DOI: https://doi.org/10.46632/daai/2/4/14

Data Analytics and Artificial Intelligence
Vol: 2(4), 2022
REST Publisher; ISBN: 978-81-948459-4-2
Website:http://restpublisher.com/book-series/data-analytics-and-artificial-intelligence/

Blockchain Based Privacy Preserving Recommendation System (BPPRS) Using Digital Signatures

* **Shimona E, W. Thamba Meshach**
Prathyusha Engineering College, Tiruvallur, Tamilnadu, India.
*Corresponding author Email: shimona.cse@prathyusha.edu.in

Abstract. Recommendation system plays an important role in the digital world, since it helps to find users interest. Collaborative filtering is the widely used algorithms in recommendation systems due to its simplicity and efficiency. However, when the user rating data is sparse, which leads to generate unreasonable recommendations for those users who provide no ratings. Faced with these problems, the proposed Blockchain based Privacy Preserving Recommendation System (BPPRS) model is built using a new collaborative filtering algorithm for recommendation by combining the Jaccard Similarity and Triangle Similarity (JSTS). Using the values calculated from JSTS, an item similarity matrix is constructed in order to find the zero rated users so that the item name of these users are recommended to the particular user who are closely similar. The experimental result shows that the proposed model mitigates the sparseness of the information. Further, the BPPRS model is enabled by the use of Blockchain technology to generate the hash value for each of the items recommended to the user; previous hash value is used for current recommended items hash value and stored in different blocks. The encryption of the data ensures the secure recommendation and also the use of MECDSA makes it a better trap door function.
Keywords: Blockchain, Jaccard Similarity, Triangle Similarity, Recommendation System, Collaborative Filtering

1. Introduction

Recommendation systems find the user's interested things among an enormous amount of digital information. When user prefers for a specific product or an item, the Recommendation System come into play where the similar choices between two or more user are recommended to each other. Recommender systems usually make use of either content-based filtering or collaborative filtering. Content based filtering [4] methods support the overview of an item and the profile of the user's preferences. Content based recommenders treat recommendation as a user specific classification problem and learn a classifier for the user's interest and needs supported by an item's features. The recommendation is made based on the type of content the user prefers. The ratings given for similar content of the same category which they have preferred earlier by the user, that content is preferred to the particular user. Collaborative filtering [9],[22] CF is used to generate recommendations. It can be categorized into user-based CF and item-based CF. In Collaborative Filtering, items are recommended based to other user's with similar preferences and favourites to these items [7]. Collaborative filtering represents the user-item as a rating matrix. The goal of a privacy preserving recommendation system is to preserve users privacy by hiding their ratings from other users.. In order to secure the datas, the Data Privacy can be achieved. The Data Privacy ensure whether the user's data is not shared with the third party. It can be achieved by encryption, shuffling and suspension. The Digital Signature can be used to encrypt the data of the user.

2. Motivation

Recommendation systems cause the information overload and data sparsity problem due to the large volume of information. The user's data on the Recommendation System is not encrypted [2] which can be shared with the third party so the Digital Signature can be adopted to authenticate the users. The user may provide incorrect ratings or may not give any ratings as they believe that their profiles are being exposed. The organization acquires data about the rating preferences of many users and these data's could be transferred to third parties. **Major Contributions:** We aim to present an architecture that combines the Jaccard and Triangle Similarity which is used to calculate the similarity values. The recommendation is based on the similarity calculation between the user's by comparing the items. Recommendation of items to user is based on the similarity values calculated. Items of the similarity user's is recommend to the particular user's which are closely similar.Instead of generating the signatures for a group, the hash function can be computed for each user in their respected cluster. When storing it to a Blockchain, the previous hash value is used for current recommended items hash value and stored in different blocks. The signature should also prove that the item of the private key, who is by implication the item of the recommended user, has authorized which cluster they belong. **Organisation of this Work:** In the next section, we present the related work on recommendation system based on collaborative filtering and privacy preserving systems are thus discussed. The current techniques with respect to Blockchain based privacy preserving system is identified and described. Then, we present our proposed BPPRS model and the implementation of our work. Afterward, we present some experimental results with a conclusion of our work.

Related work: Collaborative filtering approaches build a model from a user's past behaviour also as similar decisions made by other users. To determine the user's similarity, Pearson correlation coefficient is computed [4]. Pearson correlation coefficient does not rely on privacy issues as there is a leakage of data. [23] Web service recommendation approach, such that time aware user-based and service-based QoS predictions are made. In this system, the QoS properties evaluate the values provided by the user's and does not cover the privacy issue of the user. [18] Proposed a Web Service Recommendation using PCC based Collaborative Filtering in which it makes use of Pearson Correlation Coefficient that produces a better value for QoS prediction for web services. [24] Proposed a k-nearest neighbour graph which considers both the preprocessing time and the query processing time. Here, it constructs the recommendation systems based on KNN based on Collaborative Filtering with cosine similarity and thus results in 64% of accuracy. [25] Proposed an item based collaborative filtering with slope one and nearest neighbour prediction in which it gives almost 92% of accuracy. [22] Proposed a new collaborative filtering approach with k-means clustering. The usage of clustering technology requires a session file and transaction file which can easily be hacked. [9] Proposed a novel recommendation system using the advantages of block-chain supported secure multi-party computation. A typical fraud is conducted by creating dummy profiles to manipulate the desirability of items and products. [5] Proposed a unique classification based shilling attack detection protocol in which the unauthorized profiles in arbitrarily distributed configurations are analysed without compromising the privacy of collaborating parties. [11]proposed a Blockchain based authentication key exchange protocol which is based on one-way hash function. The usage of one-way hash function proves that it is more secure and flexible. [13] proposed different approaches for building a recommendation system. The recommendations generated for an intended user should be available only to that user, server should not learn about predicted ratings of any item for the specific user. A typical fraud is conducted by creating dummy profiles to manipulate the desirability of items and products.

3. BPPRS model

Collaborative Filtering algorithm constructs similarity matrix to predict target ratings by finding user sets or item sets similar to target user's or items. From the Fig. 1, by combining Jaccard similarity and triangle similarity, a new similarity calculation method is proposed to improve the accuracy of the rating. After calculating similarity, the zero rated users are identified the items of both user's. Recommendation of items to user is based on the similarity values calculated. Items of the similar user is recommended to the particular user who are closely similar. The proposed Multiple Elliptic Curves Digital Signature Algorithm (MECDSA), detects the unauthorized data and the authentication of the user.

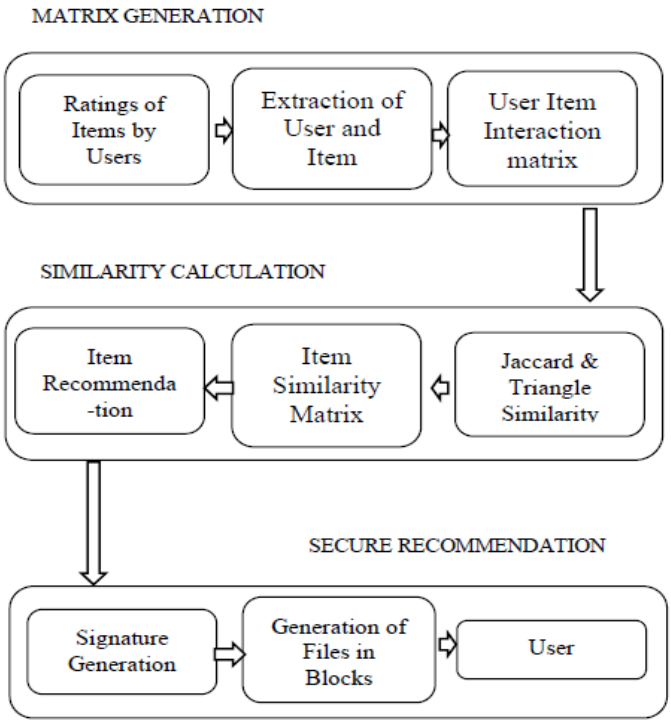

FIGURE 1. BPPRS Model

A digital signature uses both the functions of hash and public key cryptography. Firstly, using the functions of hash, the hash value is created. Then the hash value is encrypted which has been created using users key and thus leads to the generation of signature. In this BPPRS model, the usage of multiple elliptic curve digital signatures algorithm makes it more secure and efficient.

Item Recommendation: Firstly, the extraction of user ID and item ID is used to cluster the item user IU matrix, divide the user into corresponding clusters, and then label each user to identify the cluster they belong to by using a User ID. The feature extraction is also performed to construct a new user group UG matrix. The new matrix is constructed using the item as the row and the user's cluster as the column. In the matrix, each element represents a group of user's belonging to the same cluster and rating the same item. The new user group matrix mitigates the sparsity of the original matrix rating data. Combining Jaccard similarity and Triangle Similarity (JSTS), a new similarity calculation method is proposed to improve the accuracy of the rating. Triangle similarity considers common rating user's or items and does not work well when used alone. Therefore, the Jaccard similarity performs well in the similarity calculations of no common rating user's or items. Hence, combining the triangle similarity with Jaccard similarity, a new similarity is constructed. Therefore, an improved similarity calculation method is proposed

$$\text{Sim(user1,user2)}=\frac{\sum(\text{user 1}*\text{user 2})}{\sqrt{\sum(\text{user 1}*\text{user 1})}*\sqrt{\sum(\text{user 2}*\text{user 2})}} \qquad (1)$$

From Equation 1, the similarity values between the users should be calculated for all the items in the data. The Summation of user1*user2, user1*user1 and user2*user2 are the values derived from the ratings of user 1 and 2. The Items are recommended to the other users of the clusters by performing rate prediction only after clustering the items. Combination of Triangle and Jaccard Similarity, the user group UG matrix is constructed based on item_user IU matrix to calculate the similarity between two different items to perform rate prediction on items for the users of each cluster. Find the zero rated users from the item similarity matrix and insert into ratings table of the SQLyog database. Extract the item ID for zero rated user. Recommend the corresponding item name of the item ID that has been extracted to the user ID that they belong to.

Secure Recommendation: The items are recommended to the users of each group are given as input message (M) to the SHA1PRNG hash function as illustrated in the Fig. 2. The SHA1PRNG generates a unique 256 bit hash code which is termed as Message Digest H(M). In MECDSA, the sender chooses the number of curve and the parameters of the curves as (Ai,Bi). After this, the sender chooses t random numbers $ki \in [1, Ni]$ as a private key, and computes the temporary public key TKi. The private key of sender (PvKi), Temporary Public Key (TKi) and Message Digest H(M) is given as input to the signing operation of MECDSA in which the R and Si are computed. Therefore the Digital Signature (R,Si) can be obtained as output through the MECDSA.

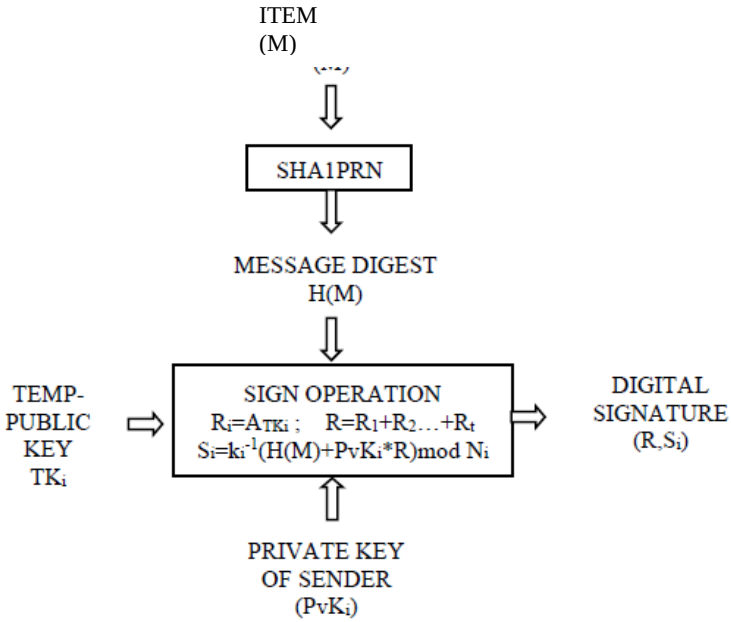

FIGURE 2. Flow Diagram for Signature Generation

The hash values generated by the Digital Signature are used to store the data in blocks of the Blockchain. Blockchain Technology has a broad set of features which helps in preserving the privacy of the user. The block is like a link in the chain of a system where it stores the records of the transaction. Therefore the Blockchain technology is adopted to store the hash value for each of the items recommended to the user; previous hash value is used for current recommended items hash value and stored in different blocks. Therefore, these hash values are used to recommend the items to the users in a secure manner.

Implementation: This section describes the implementation of the BPPRS model. The experimental setup is made with all the default packages and the library files are imported. The extraction of user ID and item ID is done and a matrix generated for which the similarity calculation and zero rated users are identified. The signatures are generated for each recommendations and they are stored in a Blockchain.

Experimental Setup: The experimental setup is made with all the default packages and the library files are imported. The Front end and Back end connectivity are established to build the recommendation system. The Front end connectivity is established using Net beans and Java package with all the default library files. The Back end connectivity is established using SQLyog database with all the necessary library files to establish a Java Database Connectivity (JDBC) connection.

Extraction and Matrix Generation: A user.csv dataset is used where it consists of 6000 ratings from 300 users on 20 items. The dataset contains the user ID, item ID, item name, rating, and the type of customer. In this dataset at least 20 items each user rated. The dataset consists of the ratings of 20 dishes given by 300 users from 5 different kinds of cities. This module extract the item ID and the user ID separated by a ','. Further they are stored onto the SQLyog as a table. Initializing the item_user IU matrix for item iwhere i=(1,2,…n) and for cluster jwhere j=(1,2,…k).Then find all users who have rated item i. Later, save the corresponding user to the corresponding IU[i,j] according to the cluster label to which the user belongs. Algorithm 1 describes the generation of user group matrix. The matrix generation stage includes the clustering of the data's based upon the user ID. Based upon the type of customer, the clustering is done. For example, user 1 and user 8 belongs to Chennai, so they are grouped under cluster 1 and user 2 and user 6 belongs to Coimbatore, so they are grouped under cluster 2. Similarly Madurai, Trichy and Tirunelveli are grouped as Cluster 3, Cluster 4 and Cluster 5 respectively.

Algorithm 1: User-Group Algorithm

Building user group matrix (UG) from excel file (user.csv).
Input: user.csv file, A MATRIX
Output: user group matrix (UG).
1. Begin
2. Import (user.csv) file and store content in an array (item_user IU). //such that no. of rows equals to user file rows and first column contain user id second column is item id and last column is type of customer.
3. Alldata[i]=IU[i] ,tempitem=Alldata[1] and item[0]=tempitem.
4. If (items[i].equals(tempitem)) = Alldata[i++] // to check whether the item ID of tempitem already exists in item array.
5. Similarly proceed with extraction of user ID from IU.
6. Initializing UG matrix with zeros.
7. If (items[i].equals(Alldata[1]) && users[i].equals(Alldata[0])) do
8. UG [i][i]=Alldata[3]; // store the ratings of user 1 for all 20 items.
9. End

Similarity Calculations: By the use of Combining Triangle and Jaccard Similarity, the similarity of item i and item j of matrix $M_{n\times k}$ is calculated and then store the similarity in S[i,j] and S[j,i]. Algorithm 2 describes the generation of item similarity matrix from the clustered matrix UG by evaluating the combination of Jaccard and Triangle similarity calculation. For each of the user's in the clusters, the similarity calculations are made and the values are stored in SQLyog database.

Algorithm 2: Similarity Calculation

Construct user based similarity matrix.
Input: User Group matrix UG.
Output: Item Similarity matrix SM.
1. Begin
2. Group the users (g) from the UG.
3. If UG [i][j] .equals (g) do // where i is user ID and j is Item ID.
4. Calculate the summation of ratings between two users and also within the user.
5. Initialize the item similarity matrix SM;
6. For item i from i = 1 to n - 1 do
7. For item j from j = i + 1 to n do
8. Calculate the similarity between two users from Equation (1).
9. Store the similarity in SM [i, j] and SM[j,i].
10. End for
11. End for

Zero rated users: The items are recommended to the user by finding the zero rated users through similarity calculation. The similarity between different users of each cluster is calculated in the item similarity matrix SM. As described in the Algorithm 3, find the zero rated users from the item similarity matrix and insert into ratings table of the SQLyog database. Extract the item ID for zero rated user. Recommend the corresponding item name of the item ID that has been extracted to the user ID that they belong to.

Algorithm 3: Recommendation of items

Find the zero rated users from the item similarity matrix.
Input: Item similarity matrix (SM).
Output: Recommendation of items that have not been reviewed by users.
1. Begin
2. Retrieve the similarity values for target user with all items in the SM matrix.
3. Find the zero rated users from the SM matrix and extract the item ID.
4. Retrieve the item name for the extracted item ID from the SQLyog database.
5. Recommend the retrieved item name to the corresponding user that they belong.
6. End

Signature Generation: The items are recommended to the users of each cluster are given as input message (M) to the SHA1PRNG hash function as illustrated in the Algorithm 4. The SHA1PRNG generates a unique 256 bit hash code which is termed as Message Digest H (M). When there is a change in the recommendation, then there will be a change in the hash codes. The elliptic curve is used to generate the private keys and public keys. The point of origin (P_i) is randomly selected from an order N_i (number of points on curve) which is called as reference point. Generate a random prime number from P_i where $P_i<N_i$ called as the private key (PvK_i). Choose a random number k_i. A temporary public key $TK_i=P_i*k_i$ is calculated using the point of multiplication. The tangent from Pi intersects at a new point on the curve of order N_i. This new point is symmetric point from P_i and are computed for private times. The addition of all the symmetric points gives the point of multiplication. The coordinates of TK_i is (A_i,B_i) in which A_i is taken as the R_i of signature. In which $R=R_1+R_2+R_3+\ldots+R_t$ of the signing operation. From Equation 2, the S_i can be calculated.

$$S_i=k_i^{-1}(H(M)+PvK_i*R)\text{mod } N_i \qquad (2)$$

Where k_i^{-1} is the modular multiplicative inverse of k_i which is calculated from $(k_i^{-1}*k_i)\text{mod}N_i=1$, hash code H(M), private key of sender PvK_i and R of the signature. The resulting (R,Si) gives the digital signature generated through MECDSA.

Algorithm 4: Signature Generation

Creating a signature using MECDSA.
Input: Items recommended (M).
Output: Digital Signature (R,S_i).
1. Begin
2. Generate Message Digest H(M), 256 bit hash code using SHA1PRNG.
3. Choose a random point on the curve as the point of origin P_i from an order N_i.
4. Generate a random prime number from P_i where $P_i<N$ called as the private key (PvK_i) and choose a random number k_i.
5. Calculate temporary public key $Tk_i=P_i*k_i$ for private times using the point of multiplication and consider the x-coordinate of Tk_i as R_i of the signature where $R= R_1+R_2+R_3+\ldots+R_t$.
6. Calculate modular multiplicative inverse of k_i from $(k_i^{-1}*k_i)\text{mod}N_i=1$.
7. Calculate S_i of the signature from the equation 6.1.
8. Return (R,S_i).
9. End

Thus the BPPRS model has been implemented successfully. The experimental setup is made in which the similarity calculation and zero rated users are identified. The signature generations are made for each of the recommended items and are stored in Blockchain.

4. results and discussions

The implementation of BPPRS model was built with the following steps that have been discussed earlier. The result of the BPPRS model was discussed in this chapter with the procedural format. Load the csv file and obtain the input bytes from the file. Further read each text from the input byte. The user ID, Item ID, Item name, Rating and type of customer are extracted from the dataset and viewed in the application which are separated by ",". The entire 300 user ID's for each 20 items are displayed in a total of 6000 records. The items for each user ID's are examined and are stored in a separated string. These item ID's are uploaded to the SQLyog database. The main motive of extracting the item ID is they are helpful in building the item user IU matrix. The same procedure is carried out for the extraction of user ID's as done in the Extract Item. In order to know the number of user's from the 6000 records, this process is done. Therefore we have 300 user ID's from 6000 records. Further, the extracted user ID's and Item ID's are stored into the SQLyog database. The item name along with the item ID is stored as a separate table in the database so that during the recommendation, the item names are used instead of item ID's. Fig. 3. illustrates the clustering of user's from the extracted items and users. The clustering is done based upon the type of customer they belong to. The type of the customer portrays the location where they belong. The users are divided into 5 types of clusters namely Chennai, Coimbatore, Madurai, Trichy and Tirunelveli are grouped as Cluster 1, Cluster 2, Cluster 3, Cluster 4 and Cluster 5 respectively. Instead of calculating the similarity for individual users, the clustering makes it efficient by comparing the users containing in each cluster. Based upon the locations, the similar users are identified and further the recommendation of items are made.

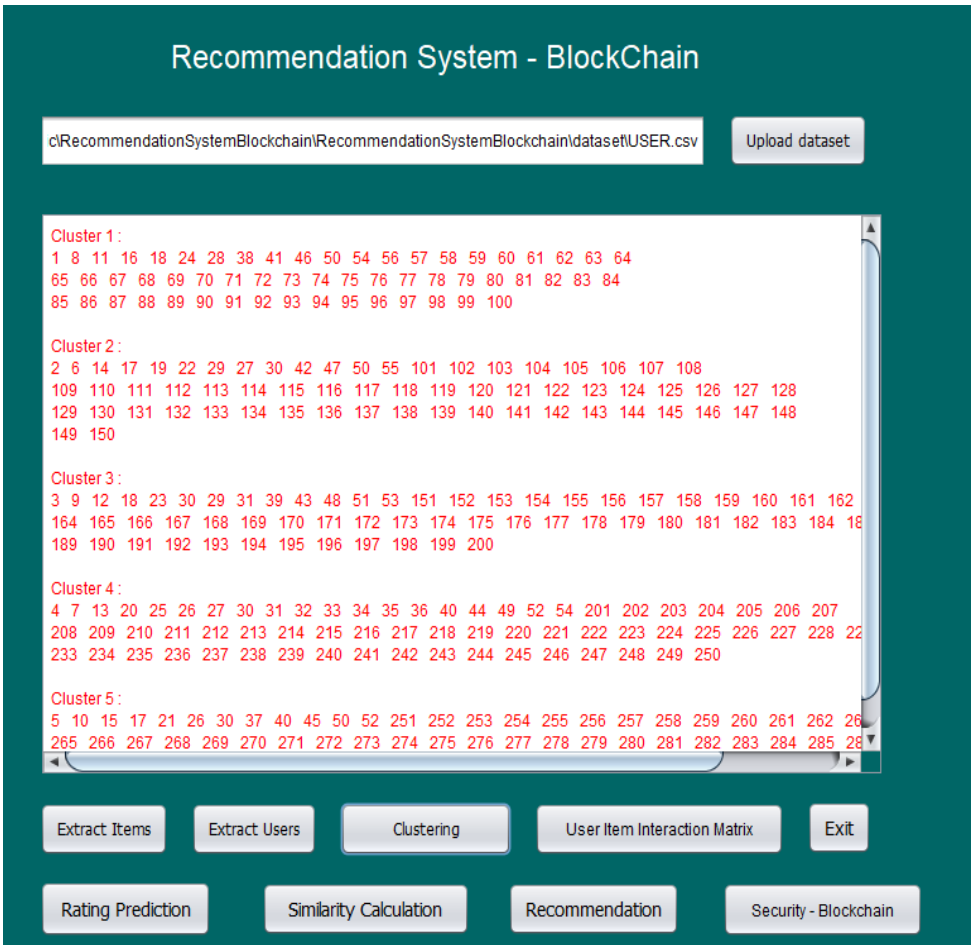

FIGURE 3. Clustering of Users

From the data, the User Group matrix is generated otherwise called as User Item Interaction Matrix using the clustering technique as illustrated in the Fig. 4. The users belonging to each city are clustered i.e. based on the type of location. The ratings of each items are displayed for similar users in a cluster.

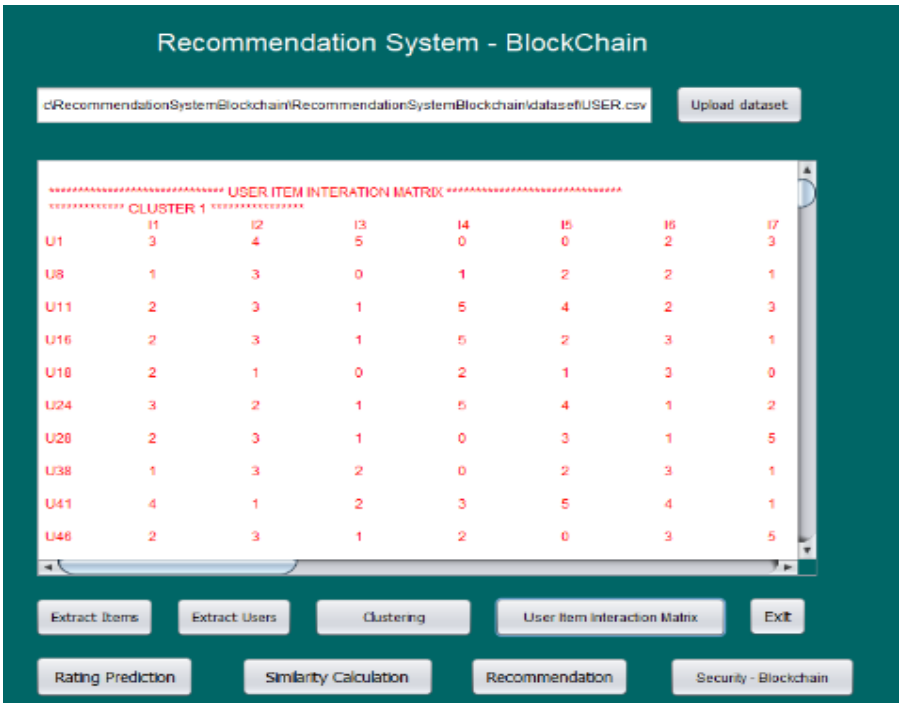

FIGURE 4. User Item Interaction Matrix

The Jaccard and Triangle Similarity is performed for all the users in the cluster. Later, the items are recommended to each of the users in the cluster.Fig. 5. illustrates the similarity values for each cluster that has been calculated using Jaccard and Triangle similarity. For example, from the given ratings of the user 1 and user 8 in cluster 1, the similarity calculations are made. These values are then stored in matrix called Item Similarity matrix for identifying the zero rated users. For extracting the item name for recommendation, the similarity values between two users are stored in the SQLyog DB for all types of possible combination of user's in each clusters.

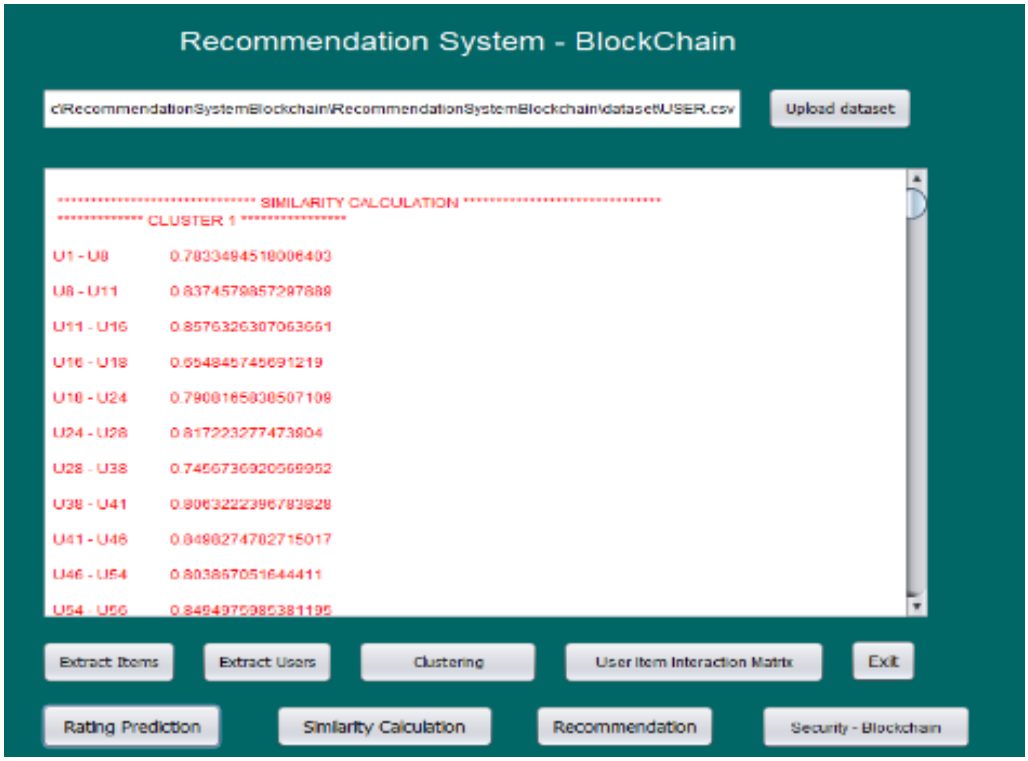

FIGURE 5. Similarity Calculations

By finding the zero rated users from the item similarity matrix and inserted into ratings table of the SQLyog database. the item ID for zero rated user are extracted and further recommend the corresponding item name of the item ID that has been extracted to the user ID that they belong to. Therefore the users who haven't had any rating for the items are recommended to them by those who belong to the same cluster i.e. same city. For example, the user 1 haven't given any rating for the items like vadai, roast, egg biryani, curd rice and ghee rice but the other user who belong to same cluster1 (Chennai) may or may not have the ratings for these item. Hence these items are recommended to the user1.

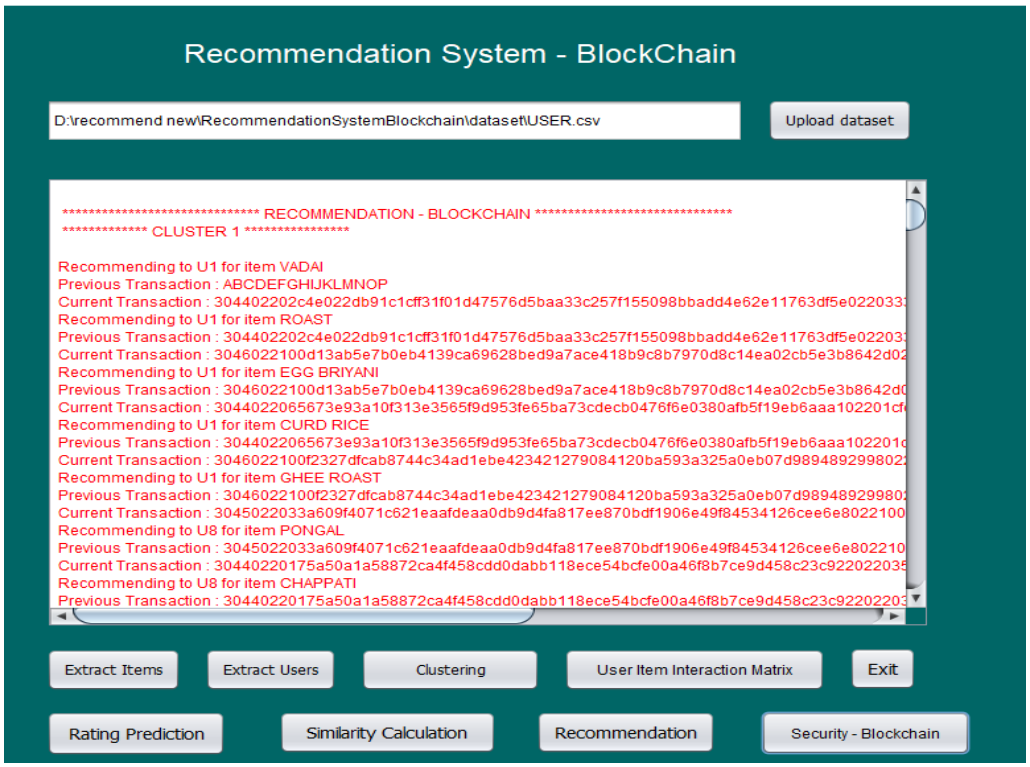

FIGURE 6. Secure Recommendation

Each Items recommended is considered as the transactions as illustrated in the Fig. 6. Each transaction is hashed by the digital signature algorithm in the above module and store in the text file as one block. Next transaction is linked with current transaction hash data and stored as the next block in text file of the local directory.

5. Conclusion

Thus a recommendation model is proposed to eliminate the data sparsity problem in CF. The proposed model clusters the item user matrix to extract corresponding features to construct a new user group matrix to eliminate the impact of data sparseness on zero rated users. The proposed recommendation system is based on the similarity calculation between the users by comparing the items. The zero rated users are identified based upon the similarity values. Items of the similar users are recommended to the other users in the corresponding cluster based upon the zero rated users. The digital signatures are generated for the recommended items and the Blockchain technology is adopted to store the digital signatures. The encryption of the data ensures the secure recommendation and also the use of MECDSA makes it a better trap door function.

Reference

[1]. AidmarWainakh, Tim Grube, Jorg Daubert and MazxMuhlhauser (2019), 'Efficient privacy-preserving recommendations based on social graphs', ACM Transaction on Recommender Systems, Vol. 21, No. 4, pp. 78-86.

[2]. Bernal Bernabe, Canovas J., Hernandez Ramos and Torres Moreno (2019), 'Privacy-Preserving Solutions for Blockchain: Review and Challenges', IEEE Transactions on Engineering Management, Vol. 7, No.9, pp. 164908-164940.

[3]. Bin Yu, Chenyu Zhou, Chen Zhang and Yiming Fan (2020), 'A Privacy Preserving Multi-Task Framework for Knowledge Graph Enhanced Recommendation', IEEE Access, Vol. 8, No.3, pp. 115717 – 115727.

[4]. Bogdan Walek and Petra Spackova (2018), 'Content-based recommender system for online stores using expert system', IEEE Transactions on Engineering Management, Vol. 8, No. 4, pp. 164-166.

[5]. BurcuYilmazel, Alper Bilge and CihanKaleli (2019), 'Privacy-Aware Detection of Shilling Profiles on Arbitrarily Distributed Recommender Systems', IEEE Transactions on Engineering Management, Vol. 48, No.10, pp. 28863-28885.

[6]. David Golberg, David Nichols, Brian M Oki and Douglas Terry (1997), 'Using Collaborative Filtering to Weave Information Tapestry', ACM Transaction on Recommendation system, Vol. 4, No. 3, pp. 56-58.

[7]. Fran Casino and ConstantinosPatsakis (2019),'An Efficient Blockchain Based Privacy Preserving Collaborative Filtering Architecture', IEEE Transactions on Engineering Management, Vol. 67, No. 4, pp. 1501-1513.

[8]. Fran Casinoa, Thomas K. Dasaklisb and ConstantinosPatsakisa (2018), 'A systematic literature review of blockchain based applications current status, classifications and open issues', ACM Transaction Privacy and Security, Vol. 36, No. 7, pp. 55-81.

[9]. Frey, Remo, Wörner, Dominic, Ilic and Alexander (2016), 'Collaborative Filtering on the Blockchain: A Secure Recommender System for e-Commerce', IEEE Transactions on Engineering Management, Vol. 53, No.3, pp. 278-281.

[10]. Gediminas Adomavicius and Alexander Tuzhilin (2005), 'Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions', IEEE Transactions on Knowledge and Data Engineering, Vol. 17, No. 6, pp. 734-749.

[11]. Hailong Yao, Caifen Wang, Bo Hai and Shiqiang Zhu (2018), 'Homomorphic Hash and Blockchain Based Authentication Key Exchange Protocol for Strangers', ACM Transaction Privacy and Security, Vol. 40, No. 7, pp. 243-248.

[12]. Jinsu Kim, Dongyoung Koo, Yuna Kim, Hyunsoo Yoon, Junbum Shin and Sungwook Kim (2016), 'Efficient Privacy-Preserving Matrix Factorization for Recommendation via Fully Homomorphic Encryption', ACM Transaction Privacy and Security, Vol. 21, No. 4, pp. 284-287.

[13]. Kunal Shah, AkshaykumarSalunke, Saurabh Dongare and KisandasAntala (2017), 'Recommender systems: An overview of different approaches to recommendations',IEEE Transactions on Engineering Management, Vol. 75, No. 6, pp. 125-128.

[14]. Lo Yao Yeh, Peggy Joy Lu, Szu-Hao Huang and Jiun-Long Huang (2020), 'SoChain: A Privacy-Preserving ddos Data Exchange Service Over SOC Consortium Blockchain', IEEE Transactions on Engineering Management, Vol.2, No.4, pp. 353-356.

[15]. Mingdong Tang, Zibin Zheng, Guosheng Kang, Jianxun Liu and Yatao Yang (2016),'Collaborative Web Service Quality Prediction via Exploiting Matrix Factorization and Network Map', IEEE Transactions on Network and Services, Vol.13, No.1, pp. 126-137.

[16]. Nguyen, Hoang, Nguyen, Niyato, Nguyen and Dutkiewicz (2019), 'Proof-of-stake consensus mechanisms for future blockchain networks: fundamentals, applications and opportunities', IEEE Access, Vol.7, No.5, pp. 8572–85745.

[17]. Nunung Nurul Qomariyah and Ahmad Nurul Fajar (2019), 'Recommender System for e-Learning based on Personal Learning Style', ACM Transaction on Recommendation Systems, Vol. 21, No.54, pp. 563-567.

[18]. Sabanaz Sirajuddin Peerzade (2017), 'Web Service Recommendation using PCC based Collaborative Filtering', IEEE Transactions on Engineering Management, Vol. 31, No. 15, pp. 2920-2924.

[19]. Shakila Shaikh, Sheetal Rathi and Prachi Janrao (2017), 'Recommendation System in E-Commerce Websites: A Graph Based Approached',IEEE Transactions on Service Computing, Vol. 4, No. 24, pp. 931-934

[20]. Shu Yun Lim , Pascal TankamFotsing , Abdullah Almasri , Omar Musa, Tan Fong Ang and Reza Ismail (2018), 'Blockchain Technology the Identity Management and Authentication Service Disruptor: A Survey', International Journal on Advanced Science Engineering and Information Technology, Vol. 8, No. 1735, pp. 4-2.

[21]. Toqeer Ali Syed, Ali Alzahrani, Salman Jan, Muhammad Shoaib Siddiqui and Adnan Nadeem (2019), 'A Comparative Analysis of Blockchain Architecture and its Applications: Problems and Recommendations, IEEE Access, Vol. 7, No.4, pp. 176838–176869.

[22]. Xingyuan Li (2011), 'Collaborative filtering recommendation algorithm based on cluster', IEEE Transactions on Service Computing, Vol. 13, No. 10, pp. 2682-2685.

[23]. Yan Hu, Qimin Peng, Xiaohui Hu and Rong Yang (2018), 'Time Aware and Data Sparsity Tolerant Web Service Recommendation Based on Improved Collaborative Filtering', IEEE Transactions on Service Computing, Vol. 8, No. 5, pp. 782-794.

[24]. Youngki Park, Sungchan Park, Sanggoo and Lee Woosung Jung (2014), 'Fast Collaborative Filtering with k-nearest neighbour graph', IEEE Transactions on Big Data, Vol. 41, No.17, pp. 92-95.

Zhonghuo Wu, Jun Zheng, Su Wang and Hongfeng Feng (2013), 'A Combined Predictor for Item-Based Collaborative Filtering', IEEE Transactions on Engineering Management, Vol. 11, No.4, pp. 261-265.

