International Journal of Computer Science and Information Technology
ISSN: 3005-9682 (Print), ISSN: 3005-7140 (Online) | Volume 4, Number 3, Year 2024
DOI: https://doi.org/10.62051/ijcsit.v4n3.23
Journal homepage: https://wepub.org/index.php/IJCSIT/index

Research on Personalized Recommendation Algorithm of E-Commerce Platform Based on Big Data

Xupeng Gu *

School of Future Technology South China University of Technology, Guangzhou, Guangdong, 511442, China

*Corresponding Author: gu20030419@163.com

ABSTRACT

This study aims to improve the accuracy and user satisfaction of personalized recommendation algorithms for e-commerce platforms. By analyzing the advantages and disadvantages of the existing recommendation algorithms, combined with big data and machine learning technology, this paper proposes an improvement method. During the research process, the parameter configurations of the two recommendation algorithms, SVD and NMF, are optimized using GridSearchCV. The experimental results show that the optimized SVD algorithm outperforms the NMF and the benchmark algorithm in terms of root mean square error (RMSE) and mean absolute error (MAE) indexes, and exhibits high recommendation accuracy. The study concludes that the optimized recommendation algorithm provides more accurate recommendations on e-commerce platforms, enhancing user satisfaction and platform conversion rates.

KEYWORDS

Personalized recommendations; Big data; Machine learning; Collaborative filtering; Matrix decomposition techniques; Grid search

1. INTRODUCTION

With the advent of the big data era, the amount of global information has exploded, leading to severe information overload, particularly in the field of e-commerce. As the number of users and products on e-commerce platforms grows, personalized recommendation systems have emerged. These systems can quickly and accurately analyze each user's product preferences, needs, and purchasing habits, helping users find the products they need amid the vast amount of information, thereby effectively improving user satisfaction and platform conversion rates. However, the existing recommendation algorithms still have many problems in practical applications, such as limited feature extraction application capability and data sparsity problem. This paper aims to study and optimize the personalized recommendation algorithm for e-commerce platforms, by analyzing the advantages and disadvantages of the existing algorithms, and combining big data and machine learning technologies, it proposes an improvement method to improve the accuracy and user satisfaction of the recommendation system.

2. TRADITIONAL RECOMMENDATION ALGORITHMS

In recent years, scholars have categorized recommendation technologies from multiple perspectives, each providing unique definitions and understandings of recommender systems. Currently, traditional recommender systems are divided into three categories [1]: content-based filtering recommendation

(CB) [2], collaborative filtering recommendation (CF) [3] and hybrid recommendation (Hybrid Recommendation) [4], as shown in Figure 1. Filtering recommendation (CF) [3] and Hybrid Recommendation [4], as shown in Figure 1. This paper will not delve deeply into Hybrid Recommendation, which can be summarized as a recommendation approach that combines the advantages of different recommendation techniques while avoiding their disadvantages.

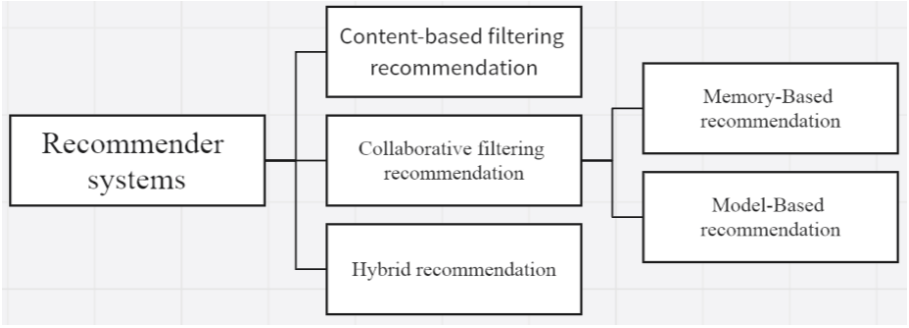

Figure 1. Classification of traditional recommender systems [5]

2.1. Recommendations Based on Content Filtering

Recommender systems were first used in e-commerce websites, which usually recommend items to users with similar preferences to their needs based on their purchase behavior records or purchase reviews [6]. The core idea of content filtering-based recommendation (CB) techniques is to utilize the user's past records of choices or preferences and explore other unknown records of items that are highly relevant to these preferences as recommendations. By analyzing users' explicit feedback (e.g., ratings, likes/dislikes) and implicit feedback (e.g., browsing time, clicking frequency, search behavior, dwell time, etc.), the system obtains records of users' interactions over a specific time period. Subsequently, the system learns the user preferences reflected in these records and transforms them into feature tokens; then it calculates the degree of similarity or match in content between the objects to be recommended and the user preferences; ultimately, the system provides the user with recommended choices that match his/her interests and preferences based on the similarity ranking. The non-negative matrix factorization (NMF) used in this paper is also categorized as a content-based recommendation algorithm; NMF is mainly used to extract features from text data, image data or other types of non-negative data, while it can be used to model the feature vectors of the items in the recommendation system.

The framework of the content-based filtering (CB) system is shown in Fig. 2, which is divided into two major parts: data mining processing and adaptive recommendation, and the user does not need to have direct access to these technical details. The data mining part mainly analyzes and extracts the user's preference features using vector space models. The adaptive recommendation part is responsible for generating recommendation lists based on the similarity ranking of user preferences and recommending these lists to the users through the web server.

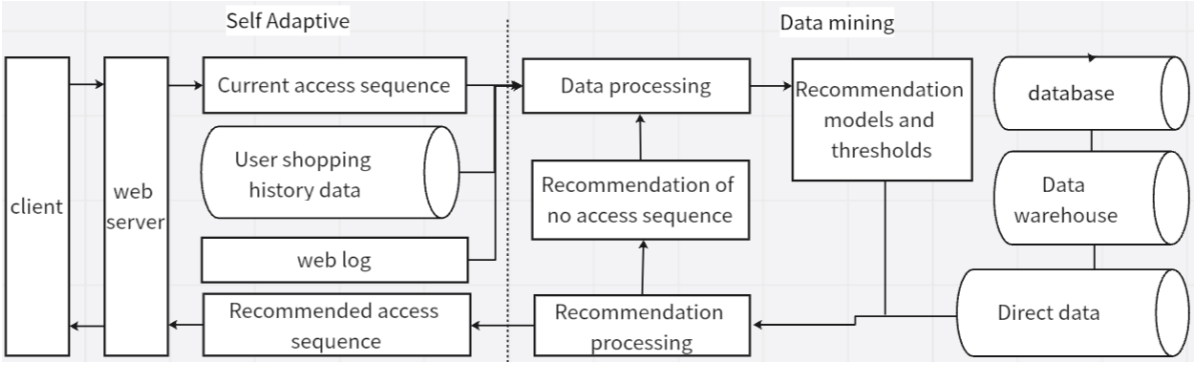

Figure 2. CB system framework [5]

2.2. Collaborative Filtering Based Recommendation

The core of collaborative filtering recommendation (CF) algorithms lies in analyzing the ratings matrix (usually the ratings of users on items) to reveal the interactions between users and items, and thus predicting the correlations between new users and items. As one of the first recommendation techniques to be studied and researched, CF algorithms have significantly contributed to the advancement and application of personalized recommendation systems. In 1992, a paper [7] successfully solved the spam classification problem using traditional collaborative filtering techniques. Amazon, one of the leading online shopping platforms today, extensively uses CF algorithms to recommend products to users. Similarly, Netflix also applies CF algorithm on its main interface to recommend users TV programs they may like. Nowadays collaborative filtering techniques are widely used in music recommendation, movie recommendation, e-commerce, etc. [8], CF is mainly categorized into Memory-Based and Model-Based recommendation.

Memory-based methods directly utilize user-item rating data to calculate the similarity between users or between items, and then make recommendations by similarity. This approach is simple and intuitive, but the computational complexity increases as the data size increases.

Model-based approaches use machine learning and statistical models to predict user preferences, including Probabilistic Matrix Factorization (PMF) [9] and Singular Value Decomposition (SVD) [10]. The main idea of PMF and SVD is to first build appropriate models from historical data records of user-item interactions, and then generate a recommendation list that meets user needs. The main idea is to first build an appropriate model for the historical interaction data records of users and items, and then produce a recommendation list that meets the user's needs. The SVD used in this paper is to reduce, decompose, and compute the user-item ratings matrix into three low-order matrix products by downscaling, decomposing, and training these three low-order matrices and finally restoring them back to the initial matrices.

3. NOVEL RECOMMENDATION ALGORITHMS

With the increasing demand for personalized services, recommender systems are rapidly developing in the direction of diversification and intelligence. The introduction of new recommendation algorithms not only improves the recommendation effect, but also explores how to better combine user behavioral data and advanced algorithmic technology to cope with the increasingly complex market competition and changes in user demand.

3.1. Deep Neural Network Based Recommendation

Deep Neural Network (DNN) is one of the deep learning models [11], which can also be called Multi-Layer Neural Network or Multi-Layer Perceptron (MLP). Currently, there is a growing trend to introduce deep neural network techniques in personalized recommendation problems.

3.2. Knowledge Graph and RNN Based Recommendation

In recent years, with the rapid development of Internet technology, knowledge graph, as a method of structured network describing the relationship between entities in the objective world, has not only had significant applications in search engine optimization, but also begun to show its potential in personalized recommendation systems. For example, the concept of knowledge graph proposed by Google has opened up new ideas for the development of recommender systems. In the literature [12], researchers combined knowledge graph and RNN model to construct a serialized recommendation model that can capture user interest changes in real time. The model effectively embeds the relationship between different data by integrating multi-source heterogeneous data (including graphical data, textual data, and visual data) from music platforms into the knowledge graph, and

utilizes RNN and feed-forward layers to dynamically analyze and recommend predictions of user interests.

Although the model demonstrates its potential mainly in the field of music recommendation, its approach to multi-source data fusion and recommendation efficiency improvement is universal. In large-scale data environments such as e-commerce platforms, similar knowledge graph and multi-source data fusion strategies can be applied to a wider range of product recommendations, thus enhancing the degree of personalization and user satisfaction of the recommendation system. In the future, this model can not only be extended to recommendation in video, text, etc., but also bring more flexible and scalable recommendation solutions for e-commerce platforms.

4. EXPERIMENTAL DESIGN AND ANALYSIS

4.1. Data Set Acquisition and Processing

4.1.1. Data acquisition

First of all, we choose a real e-commerce platform dataset containing user behavior records, and obtain the Ali mobile recommendation algorithm dataset from the "Ali Mobile Recommendation Algorithm Challenge" from AliCloud Tianchi (tianchi. aliyun. com / competition / entrance / 1 / information), which is desensitized Alibaba mobile e-commerce platform data, contains a certain amount of sampled users' mobile behavior data and product subset data within one month (2014.11.18~2014.12.18), including users' ratings of products as well as other key information such as timestamps and geographic locations. These data not only reflect user preferences and behavioral patterns, but also have challenging data sparsity and diversity, which is one of the common challenges in recommender system research. The overall data size is 12256906; the number of users in the dataset is: 10000; the number of products in the dataset is: 2876947; and the number of product categories in the dataset is: 8916.

In order to process the data efficiently, we performed the following preprocessing steps:

4.1.2. Data processing

First, user behavior data and product subset data are loaded and the raw data are converted into a format suitable for the recommender system. User behavior data includes user identification, commodity identification, behavior type, user location, commodity classification and behavior time. In order to simplify the analysis, we selected the four fields of user identification, commodity identification, behavior type and behavior time. Since the behavior type field indicates different behaviors of the user (browsing, bookmarking, adding shopping cart, purchasing), we mapped them to ratings (1, 2, 3, 4, respectively) for subsequent processing by the recommendation algorithm. The product subset data contains three fields: product identification, product location and product classification. In order to ensure the completeness and accuracy of the data, we cleaned and populated the geographic information for the merchandise data, removing the records where the merchandise location is empty.

4.1.3. Data consolidation

We merged the user behavior data with the product subset data based on product identification. The merged dataset contains information such as user identification, product identification, rating, behavior time, product location and product classification. The purpose of this step is to associate the user behavior with the detailed information of the commodity for subsequent recommendation model training.

4.1.4. Data sampling

Due to the large amount of data, we randomly selected 200,000 records from the merged dataset for model training and evaluation. This can reduce the consumption of computational resources and training time while ensuring the representativeness of the data.

4.1.5. Dataset division

We divided the transformed dataset into a training set and a test set with a ratio of 8:2. The training set is used to train the recommendation model and the test set is used to evaluate the performance of the model.

4.2. Experimental Methods and Algorithm Selection

In this study, we have chosen the classical collaborative filtering algorithm (SVD), the recommendation algorithm classified as content filtering based (NMF), and the benchmark algorithm (NormalPredictor) as our experimental subjects to analyze their performances in the personalized recommendation task. These algorithms have different characteristics and applicability scenarios, and through the step of defining parameter grids we define a set of parameter grids for each algorithm. The parameter lattice includes the number of factors, learning rate, regularization parameters, etc. Different combinations of these parameters will be used in the grid search to find the optimal parameters. We used GridSearchCV technique to optimize SVD and NMF their hyperparameters to improve recommendation accuracy and generalization. Finally, we train the final model using the optimal parameters and make predictions on a test set, calculate the Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) as model performance evaluation metrics and compare them with the performance of the baseline benchmark model.

4.3. Experimental Results and Analysis

After training and evaluation, we obtained the following experimental results:

```
Running GridSearchCV for SVD...
RMSE: 0.4397
MAE:  0.1802
Running GridSearchCV for NMF...
RMSE: 0.4816
MAE:  0.2110
Running GridSearchCV for Baseline...
RMSE: 0.5446
MAE:  0.3042
SVD - Best Params: {'n_factors': 20, 'lr_all': 0.002, 'reg_all': 0.02}
SVD - RMSE: 0.4397254402037781, MAE: 0.18017094909873016
NMF - Best Params: {'n_factors': 20, 'n_epochs': 50, 'lr_bu': 0.005, 'lr_bi':
0.005, 'reg_pu': 0.02, 'reg_qi': 0.02}
NMF - RMSE: 0.4816453525979314, MAE: 0.21102453776002494
Baseline - Best Params: N/A
Baseline - RMSE: 0.5446109238546166, MAE: 0.30419627794981885
```

Figure 3. Experimental results

We evaluated the effectiveness of each algorithm in personalized recommendation for e-commerce platforms by comparing the performance of SVD, NMF and Baseline algorithms. The following is a detailed analysis and explanation:

4.3.1. Comparison of model performance

```
SVD - Best Params: {'n_factors': 20, 'lr_all': 0.002, 'reg_all': 0.02}
SVD - RMSE: 0.4397254402037781, MAE: 0.18017094909873016
```

Figure 4. Results of SVD experiments

SVD: Optimal parameters: {'n_factors': 20, 'lr_all': 0.002, 'reg_all': 0.02}; RMSE: 0.4397; MAE: 0.1802

From the above results, we can conclude that the SVD algorithm effectively captures the potential features of users and items through matrix decomposition. It predicts user preferences better and performs the best in this experiment. Its RMSE and MAE are lower than those of other algorithms, indicating high accuracy when dealing with large-scale user behavior data.

NMF:

```
NMF - Best Params: {'n_factors': 20, 'n_epochs': 50, 'lr_bu': 0.005, 'lr_bi':
0.005, 'reg_pu': 0.02, 'reg_qi': 0.02}
NMF - RMSE: 0.4816453525979314, MAE: 0.21102453776002494
```

Figure 5. NMF experimental results

Optimal parameters: {'n_factors': 20, 'n_epochs': 50, 'lr_bu': 0.005, 'lr_bi': 0.005, 'reg_pu': 0.02, 'reg_qi': 0.02}; RMSE: 0.4816; MAE: 0.2110

The NMF algorithm performs second to SVD in this experiment. Although NMF also employs matrix decomposition, its non-negativity constraints may limit its ability to capture complex user-item relationships. As a result, the prediction error of NMF is slightly higher than that of SVD.

Baseline: RMSE: 0.5446MAE: 0.3042

```
Baseline - RMSE: 0.5446109238546166, MAE: 0.30419627794981885
```

Figure 6. Baseline experiment results

The Baseline algorithm (NormalPredictor) predicts based on global mean only and fails to utilize the interaction information of the user and the item, thus its performance is significantly lower than that of SVD and NMF. This suggests that simple global mean approach is not sufficient to provide accurate personalized recommendations in recommender systems.

4.3.2. Effect of parameter optimization

The optimal parameter settings for the SVD are {'n_factors': 20, 'lr_all': 0.002, 'reg_all': 0.02}. These parameters effectively balance the model's complexity and generalization ability, resulting in the lowest prediction error.

The optimal parameters of NMF are {'n_factors': 20, 'n_epochs': 50, 'lr_bu': 0.005, 'lr_bi': 0.005, 'reg_pu': 0.02, 'reg_qi': 0.02}. Although these parameters improve the performance of the model, NMF is still slightly less effective in handling sparse data compared to SVD.

4.3.3. Analysis of experimental results

Experimental results show that optimizing the parameters allows the SVD algorithm to achieve the best recommendation accuracy, significantly outperforming the NMF and Baseline algorithms. This means the SVD algorithm can provide users with more accurate product recommendations, improving user satisfaction and platform conversion rate.

Algorithm selection and application: In the practical application of e-commerce platforms, it is crucial to select appropriate recommendation algorithms. Based on the results of this experiment, SVD algorithms should be prioritized, especially in scenarios where large-scale data need to be processed and recommendation accuracy needs to be improved.

The NMF algorithm, although slightly underperforming in this experiment, still has its advantages in specific application scenarios (e.g., non-negative data processing).

4.3.4. Summary

Through comparative experiments of the SVD, NMF and Baseline algorithms, we found that SVD algorithm performs best in handling personalized recommendations on e-commerce platforms and

significantly improves the accuracy of recommendations. The optimized recommendation algorithm is able to provide more accurate recommendations on e-commerce platforms and improve user satisfaction and platform conversion rate. Future research can further explore other advanced recommendation algorithms, such as hybrid recommendation models combining deep learning and knowledge graph, to further improve the performance of recommendation systems. In addition, attempts can be made to test and optimize the algorithms under larger datasets and more complex user behavior patterns to verify their generalization ability and practical application effects. Future work will continue to explore more advanced recommendation techniques to further enhance the performance and user experience of recommendation systems.

REFERENCES

[1] VERBERT K, MANOUSELIS N, OCHOA X, et al. Context-aware recommender systems for learning: a survey and future challenges [J]. IEEE Transactions on Learning Technologies, 2012, 5(4): 318-335. 10.1109/TLT.2012.11

[2] MOONEY R J, ROY L. Content-based book recommending using learning for text categorization [C]//Proceedings of the 5th ACM Conference on Digital New York: ACM, 2000: 195-204. 10.1145/336597.336662

[3] BREESE J S, HECKERMAN D, KADIE C. Empirical analysis of predictive algorithms for collaborative filtering [C] // Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence. San Francisco: Morgan Kaufmann Publishers Inc. 1998: 43-52.

[4] BALABANOVIĆ M, SHOHAM Y. Fab: content-based, collaborative recommendation [J]. Communications of the ACM, 1997, 40(3): 66-72. 10.1145/245108.245124

[5] Yu Meng, He Wentao, Zhou Xuchuan, Cui Mengtian, Wu Keqi, Zhou Wenjie. A review of recommender systems. Computer Applications [J], 2022, 42(6): 1898-1913 DOI:10.11772/j.issn.1001-9081.2021040607

[6] LIU L W, LECUE F, MEHANDJIEV N. Semantic content-based recommendation of software services using context [J]. ACM Transactions on the Web, 2013, 7(3): No.17.10.1145/2516633.2516639

[7] GOLDBERG D, NICHOLS D, OKI B M, et al. Using collaborative filtering to weave an information tapestry [J]. Communications of the ACM, 1992, 35(12):61-70. 10.1145/138859.138867

[8] CAI Y, LEUNG H F, LI Q, et al. Typicality-based collaborative filtering recommendation [J]. IEEE Transactions on Knowledge and Data Engineering, 2014, 26(3): 766-779. 10.1109/tkde.2013.7

[9] SALAKHUTDINOV R, MNIH A. Probabilistic matrix factorization [C] // Proceedings of the 20th International Conference on Neural Information Processing Systems. Red Hook, NY: Curran Associates Inc. 2007: 1257-1264. 10.1145/1390156.1390267

[10] Funk S. Funk-SVD [EB/OL]. (2006-12-11) [2020-11-01]. 10.33268/met.2020.6.4

[11] COVINGTON P, ADAMS J, SARGIN E. Deep neural networks for YouTube recommendations [C] // Proceedings of the 10th ACM Conference on Recommender Systems. new york: acm, 2016: 191-198. 10.1145/2959100.2959190

[12] HUANG J, ZHAO W X, DOU H j, et al. Improving sequential recommendation with knowledge-enhanced memory networks [C] // Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM, 2018: 505-514. 10.1145/3209978.3210017