Scalable Computing: Practice and Experience, ISSN 1895-1767, http://www.scpe.org
© 2024 SCPE. Volume 25, Issues 4, pp. 2645–2655, DOI 10.12694/scpe.v25i4.2938

RESEARCH ON COMPUTER INTELLIGENT COLLABORATIVE FILTERING ALGORITHM FOR PERSONALIZED NETWORK DATA RECOMMENDATION SYSTEM

YONG YU *

Abstract. In order to meet the protection needs of user privacy data in social networks, this paper proposes a computer intelligent collaborative filtering algorithm for personalized network data recommendation systems. This algorithm predicts user preferences for specific items by utilizing user evaluation information on groups of similar feature items, thereby achieving personalized recommendations. The experimental results show that as the number of project feature selections N increases, the MAE, RMSE, and NDCG5 of the algorithm gradually improve. This is mainly attributed to increasing the number of features under a fixed similarity threshold, which makes the data granularity finer and helps to describe project features more accurately. In the case of a fixed number of project feature selections N, the impact of the number of nearest neighbors s in similar groups on algorithm performance was further studied. The results showed that with the increase of s, MAE, RMSE, and NDCG5 showed a decreasing trend. Although the algorithm suffers from certain losses in recommendation accuracy, it is still within an acceptable range. It is worth noting that due to the system only using generalized data as input, user privacy data is effectively protected. Based on the comprehensive experimental results, this algorithm has significant value in practical applications.

Key words: Social networks, Collaborative filtering recommendation algorithm, User privacy data, User similarity

1. Introduction. With the development of the times, the internet has gradually entered the daily life of the public, and obtaining information through the internet has become routine [1]. In recent years, internet data and information have shown an exponential explosive growth, with massive amounts of information flooding the internet, making it increasingly difficult for users to find the information they need. As an information demander, finding what one needs and is interested in from a large amount of information is often not an easy task; For information providers, it is also very difficult to make their information accessible to everyone. The recommendation system is the main tool to solve this contradiction.

The extraordinary ability and practical value demonstrated by recommendation systems in dealing with information overload have increasingly attracted the attention of the academic community. In recent years, research on recommendation systems has made rapid progress, with significant progress in recommendation effectiveness and accuracy. However, there are still some unresolved issues, such as data sparsity and cold start in collaborative filtering algorithms, in order to address these issues and improve recommendation accuracy as much as possible, it is necessary to consider integrating other information into collaborative filtering algorithms. Due to the widespread development of social networks, user social network information has been applied to many websites. As early as the last century, scholars began researching social network information. Nowadays, with the rapid development of the Internet, it has become a research hotspot. The current research on social network analysis has achieved significant results in data mining tasks such as link based node sorting, node classification and clustering, link prediction, and subgraph discovery [2,3]. However, in the research of recommendation systems, there are not many mature algorithms that apply social network analysis to assist in recommendation system research. One of the author's research topics is to consider using social network analysis methods and combining machine learning with collaborative filtering algorithms in recommendation systems to form a new type of effective and practical algorithm. On the other hand, the personalized recommendation methods currently used in internet applications are based on collecting user personal information and behavioral data. Although these internet applications anonymously process user data when personalized recommendations are made to users, the original data of user information is often retained in the databases of various internet

*Henan Institute of Economics and Trade, Zhengzhou, Henan, 450001, China (yuyong_806hs@163.com).

companies, due to some vulnerabilities in various systems, incidents of user data leakage often occur. This may cause incalculable losses to users and businesses.

2. A User Privacy Data Protection Recommendation Algorithm Based on Project Feature Grouping.

2.1. Privacy issues in recommendation systems. In a recommendation system, recommendation effectiveness and user privacy data protection are a contradiction. In order to achieve better recommendation effectiveness, it is necessary to obtain sufficient personal and behavioral information of users, which is the risk of exposing user privacy issues [4]. The main ways to expose user privacy information in personalized recommendation are as follows: The server in the personalized recommendation system automatically obtains and monitors user behavior information; The recommendation system has been attacked by network hackers, and user information has been leaked; Users submit their own privacy information independently; Resale of user private information; Among them, automatic monitoring of user behavior information is the most common and covert.

In personalized recommendations, the user's privacy behavior information mainly includes the following information:

Personal information: Mainly includes information submitted by users during registration or subsequent self submission. These include name, gender, birthday, email, interest preference, work, location, and even ID card number. These are personal privacy information for users [5,6]. Personal behavior information: mainly includes user access frequency, time, messages posted by users on social networks, @ friends, comments, search records, etc. Personal customization information: This includes users customizing their own homepage, profile, etc. according to their own habits. Currently, it is not uncommon for news to be leaked due to ineffective protection of user data, and users are increasingly paying attention to their privacy protection issues. Therefore, various privacy protection technologies have emerged. Common privacy protection technologies mainly include cryptography based techniques, anonymity, and data perturbation.

Cryptography based technology is a relatively traditional privacy protection method, mainly applied to the security protection of information data transmission processes in centralized or distributed scenarios. Among them, applications in distributed environments are the most widespread, such as distributed privacy queries, distributed data mining, etc. This technology typically requires communication points to be trusted and secure. The commonly used methods in cryptography based technologies include secure multi-party computation (SMC) homomorphic encryption algorithms, etc. Both secure multi-party computation and homomorphic encryption algorithms require complex cryptographic computation support. Due to the extremely large number of users and information in recommendation systems, cryptography based methods are not suitable for privacy protection in recommendation systems.

K anonymity technology, as the name suggests, is to remove some identifiers and directly hide the user's identity information, in order to prevent locating the user through the published information and obtaining further privacy information from the user. The technical principle of this anonymous method is simple, but there are also some security issues, such as its inability to effectively resist link attacks [7]. Link attack refers to the use of other relevant resources by users to locate relevant users after data is published. There are two main ways to achieve anonymity: generalization and removal. These two methods can effectively protect users' privacy data. The basic idea of generalization is to replace an attribute value with a universal value or interval. The basic idea of removal is to protect user privacy by removing sensitive information. The application of removal technology has led to a reduction in data in the data table. Anonymous technology in k often combines generalization and removal simultaneously. For particularly sensitive information, direct deletion is adopted, while others are generalized.

Data interference technology is the most fundamental privacy protection technology. Data interference technology is the use of certain algorithms to modify and interfere with raw data in order to protect user privacy data. Even through published information, raw data and statistical information cannot be obtained. Common data interference techniques mainly include data cleaning, randomized interference, etc [8,9]. The research goal of data interference technology is to ensure the accuracy of data mining under the condition of reaching a certain level of data interference, so that the accuracy of data mining results is infinitely close to the original data mining results without interference. Random interference technology is the most common

technique in data interference technology. It hides data by adding other noisy data to the original data, and the main methods of adding noisy data include additive and multiplicative addition. Although random interference technology has changed the original data, it can still obtain data features that are similar to the original data. Data cleaning and data exchange are also important data interference technologies. Data cleaning mainly reduces the support of frequent itemsets by removing or modifying data. And data exchange technology hides data by exchanging partial data in the dataset.

2.2. Establishment and analysis of recommendation models. In traditional personalized recommendation systems that rely on weights, there is a drawback of users' privacy information being easily leaked [10]. Because personalized recommendation systems need to obtain user preference information to construct user models, obtain user interest preferences, and recommend content information that interests them to users, recommendation systems need to collect detailed information about user behavior to analyze user interests. In a weighted network, user information can easily be fully exposed to the recommendation system through weights, such as obtaining the user's purchase record through the user's rating records. With the increasingly serious issue of privacy exposure, most users are concerned about the protection of their privacy information and therefore unwilling to provide private or sensitive information, therefore, in order to provide better personalized services and meet user concerns, current recommendation systems must provide an effective mechanism for protecting user privacy data.

Currently, many studies are committed to effectively protecting user privacy data without compromising recommendation accuracy [11]. In the personalized recommendation industry, collaborative filtering is a widely used and mature technology that has been applied in many practical application systems. The commonly used privacy protection technologies for collaborative filtering recommendations include encryption based technology, k anonymity technology, and random perturbation technology. Encryption based technology is mainly applied to privacy protection data mining research in distributed data storage. K-anonymity technology achieves privacy protection by generalizing and hiding user data, while random perturbation technology is commonly used in centralized data storage. In a recommendation system, any item can be divided into different groups based on its characteristics. For example, for music, it can be divided into corresponding groups based on the characteristics and attributes of the singer, style, and so on. The user's evaluation of a specific project can be inferred from the user's evaluation of the group in which the project is located. Based on the user's interaction behavior with these groups, it can to some extent reflect their preference for a specific project. And the division of these groups needs to be determined by the feature division of the project, each project has its own feature information and exists as an attribute of the project itself. These feature information implicitly contain user preference information for a project, because when a user makes a selection or evaluation of a project, their evaluation of projects with similar project features should be similar. Therefore, it can be considered to partition groups based on project features for recommendation. As the recommendation is only based on the feature information of the project itself and the evaluation information of the user's project feature group, without collecting user privacy information, it achieves the protection of user privacy data at the root.

The traditional item based collaborative filtering recommendation algorithm (1tem based Collaborative Filtering) predicts a user's rating for a target item based on their rating for similar items. It is based on the assumption that if most users have similar ratings for certain items, the current user's rating for these items is also similar [12,13]. The project-based collaborative filtering recommendation system uses statistical methods to find several sets of nearest neighbors of the target item. Due to the similarity between the current user's rating of the nearest neighbor and the target item's rating, the current user's rating of the target item can be predicted based on the current user's rating of the nearest neighbor, thus achieving recommendation. Similar to the analysis of user based collaborative filtering algorithms, it can be seen that the key to traditional project-based collaborative filtering recommendation algorithms lies in the determination of the nearest neighbor set of the project. The calculation of nearest neighbors cannot be separated from the measurement of similarity and the calculation of nearest neighbors. However, traditional project similarity calculation methods, whether they have Pearson correlation similarity, cosine similarity, or modified cosine similarity, are difficult to obtain accurate results when the evaluation data is very sparse. Similar to the improvement of user based collaborative filtering in Chapter 3, which takes into account the user's feature information, while considering the protection of user privacy data, the system needs to collect as little personal privacy information as possible. Therefore,

starting from the analysis of project feature information, a collaborative filtering algorithm based on project feature model is adopted. We have comprehensively considered the project's own attributes and user evaluation information. Firstly, a project feature similarity matrix is constructed based on the attribute feature vectors of the project, and combined with the existing user project rating matrix, the nearest neighbor project group set about the project is obtained. Furthermore, the filling matrix is predicted to obtain the predicted score and ultimately achieve recommendation.

The process of establishing a project feature similarity model mainly includes quantifying project feature data, calculating project feature similarity, and forming project neighbors based on feature similarity. Vector representation is generally used to quantify the feature attributes of a project, for example, dividing a project set into n mutually independent features, and forming an n-dimensional vector representing n important feature attributes of a specific project as $(c_{i1}, c_{i2}, ..., c_{in})$, among them, c_{in}represents the characteristic value of the nth attribute of the item i [14]. The eigenvalues of attributes can be numerical or descriptive, and different similarity calculation methods should be used for different types of eigenvalues. Numerical data can be divided into two forms: fixed values and interval ranges. For the form of interval ranges, the proportion they occupy can be used for calculation. Descriptive data can be represented using a commonly used TF-IDF formula to calculate the frequency of keywords appearing in the text as eigenvalues. The calculation formula for project feature similarity is:

$$sim(i,j) = \frac{|n_{ij}|_C^{t_i=t_j}}{|n_{all}|_C - |n_{ij}|_C^{t_i \neq t_j}} \tag{2.1}$$

Among them, t_i, t_j represents the feature groups of project i and project j, C represents all feature sets, and $|n_{ij}|_C^{t_i=t_j}$ represents the number of identical features that project i and project j have, $|n_{all}|_C$ represents the number of all features, and $|n_{ij}|_C^{t_i \neq t_j}$ represents the number of features that neither item i nor item j has. After calculating the similarity, the process of establishing the project feature similarity model is completed. In response to the problem of inaccurate calculation in traditional similarity measurement methods when the user rating matrix data is extremely sparse, a novel similarity measurement method is used to calculate the similarity between items. The main idea is that if you want to calculate the similarity between project i and project j, first count all users who have evaluated these two projects. Users can evaluate either one of them or both at the same time, if $r_{ui} = 0$ here indicates that user u did not evaluate project i, then these user sets are recorded as:

$$U_{ij} = \{u | u \epsilon U \cap (r_{u,t} \neq 0 || r_{u,j} \neq 0)\} \tag{2.2}$$

Then, pre score the items that have not been evaluated by users based on the scores of items in the project feature similarity group; Finally, the Pearson correlation coefficient method is used on the user set U_{ij} to calculate the similarity between terms i and j, and ultimately form the nearest neighbor of the target term. Among them, the rating of all users u on item i in U_{ij} is:

$$R_{ui} = \begin{cases} r_{ui}, r_{ui} \neq 0 \\ p_{ui}, p_{ui} = 0 \end{cases} \tag{2.3}$$

When user u has evaluated project i, the rating is equal to their actual rating r_{ui}. When user u has not evaluated project i, the rating is the predicted value p_{ui} obtained based on the rating of project i's characteristic neighbors. The calculation formula for the predicted value p_{ui} is:

$$p_{ui} = \frac{\sum\limits_{j \epsilon C_i} sim(i,j) * r_{uj}}{\sum\limits_{j \epsilon C_i} |sim(i,j)|} \tag{2.4}$$

Among them, C_i is the characteristic neighbor of project i. After the above processing, the data sparsity of items i and j has been alleviated, and the user set for joint evaluation of the project has increased. This can

ensure the accuracy of the nearest neighbor set of the calculated items [15]. In addition, for the analysis of algorithm efficiency, considering that the feature model of the project can be established offline, the calculation of feature similarity between each project can also be done offline, and the calculation results can be saved in the database. Therefore, the above prediction process will not have a significant impact on the operational efficiency of the recommendation system. The author's recommendation model is based on the feature matrix of users, projects, and groups with similar project features, and uses the calculated feature vectors to predict users' predictive ratings for specific projects. Although the graph model of the method proposed in the previous chapter is similar, their research objectives are not the same. The focus of the method proposed in the previous chapter is to combine social network analysis methods and fully utilize user social network information data, with the aim of improving recommendation accuracy; Although the recommendation model in this chapter also considers the direct relationship between users and projects, it focuses more on protecting user privacy while implementing recommendations. The author's algorithm can be used as an anonymous algorithm from the perspective of privacy protection calculations. We know that the classic k-anonymity algorithm is based on the construction of attribute hierarchy, and then the hierarchical structure of this construction is generalized to achieve the purpose of anonymity protection privacy. There are various ways to construct the hierarchy of k-anonymity algorithms when applying them, but obtaining the optimal k-anonymity is an NP challenge, which is also a major obstacle to the application of k-anonymity algorithms. For the author's recommendation model, it is a hierarchical structure, and its generalization method does not have strict limitations. Just like music, projects can be generalized based on the style of the music or the performers of the music. Based on this premise, the granularity selection of generalization will ultimately determine the implementation effect and recommendation accuracy of the recommendation model. If the granularity of generalization is coarse, the less user privacy data information will be included in the feature partition in the project group. This is helpful for protecting user privacy data, but it will have a negative impact on the accuracy of the system's implementation effect; On the contrary, if the granularity is too fine, although the accuracy of the system is improved, user information will become easily exposed, which clearly violates the principle of protecting user privacy data. So in practical model applications, it is necessary to constantly try to change the granularity division and find a balance point that effectively protects user privacy data with good implementation results. In addition, we do not actually generalize the recommended main users. On the one hand, this is because user privacy protection is something we must pay attention to, so we should try to obtain user information as little as possible. On the other hand, considering that the number and status of users change much faster than the speed of project changes, it is also difficult to generalize users.

2.3. System Implementation Framework. Figure 2.1 is an overall description of the algorithm framework. In practical recommendation system applications, the main steps are: (1) The system background collects relevant data generated by users, that is, input data. Here, the system collects user's pan data, which refers to the behavior and preference data of users acting on project feature groups. This is different from traditional recommendation algorithms [16]. For example, a user's music trial record of a certain genre, track information in a playlist created by the user, etc. By processing the user's pan behavior data, obtain the user group evaluation matrix and the relationship between the group and the project; (2) Quantify the project set based on project features, calculate the similarity of project features, and obtain the group division of project features. Fill in the rating matrix of user projects; (3) Predict users' preferences or ratings for specific projects based on their group ratings for projects with similar characteristics; (4) The recommendation system makes top-n recommendations based on the predicted score. The steps with high computational complexity throughout the entire system implementation process are concentrated in steps (1), (2), and (3). In order to improve the operational efficiency of the system, it is considered to process these steps offline. Firstly, the feature vectors of the user matrix and product feature matrix are calculated offline from existing data. Online, the recommendation results are mainly calculated based on the feature vectors, thereby improving the system's operational efficiency. On the other hand, the system then transfers data such as the user's behavior towards the recommended item to the storage system. After the data is updated, the updated recommendation model is obtained through offline calculation steps (1), (2), and (3). In addition, we can see that this recommendation model not only achieves privacy protection, but also makes the system more convenient to implement because the implementation of the system only requires user evaluation information on project feature groups. Compared to traditional recommendation

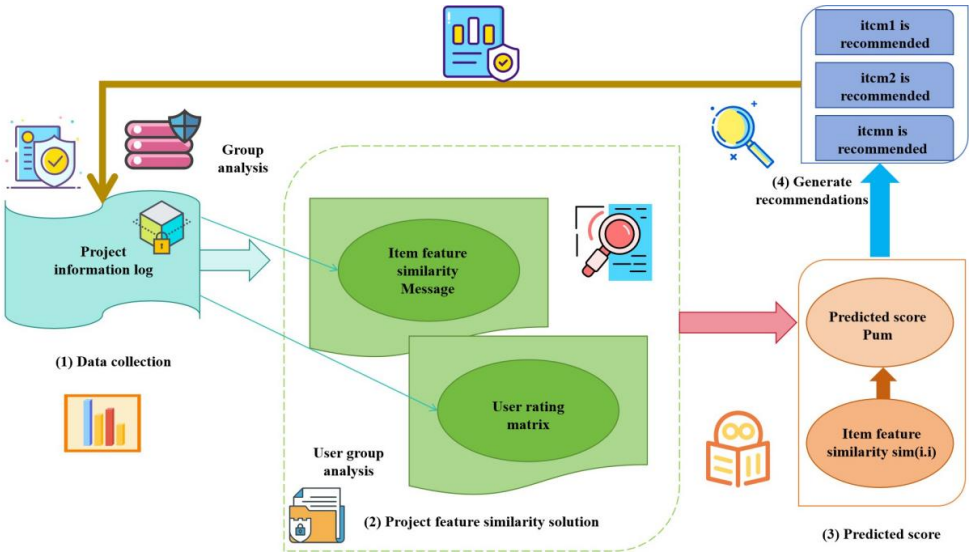

Fig. 2.1: Recommendation Framework

Table 3.1: Experimental Dataset Statistics

User data	Item Quantity	Number of ratings	Project category
944	1683	100 000	21

systems that need to collect user evaluation information on specific projects, this type of user evaluation data for groups is easier to collect because users rarely evaluate a large number of specific projects. Secondly, due to the coarse-grained macro evaluation information collected by the system, this data is relatively less noisy.

3. Experimental Results and Analysis.

3.1. Experimental Plan Design. The experimental data adopts the MovieLens dataset provided by the GroupLens project team at the University of Minnesota in the United States. The MovieLens movie recommendation system is a web-based research type that allows users to rate movies they have watched. Based on the user's historical rating information, the system predicts their ratings for other movies they have not watched and recommends movies with high predicted scores to users, I believe these movies are the next ones that users are interested in. The data of its system is also a commonly used dataset for experimental analysis by researchers. The MovieLens dataset contains 100000 rating data from 943 users on 1682 movies. The rating score is an integer from 1 to 5, and the larger the value, the higher the user's preference for the movie. Each user has rated at least 20 movies. The basic information of the dataset is shown in Table 3.1 [17].

The author randomly selected rating data from 300 users on 600 movies from the MovieLens dataset and compiled a feature description matrix for these 600 movies based on the website's demographic information. All experiments by the author will use this dataset, and 80% of it will be used as the training set and 20% as the testing set. Considering that the algorithm proposed by the author is based on the project feature model, it is necessary to first establish a film feature model. The feature attributes of the film are selected from the original data source, including 18 types such as Action, Adventure, Animation, Children's, Comedy, Crime, Documentary, Drama, Fantasy, Horror, Musical, Mystery, Romance, Sci-fi, Thriller, War, Western, Film noir, etc, Each film may have multiple attributes mentioned above at the same time [18]. For the given feature attributes mentioned above, it is first necessary to quantify them and convert them into computable data formats. Determine the value of the vector component based on whether it exists in the project. For the MovieLens movie dataset used by the author, for example, if movie a is both an action film and a crime and

war film, the feature vectors of movie a are sorted according to the familiar movie features mentioned earlier. The first, sixth, and sixteenth components of the feature vector are 1, and the other unfamiliar components are set to zero, i.e. (1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0). After quantifying the feature attributes of all test items, the project feature similarity matrix can be obtained. The similarity between projects can be calculated using formula (5), and the project group information can also be obtained by setting the number of groups to Ng and the number of items contained within the group to s (for this dataset, one project is also a movie). The author categorizes projects based on project feature categories, but using this classification method results in a fixed number of project groups Ng, where the number of projects s can be adjusted and determined by setting a similarity threshold.

$$sim(i,j) = \frac{|n_{ij}|_C^{t_i=t_j}}{|n_{all}|_C - |n_{ij}|_C^{t_i \neq t_j}} \tag{3.1}$$

In addition, for the author recommendation model, in its practical application, it only utilizes user rating data information for project feature groups, and does not require user ratings for specific projects. In this experiment, the selected dataset already includes specific ratings of user projects. In order to verify the effectiveness of the model, we used the above method to process project rating data in the experiment. In practical applications, only user behavior data about the group needs to be collected. This experiment mainly used three evaluation criteria, namely the root mean square error (RMSE) index to measure the accuracy of recommendation scoring, the mean absolute error (MAE) to measure the degree of deviation, and NDCG (Normalize Discounted Cumulative Gain) to measure the accuracy of recommendation item ranking. Mean Absolute Error (MAE) is the average absolute value of the deviation between all individual observations and the arithmetic mean. Its definition is:

$$MAE = \frac{\sum\limits_{i \epsilon I} |\hat{r_i} - r_i|}{|I|} \tag{3.2}$$

I represents the test set, $\hat{r_i}$ represents the predicted score, and r_i represents its actual score.

The expression for root mean square error is defined as follows:

$$RMSE = \sqrt{\frac{\sum\limits_{u,j \epsilon T} (r_i - \hat{r_{ui}})^2}{|T|}} \tag{3.3}$$

The symbol in the equation represents the actual score as r_{ui}, and the system predicted score as $\hat{r_{ui}}$.

NDCG is an indicator used to measure the quality of sorting. NDCG is a numerical evaluation of whether the ranking of recommendation lists provided by a recommendation system is reasonable, in other words, it evaluates whether the ranking of recommendation items is similar to the order of actual ratings. For example, NDCGp represents the NDCG value for the top p item positions in the recommendation list, calculated as follows:

$$NDCG_p = \frac{DCG_p}{IDCG_p}, DCG_p = rel_l + \sum_{i=2}^{p} \frac{r_i}{log_2 i} \tag{3.4}$$

r_i represents the user's rating on item i, and IDCG (ideaDCG) is the ideal DCG. In this experiment, the actual ranking result is calculated based on the actual rating of the test set, and then DCG_p is calculated to obtain $IDCG_p$.

3.2. Comparative experiments. Two comparative experiments were used in this experiment:

1. The user project rating matrix is filled in with an average score (Baseline), which is the average of the existing user ratings for the project [19].
2. Using the Probability Matrix Decomposition Algorithm (PMF), the principle of PMF is to reconstruct the rating matrix by utilizing the feature vectors of users and projects, thereby achieving recommendations.

Table 3.2: Comparison of Results

D	5		10		18	
Model	RMSE	$NDCG_5$	RMSE	$NDCG_5$	RMSE	$NDCG_5$
Baseline	1.0513	0.6953	1.0513	0.6953	1.0513	0.6953
PMF	0.9239	0.8162	0.9254	0.8156	0.9249	0.8178
Consider privacy algorithms	0.9878	08122	0.9663	0.8126	0.9669	0.8126

Table 3.3: Comparison of Loss Rates

D	5		10		18	
Model	RMSE	$NDCG_5$	RMSE	$NDCG_5$	RMSE	$NDCG_5$
Loss rate	4.23%	0.48%	4.43%	0.34%	4.44%	0.36%

If analyzed theoretically alone, the PMF algorithm will have higher accuracy than the author's algorithm that uses coarse-grained information based on user project feature similarity division as the training set, as it uses a training dataset with user ratings for specific projects and has finer granularity. After comparing the recommendations of Baseline, it is not difficult to find that the implementation effect of Baseline is actually relatively poor. This is because Baseline recommendations only consider the average rating of users on the project and do not conduct specific learning on the project.

3.3. Experimental Results and Analysis.

(1) Experiment 1: Selecting Different Project Feature Dimensions to Calculate Project Feature Similarity. This experiment compared the implementation effects of various algorithms under different feature vector dimensions. In the experiment, the feature vector dimensions D=5, 10, and 18 were selected. Regarding the algorithm proposed by the author, all other parameters are set to achieve optimal values for each algorithm. In order to intuitively evaluate the difference between the author's algorithm and the PMF algorithm in the actual accuracy of recommendations, a loss rate indicator has been defined, and its expression is as follows:

$$\delta = \frac{|a_{PMP} - a|}{a_{PMP}} \tag{3.5}$$

Table 3.2 shows the comparison of RMSE and NDCGs calculation results for each algorithm under different feature matrix dimensions, while Table 3.3 shows the comparison of loss rates for each algorithm under different feature matrix dimensions. The comparison of experimental results is shown in Tables 3.2 and 3.3 below.

From the experimental results, we can analyze and draw the following conclusions: As the dimension D of project features increases, there is a certain improvement in the accuracy of each algorithm, but the impact on the Baseline algorithm is not significant. The reason for the analysis is that the increase in the dimension D of the feature vector makes the description of project features more specific, which can effectively reduce errors. However, it must be noted that the increase in the dimension D of the feature vector can lead to an increase in computational complexity, so it is necessary to balance efficiency and accuracy. It can be seen that the privacy based collaborative filtering algorithm proposed by the author has poor implementation performance compared to PMF, with a loss rate of over 4%. The main reason for the analysis is due to the selection of training datasets. The training set of PMF is the user's evaluation data of the project, with finer granularity; The privacy based collaborative filtering algorithm proposed by the author uses a dataset of user evaluation data for grouping project feature similarity, with coarse granularity, resulting in insufficient accuracy. However, compared to the Baseline algorithm, privacy based collaborative filtering algorithms still have significant advantages, which also demonstrates the effectiveness of the privacy based collaborative filtering algorithm proposed by the author.

From the comparison of NDCGs indicators, combined with the results in Table 3.3, compared to the PMF algorithm, the privacy based collaborative filtering algorithm proposed by the author does not show significant changes in this indicator as the feature vector dimension decreases. This indicates that the impact on sorting

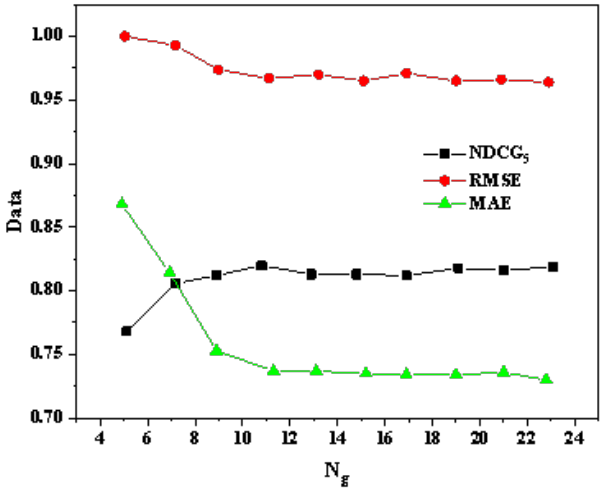

Fig. 3.1: N_g impact of AA on Algorithm

accuracy is not significant. In practical applications, the recommendation system for two-dimensional user data collects user behavior data, which is also simple "like" or "pop" information. In response to this situation, sorting accuracy has better measurement significance, and NDCG should be used as an evaluation indicator. Overall, the implementation effect of the recommendation model proposed by the author is acceptable and has practical application value.

(2) Effect of Different Data Granularities in Experiment 2. Data granularity refers to the level of granularity of data. When the number of items in each feature group is fixed, there are two main parameters that affect data granularity: The number of feature selections N and the number of nearest neighbors. Figures 3.1 and 3.2 respectively illustrate the variation of experimental results with the two parameters.

From Figure 3.1, it can be observed that as the number of feature selection N in the project gradually increases, MAE Both RMSE and NDCG5 are gradually improving, mainly due to the fact that increasing the number of feature selections N to a certain extent reduces the granularity of the data when s is fixed. The increase in the number of target feature selections will make the project feature description more specific. The calculation of project feature similarity obtained from this is more accurate, and the accuracy of similarity evaluations made by users based on similar project feature preferences is also higher, resulting in better results. However, this will also have a certain impact on the privacy protection effect of the recommendation model. It is worth noting that simply increasing the number of feature selections for a project does not result in a continuous improvement in MAE, RMSE, and NDCG5. However, when the number of feature selections for a project increases to a certain value, the measurement indicators do not show a significant improvement. This is mainly because an increase in the number of feature selections for a project can cause overfitting to a certain extent.

Figure 3.2 shows the impact of the number of nearest neighbors s within a similar group of project feature selection on the algorithm. It can be seen from the figure that, under a fixed N, MAE, RMSE, and NDCGs continuously deteriorate with the increase of s [20]. This is mainly due to the increase in the number of items within the group, resulting in larger data granularity. The reduction of useful data makes it more difficult to "purify" the data, that is, an increase in s will increase the bias of inferring users' preferences for specific items.

4. Conclusion. Currently, more and more recommendation systems are being applied to various internet applications, and users' enthusiasm for participation is constantly increasing. With this, there are issues with user privacy data security. The author extends the foundation of the proposed recommendation model to achieve

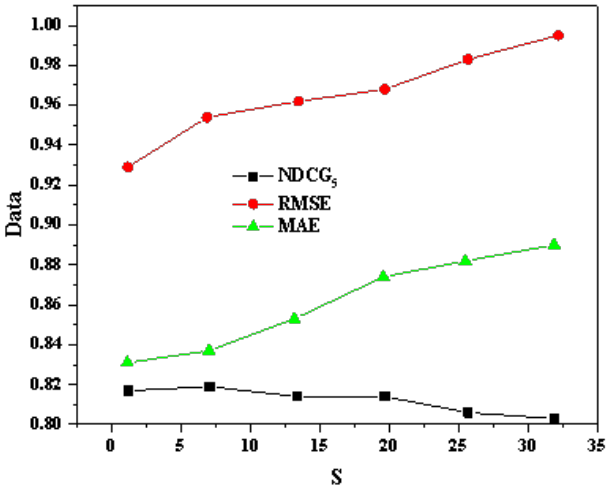

Fig. 3.2: The impact of s on the algorithm

a privacy protected recommendation model. In summary, the proposed recommendation model that considers privacy protection only collects "coarse-grained" information from users, while in terms of projects, projects are divided by calculating the project degree of project features, and these data are recommended to achieve the effect of protecting user privacy data. Firstly, the recommendation framework of collaborative filtering algorithm based on privacy protection was introduced, and the implementation principles were analyzed and derived; Subsequently, corresponding verification was conducted through experiments. It was found that although the privacy protection based collaborative filtering algorithm caused a certain degree of accuracy loss due to only utilizing coarse-grained data from users, it achieved privacy protection function and its implementation effect was also acceptable.

REFERENCES

[1] Parthasarathy, G. , & Devi, S. S. (2023). Hybrid recommendation system based on collaborative and content-based filtering. Cybernetics and Systems, 54(4), 432-453.
[2] Kim, T. Y. , Ko, H. , Kim, S. H. , & Kim, H. D. (2021). Modeling of recommendation system based on emotional information and collaborative filtering. Sensors, 21(6), 1997.
[3] Zeng, Y. , & Liu, S. (2021). Research on recommendation algorithm of graph attention network based on knowledge graph. Journal of Physics: Conference Series, 2113(1), 012085-.
[4] Pan, R. , Qishan, Y. U. , Xiong, H. , & Liu, Z. (2023). Collaborative recommendation algorithm based on deep graph neural network. Journal of Computer Applications, 43(9), 2741-2746.
[5] Du, Z. , Deng, M. , Lyu, N. , & Wang, Y. (2023). A review of road safety evaluation methods based on driving behavior. Journal of Traffic and Transportation Engineering (English Edition), 10(5), 743.
[6] Cui, Y. (2021). Intelligent recommendation system based on mathematical modeling in personalized data mining. Mathematical Problems in Engineering, 2021(3), 1-11.
[7] Jin, B. , Liu, D. , & Li, L. (2022). Research on social recommendation algorithm based on fuzzy subjective trust. Connection Science, 34(1), 1540-1555.
[8] Liang, W. , Xie, S. , Cai, J. , Xu, J. , & Qiu, M. (2021). Deep neural network security collaborative filtering scheme for service recommendation in intelligent cyber-physical systems. IEEE Internet of Things Journal, PP(99), 1-1.
[9] Song, H. Y. , Zhang, H. , & Xing, Z. H. (2021). Research on personalized recommendation system based on association rules. Journal of Physics: Conference Series, 1961(1), 012027 (10pp).
[10] Fan, Y. , Ma, H. , Chen, Z. , & Shen, K. (2021). Research and application of algorithm based on maximum expectation and collaborative filtering in recommended system. Journal of Physics Conference Series, 1754(1), 012205.
[11] Li, P. , Han, Y. , Wen, X. , & Meng, F. (2022). Improvement and research of collaborative filtering algorithm based on

penalty factor. Journal of Physics: Conference Series, 2209(1), 012026-.
[12] Wu, L. (2021). Collaborative filtering recommendation algorithm for mooc resources based on deep learning. Complexity, 2021(46), 1-11.
[13] Luo, S. (2021). Research on collaborative filtering of food information security in e-commerce platform. Journal of Physics: Conference Series, 1757(1), 012191 (12pp).
[14] Singh, V. K. , Sabharwal, S. , & Gabrani, G. (2022). A novel collaborative filtering based recommendation system using exponential grasshopper algorithm. Evolutionary Intelligence, 16(2), 621-631.
[15] Zhang, W. , Zhou, X. , & Yuan, W. (2021). Collaborative filtering algorithm based on improved time function and user similarity. Journal of Physics: Conference Series, 1757(1), 012080 (8pp).
[16] Awan, M. J. , Khan, R. A. , Nobanee, H. , Yasin, A. , Anwar, S. M. , & Naseem, U. , et al. (2021). A recommendation engine for predicting movie ratings using a big data approach. Electronics, 10(10), 1215-.
[17] Nasy'an Taufiq Al Ghifari, Sitohang, B. , & Saptawati, G. A. P. (2021). Addressing cold start new user in recommender system based on hybrid approach: a review and bibliometric analysis. IT JOURNAL RESEARCH AND DEVELOPMENT, 6(1), 1-16.
[18] Cheng, H. , Gan, B. , & Zhang, C. (2021). Research on personalized recommendation method based on social impact theory. Journal of Physics: Conference Series, 1848(1), 012128 (7pp).
[19] Lv, Y. , & Kong, J. (2021). Application of collaborative filtering recommendation algorithm in pharmacy system. Journal of Physics: Conference Series, 1865(4), 042113 (5pp).
[20] Ramalingam, J. , Polsani, P. , Shruthi, K. , & Sujatha, D. M. (2021). Online social voting recommendation system based on collabrative filtering. Journal of Resource Management and Technology, 12(1), 20-27.

Edited by: Zhigao Zheng
Special issue on: Graph Powered Big Aerospace Data Processing
Received: Nov 30, 2023
Accepted: Dec 15, 2023