; (6) Analysis of difficulties encountered in experiments and future research directions (7).

2. Literature Review

The concept of the recommendation system was proposed in the mid-1990s. The recommendation system is to recommend corresponding objects to customers according to their needs and preferences, also known as personalized recommendation system [1]. Since the concept of the recommendation system was proposed, it has received a lot of attention with the rapid development of the Internet. As early as 2004, Wang et.al. proposed a recommendation system based on data mining. Data mining is mainly customer-centric to classify customers, and then the system calculates a series of preference values as a recommendation reference based on product ratings, they did a series of experiments on the recommender system in the cosmetics field, and finally, they proposed that the place for improvement is that attributes such as time and season can be added to recommend different products to customers in different seasons [2].

After this, recommendation system became more and more popular, a mushrooming number of advanced recommendation algorithms and designs of recommendation systems have been proposed by researchers. Such as collaborative filtering recommendation, content-based recommendation, population-based recommendation, and utility-based recommendation. Among them, Liu pointed out that the basic idea of the content-based recommendation method is to make reasonable recommendations mainly based on the similarity between the products that the user has purchased in the past and the products they want to buy, rather than information like user ratings [3]. Compared with the content-based recommendation and the utility-based recommendation is easier. Huang has done research on utility-based recommendation and concluded that its essence is to use functions to describe users' preferences, and use functions to describe the preference of each item. Utility is calculated to make recommendations to users [4]. However, the most frequently used and most stable of these recommendation algorithms is collaborative filtering. In 2009, Ma et.al. carried out research on collaborative filtering algorithms. They summarized the research on collaborative filtering algorithms at home and abroad in recent years. Although collaborative filtering algorithms can provide accurate and fast recommendations, they found that collaborative filtering algorithms face two problems, namely data sparsity and real-time recommendation. They pointed out the use of technologies from other fields to solve these problems collaboratively, such as using BP neural network technology to solve the problem of data sparsity, which can improve the recommended speed [5].

The idea of collaborative filtering algorithm can be roughly summarized as finding some similarities through a series of user behaviors, such as clicking on an object, searching for some keywords, etc., and recommending or making decisions for users through the measurement of similarity. The core of collaborative filtering is how to calculate the similarity between items and items as well as users and users.

The collaborative filtering algorithm can easily calculate the similarity between a series of products and products, but the problem is how to use these huge data to make corresponding recommendations to users. In order to solve these problems, researchers have proposed many methods to deal with them. Among them, Fan et.al. made experiments combining a variety of different algorithms with data mining, and the recommendation system combining collaborative filtering and data mining can effectively solve the problems of data redundancy and complexity. Moreover, in 2016, Liu proposed a solution that combines the logistic regression model with the recommendation system to solve the problem of a large number of data samples. Based on its stability and mature online prediction function, it can be applied to large-scale data sets [6]. Another solution is to use machine learning to process this huge and cumbersome data to give users accurate and precise recommendations. IN 2018, Huang et.al. carried out research on the combination of machine learning and recommender systems. They combined the conclusions of many previous researchers and concluded that the combination of deep learning and collaborative filtering can greatly improve the recommendation efficiency, because deep learning can

learn a deep nonlinear network structure to extract deeper features from huge data, such as customer attributes, or commodity attributes, etc. [7]. These are things that a single collaborative filtering algorithm cannot do. This can effectively improve the recommendation richness and accuracy of the recommendation system

3. Methodology

3.1. Preparations

First of all, the implementation and experiments of this recommendation system are carried out under the PyCharm compiler version 2021.3.3, and the processor of the running device is M1 pro. Before experimenting with the recommender system, the selected data set needs to be pre-processed. For example, if the data set in the JSON or CSV format is adjusted to the format required by the file read by the recommender system, the format needs to be adjusted to ISO-8859-1 text files and separate columns with | symbols.

3.2. Methods and theories

The recommendation system is mainly based on the collaborative filtering method to recommend items according to the user's evaluation of the item. The recommendation system consists of 5 functions and the main program. First of all, the main function of the first and second functions is to read the data set file required for the experiment, the first function is to read the file of information about the item, and the second function is to read the file of the user-related information. When fetching files, using spaces and | symbols as separation standards to read the file data into the compiler one by one, which is convenient for later use. In the main program, the absolute path of the corresponding data set file needs to be filled in, so that the compiler can find the location of the file.

The third function is mainly to organize and classify the read data for later use, and convert the data into a dictionary for storing. The dictionary is a storage method for key-value pairs. The advantage is that there is a mapping between keys and values relationship, the corresponding value can be easily accessed through the key. For example, the user id and the evaluated items are stored in a dictionary, and the user's unique id can easily access the user's evaluated items.

The fourth function is the most important part, its function is to calculate the similarity between items. The calculation formula is:

$$\text{similarity}[i][j] = \frac{N_i \cap N_j}{(N_i \times N_j)^{1/2}} \quad (1)$$

i and j represent the id of the corresponding item, $N_i \cap N_j$ represents the number of users who have commented on items i and j at the same time, and N_i and N_j represent the number of users who have commented on 2 items respectively. Then calculating the similarity between all items in turn to establish an Item-Item matrix. The calculation of the Item-Item matrix is not to use the content attributes of the items to calculate the similarity between the items, but mainly to calculate the similarity between the items by analyzing the user's behavior records.

The last function is to sort the items in combination with the user. First of all, to determine the recommended object, you need to fill in the user id in the main program. The idea is to first find the set of items that the user has commented on, and for each item id, assuming that among them for an item $id1$, find the K movies that are most similar to the item, calculating the user's interest in each item under $id1$, and then iterate the entire set of items reviewed by the user, calculate the weighted sum, reorder, and recommend take the first n items, taking the first 10 items here.

4. Result

This experiment is about a recommendation system based on collaborative filtering algorithm, and the data set is about Video Games Sales, which comes from Alibaba Cloud Tianchi. This data set mainly counts the sales of video games in the United States, the European Union, and Japan and the data

provided by user evaluation. In this experiment, the items are the games, and the evaluation is the score given by the user to the game, the item-item matrix is calculated by the score to recommend the game to the user that they may prefer.

The following pictures are some of the results intercepted from many experimental results:

```
/Users/zhuhongyun/PycharmProjects/pythonProject/venv/bin/python /Users/zhuhongyun/PycharmProjects/pythonProject/recommend/itemCFbyMyself.py
user id      recommendation_game
=====
912         Pokemon Diamond/Pokemon Pearl
912         Super Mario Land
912         Super Mario Bros. 3
912         Grand Theft Auto V
912         Grand Theft Auto: Vice City
912         Pokemon Ruby/Pokemon Sapphire
912         Brain Age 2: More Training in Minutes a Day
912         Pokemon Black/Pokemon White
912         Gran Turismo 3: A-Spec
912         Call of Duty: Modern Warfare 3

Process finished with exit code 0
```

Figure 1. Recommended results for user id 912.

```
/Users/zhuhongyun/PycharmProjects/pythonProject/venv/bin/python /Users/zhuhongyun/PycharmProjects/pythonProject/recommend/itemCFbyMyself.py
user id      recommendation_game
=====
12          Call of Duty: Modern Warfare 3
12          Call of Duty: Ghosts
12          Call of Duty 4: Modern Warfare
12          Donkey Kong Country
12          FIFA 16
12          Animal Crossing: New Leaf
12          Mario Party DS
12          The Elder Scrolls V: Skyrim
12          Minecraft
12          Super Mario Kart

Process finished with exit code 0
```

Figure 2. Recommended results for user id 12.

```
/Users/zhuhongyun/PycharmProjects/pythonProject/venv/bin/python /Users/zhuhongyun/PycharmProjects/pythonProject/recommend/itemCFbyMyself.py
user id      recommendation_game
=====
47          FIFA Soccer 13
47          FIFA 17
47          FIFA 12
47          FIFA 14
47          FIFA 15
47          Assassin's Creed III
47          Assassin's Creed
47          Assassin's Creed II
47          Wii Party
47          Pac-Man

Process finished with exit code 0
```

Figure 3. Recommended results for user id 47.

```
/Users/zhuhongyun/PycharmProjects/pythonProject/venv/bin/python /Users/zhuhongyun/PycharmProjects/pythonProject/recommend/itemCFbyMyself.py
user id      recommendation_game
=====
130      Halo 4
130      Halo 2
130      Halo: Combat Evolved
130      Kingdom Hearts
130      Halo 3: ODST
130      Red Dead Redemption
130      Super Mario Sunshine
130      Street Fighter II: The World Warrior
130      Driver
130      Kinect Sports

Process finished with exit code 0
```

Figure 4. Recommended results for user id 130.

```
/Users/zhuhongyun/PycharmProjects/pythonProject/venv/bin/python /Users/zhuhongyun/PycharmProjects/pythonProject/recommend/itemCFbyMyself.py
user id      recommendation_game
=====
620      Super Mario Sunshine
620      Street Fighter II: The World Warrior
620      Driver
620      Kinect Sports
620      Gears of War
620      Metal Gear Solid
620      Metal Gear Solid 4: Guns of the Patriots
620      The Last of Us
620      LittleBigPlanet
620      Destiny

Process finished with exit code 0
```

Figure 5. Recommended results for user id 620.

The experimental results show that the recommendation system can provide more accurate recommendation results, because it is found that the user likes a certain game through calculation, and the recommendation results for the user will appear in the same series of games or sequels of the game. For example, in the user id is 47 recommended results The top 5 recommended games are all FIFA series games.

5. Discussion

From the results of this experiment, it can be seen that the item-based collaborative filtering algorithm recommends by calculating the similarity between games and combining with the user's historical behavior. The accuracy of the recommended results is still quite high. Some experimental results from the result section It is not difficult to see this conclusion, this recommendation system will recommend users to the same series or sequels of the games they have played or scored. However, there is still a problem that some games ranked lower in the recommendation results of some users are not highly similar to the games ranked higher. It is speculated that the reason may be that the number of samples in the data set is not enough. Therefore, not enough similar games to recommend to users.

How to solve this problem, the most intuitive way is to expand the number of users and items collected, which can greatly enrich the types of games that can be recommended to users, but this also leads to a problem that the matrix will become sparse, in the For some large e-commerce websites such as Amazon, their users only comment on 1% to 2% of the millions of books[8], which will cause the rating matrix to become extremely sparse, and it is difficult to find similar user sets, which makes the

recommendation accuracy and the recommendation accuracy better. The efficiency is greatly reduced, and the utilization of data will be greatly reduced.

Kim et al. proposed the ICHM (Item-based Clustering Hybrid Method) method to solve the problem of data sparsity. The recommendation system in this experiment also adopts a similar method [9]. The principle is to use the content information of the items and the users to calculate the similarity between items, and then use the item-based collaborative filtering algorithm for prediction and recommendation, which can avoid in the user-based collaborative filtering system, if two users have the same interests, but two the problem that the connection between the two users disappears when the user does not rate the same item.

In addition, when researching collaborative filtering algorithms, we usually face the problem of cold starts. In this experiment, the main actor faces the problem of new projects, which is not only manifested in the increasing number of new projects, but also in some early items. If no user has commented or rated early items, the similarity between the item and other items cannot be calculated in the item-item matrix, which will result in this item not being recommended. Although some researchers have proposed some methods to solve the problem of cold start of projects, more researchers are needed to work out some effective solutions in the future

6. Conclusion

Recommendation system is a popular model now. In this paper, we mainly review the development process of recommendation system, from the initial content-based recommendation, utility-based recommendation and collaborative filtering recommendation, etc., to the later integration of different algorithms with other fields. A combination of technologies, such as data mining and machine learning. But the collaborative filtering algorithm has always been an effective and stable method for implementing recommender systems. Its basic idea is to use the user's past behavior to predict which items the user might like, such as ratings or historical behavior. This paper studies item-based collaborative filtering, whose mode is similar to ICHM. Through experiments on a public dataset and previous research records of researchers, it is found that the recommender system in this mode can solve the sparsity problem of the system, and also It can improve the recommendation speed and accuracy. However, the cold start problem in the collaborative filtering system has always existed. Some researchers proposed to use the weighted summation method to solve this problem in the system under ICHM mode. Hoping more researchers can suggest more effective methods in the future.

References

- [1] Ren X L and Liu L 2008 Research progress and prospect of recommender system 2008 *China Journal of Information Systems* 2 p 82-90.
- [2] Wang Y F, Chuang Y L, Hsu M H and Keh H C 2004 A personalized recommender system for the cosmetic business *Expert Systems with Applications* 26 p 427-434
- [3] Liu M C 2016 Research on Content-Based Recommendation Technology *Modern Marketing* 6 p 243
- [4] Huang S L 2011 Designing utility-based recommender systems for e-commerce: Evaluation of preference-elicitation methods - *ScienceDirect Electronic Commerce Research and Applications* 10 p 398-407
- [5] Ma H G, Zhang G W and Li P 2009 Survey of Collaborative Filtering Algorithms *Journal of Chinese Computer Systems* 30(7) p 1282-1288.
- [6] Liu Z B 2016 Application of Machine Learning Methods in Personalized Recommender Systems *Information Research* 4
- [7] Huang L W, Jiang B T, Lv S Y, Liu Y B and Li D Y 2018 Survey on Deep Learning Based Recommender Systems *Chinese Journal of Computers* 41(7)
- [8] Sun X H 2005 Research on Sparsity and Cold Start Problem of Collaborative Filtering System (Doctoral dissertation: Zhejiang University)

- [9] Kim B M, Li Q, Kim J W and Kim J 2004 A new collaborative recommender system addressing three problems In Pacific Rim International Conference on Artificial Intelligence. (Berlin, Heidelberg: Springer) p 495-504