Hash-based and privacy-aware movie recommendations in a big data environment

Tingting Shao

Medical Information Engineering School,
Jining Medical University,
No. 669, Xueyuan Road, Donggang District, Rizhao, China
Email: Shaotingting03@163.com

Xuening Chen*

Student Affairs Office,
Qufu Normal University,
No. 80, Yantai Road, Donggang District, Rizhao, China
Email: 709403073@qq.com
*Corresponding author

Abstract: Movie recommendation is an important activity in the people's daily entertainment. Typically, through analysing the users' ever-watched movie list, a movie recommender system can recommend appropriate new movies to the target user. However, traditional movie recommendation techniques, e.g., collaborative filtering (CF) often face the following two challenges. First, as CF is essentially a traversal technique, the recommendation efficiency is often low. Second, traditional movie recommender systems often assume that the users' ever-watched movie list for decision-making is centralised, which makes it hard to be applied to the distributed movie recommendation scenarios. In view of these challenges, in this paper, we bring forth an efficient and privacy-aware online movie recommendation approach based on hashing technique. Through experiments on famous *MovieLens* dataset, we show that our proposal shows a better performance compared with other approaches in terms of recommendation efficiency and accuracy while users' private information is protected.

Keywords: movie recommendation; collaborative filtering; efficiency; privacy preservation; SimHash.

Reference to this paper should be made as follows: Shao, T. and Chen, X. (2020) 'Hash-based and privacy-aware movie recommendations in a big data environment', *Int. J. Embedded Systems*, Vol. 13, No. 1, pp.1–8.

Biographical notes: Tingting Shao is currently a Lecturer of the Medical Information Engineering School, Jining Medical University, China. She received her Bachelor's degree from School of Information Science and Engineering, Qufu Normal University, China, in 2002 and Master's degree from Medical Information Engineering School, Jining Medical University, China, in 2005. She is currently a member of a series of academic organisations, such as IEEE, IET and ACM. Her major research interests include intelligent information systems, web service recommendation and big data analyses.

Xuening Chen is currently a Lecturer of the Student Affairs Office, Qufu Normal University, China. She received her Bachelor's degree from Faculty of Education, Qufu Normal University, China, in 2006 and Master's degree from Department of Educational Psychology, East China Normal University, China, in 2009. She is currently a member of a series of academic associations, such as IEEE Computer Sciences Association, IET and ACM. Her research interests include intelligent decision systems, services computing, big data and privacy preservation.

1 Introduction

With the popularity of home entertainment, the volume and variety of movies both increase quickly, which results in difficult movie selection decisions for users (Naim et al., 2016; Zhang et al., 2017b). Under this circumstance, to alleviate such burdens, movie recommendation technique has become one of the indispensable daily entertainment tools. Typically, through analysing the movie lists ever-watched by users, a movie recommender system (e.g., collaborative filtering (CF)-based recommender system) can find out the similar friends (i.e., neighbours) of a target user and then make appropriate movie recommendation decisions to the target user (Zhang et al., 2017a). This way,

the target user's movie selection burden can be reduced significantly.

However, traditional movie recommendation approaches still face several problems or challenges as elaborated as below. First, CF is essentially a kind of 'traversal' technique for service recommendations. Therefore, the movie recommendation efficiency is often low. Under this circumstance, the volume of both users and movies, or the users' ever-watched movie list, is updated frequently with time elapsing, which blocks the online real-time movie recommendation for users severely. Second, the well-known CF-based movie recommendation approaches often assume that the ever-watched movie lists of users that are used for recommendation are stored in a centralised way. While actually, a user may watch movies or leave his/her ever-watched movie list data in different platforms. In this situation, the traditional centralised movie recommendation approaches cannot be directly applied to the distributed movie recommendation scenarios where the decision-making data are multi-source. Furthermore, to make comprehensive and accurate movie recommendation decisions, a movie recommender system needs to integrate the distributed decision-making data across different parties or platforms efficiently and properly while the above data integration process often faces the challenge of user privacy leakage, which blocks the feasibility of distributed movie recommendation severely.

In view of these challenges, we have proposed an original online movie recommendation approach, named $MR_{SimHash}$, based on a kind of hashing technique, i.e., SimHash (Caitlin and Levin, 2017). Our proposed $MR_{SimHash}$ approach can not only protect user privacy but also make efficient movie recommendation so that the online recommendation goal can be achieved.

Our major research contributions in this paper are two-fold:

1 We analyse the potential drawbacks (lack of privacy-preservation capability) of traditional movie recommendation approaches and correspondingly, propose a novel movie recommendation approach (named $MR_{SimHash}$) based on SimHash to improve the recommendation performances.

2 Experiments are conducted on a real-world movie rating dataset *MovieLens* to validate the feasibility of proposed $MR_{SimHash}$ method. Experiment results demonstrate that our proposal outperforms other cutting-edge movie recommendation approaches in terms of the accuracy and efficiency while protecting the private information of users.

The paper is structured as follows. In Section 2, we introduce the related work about service recommendation. In Section 3, the privacy-preserving movie recommendation problem is specified more formally. In Section 4, we describe our proposed movie recommendation approach $MR_{SimHash}$ that is based on a kind of hash technique, i.e., SimHash. In Section 5, several experiments are designed, deployed and tested depended on the well-known *MovieLens* dataset, to validate the effectiveness and accuracy of our proposed movie recommendation approach. Time complexity analyses are made in Section 6. In the end, in Section 7, the whole paper is concluded and the potential research directions are pointed out in the future work.

2 Related work

In this section, we introduce the related work associated with privacy-preserving service recommendation based on collaborative filtering (i.e., CF), from following two aspects: CF-based service recommendation and privacy-preserving service recommendation.

2.1 CF-based service recommendation

CF is regarded as an effective technique to make service recommendations in various recommender systems. Typical CF variants include item-based CF (Chung et al., 2014), user-based CF (Rong et al., 2014) and hybrid CF (Jiang et al., 2015). These CF-based recommendation approaches can look for the similar friends (or neighbours) of a target user or the similar services of a target service (i.e., the services preferred by the target user) based on the service set ever-invoked by users in the past and then recommend appropriate services to the target user based on the derived similar friends or similar services or their hybrids. As the service running environment is often dynamic, the running quality often depends on multiple context factors, such as service execution time and user location, a wide range of context-aware CF approaches are put forward to make more reasonable and accurate service recommendation, e.g., time-aware CF (Wang et al., 2016) and location-aware CF (Yu and Huang, 2016). However, there is a shortcoming in the abovementioned CF-based recommendation approaches, i.e., they only take the objective service usage data (e.g., users' ever-invoked service quality, users' ever-invoked service set and so on) into consideration, without considering other important factors that are crucial for recommendations, e.g., the subjective preferences of target users. To tackle this issue, a user preference-based CF recommendation method is suggested in Fletcher and Liu (2015), to support the target user's preference-aware service selection decisions.

However, the above mentioned CF-based service recommendation approaches usually depend on the assumption that the decision-making data are all locally stored, while neglecting the special cases in which the decision-making data are multi-source and the probable privacy concerns when the multi-source data are integrated or fused for comprehensive service recommendation.

2.2 Privacy- preserving service recommendation

To protect the sensitive user information during the service recommendation process, the authors in Dou et al. (2015) suggest publishing or releasing few service usage records, i.e., optimal quality to the outside world; this way, the

remaining user privacy is still protected by the user. While in Zheng et al. (2017), the trade-off relationship between data availability and recommendation accuracy is investigated. Furthermore, the released data amount was taken as a variable parameter by the authors and at last, the privacy-preserving service recommendation problem was converted into one associated with the parameter. However, although the above approaches can protect the private information of users to some extent, the partial privacy hidden in the released data is still not secure.

Encryption is an effective technique to protect the important sensitive information in many applications (Fu et al., 2015). Through encapsulating the sensitive plain text into less sensitive cipher text, the user privacy is protected well. However, encryption is a heavyweight data protection manner and hence often brings additional encryption cost and time delay and cannot suit the lightweight service recommendation requirements in most cases (Ma et al., 2015). Different from the above encryption-based data protection manner, anonymisation technique, e.g., the well-known K-anonymity (Ma et al., 2015) technique is often recruited to achieve the goal of protecting the user's privacy. However, when the decision-making data used to make service recommendation become anonymous, the service recommendation performance such as recommendation accuracy would be reduced accordingly.

To tackle this issue, an obfuscation strategy is proposed in Zhu et al. (2015) where obfuscated running quality (QoS) data is used to make recommendations without revealing the real QoS data to other parties involved in the multi-source data integration process. However, the recommendation accuracy is decreased as the obfuscated QoS data are taken as the decision-making basis of recommendation, instead of the real QoS data. Segment-based service recommendation technique is offered in Li et al. (2016), where each QoS record is divided into multiple segments which are sent to different users for storage; afterwards, the QoS segments are employed to make service recommendation. Generally, this approach can protect partial private information of users while the rest private data, e.g., service intersection of different users cannot be protected very well.

Differential privacy (DP)-based service recommendation approaches are proposed in Dou et al. (2016), Zhou et al., (2016), Li et al. (2017) and Zhu et al. (2017), where the noise data are injected into the original service quality data so as to make the real data confused. This way, the private information hidden in the original service quality data are secure. However, the time complexity of DP technique is often large therefore, these recommendation approaches cannot satisfy the users whose requirements are urgent.

Considering the drawbacks of existing CF-based recommendation approaches, an efficient and privacy-aware online movie recommendation method that is based on SimHash, i.e., $MR_{SimHash}$ is described in detail. Our proposal can work is based on the following assumption holds, i.e., the two users who watched the same movies in the past are possible friends or neighbours as they share the same or similar user preferences. The concrete recommendation approach is clarified in detail in Section 4.

3 Formalisation

Next, we will introduce the symbols to be used in the following discussions and further make a clear formulation on the privacy-preserving service recommendation problem with the help of the symbols.

1 u_{target} represents a target user who needs recommended movie list.

2 $U = \{u_1, \ldots, u_m\}$ denotes the user set.

3 $MOV = \{mov_1, \ldots, mov_n\}$ denotes the movie set.

4 if the user u_i ($u_i \in U$) has ever watched the movie mov_j ($movie_j \in MOV$), then the entry (depicts whether a user has ever watched a movie) corresponding to u_i and mov_j, denoted by $e_{i,j}$ is equal to 1

else, $e_{i,j}$ is equal to 0. Thus an m*n Boolean matrix as in (1) is obtained.

$$\begin{bmatrix} e_{1,1} & \cdots & e_{1,n} \\ \vdots & \ddots & \vdots \\ e_{m,1} & \cdots & e_{m,n} \end{bmatrix} \tag{1}$$

With the above symbols, we can formulate the private-preserving movie recommendation problem as below: according to the historical movie watch list associated with movies in set MOV and users in set U, recommend appropriate movies (never selected by u_{target}) to u_{target}, while guaranteeing the private information in (1). In the next section, we will put forward our proposal based on the SimHash technique.

4 Movie recommendation based on SimHash

Our $MR_{SimHash}$ approach is in Figure 1. There into, $H(u_i)$ is the hash value of user $u_i(u_i \in U)$ which is calculated according to SimHash. SimHash can be utilised to calculate the similarity between two sets without revealing the concrete elements in those sets. Therefore, SimHash can be recruited to make privacy-preserving similar neighbour finding.

Figure 1 Detailed steps of $MR_{SimHash}$

Step 1: According to the movie list ever-watched by users, build user indices offline based on SimHash. The user indices can be regarded as less sensitive.

Step 2: Find the neighbors of target user u_{target}. According to the less sensitive user indices generated in Step 1, we seek the target user's similar friends.

Step 3: Movie recommendation to the target user. Recommend appropriate movies to the target user according to the similar friends obtained in Step 2.

1 Step 1: According to the movie list ever-watched by users, build user indices offline based on SimHash.

For a user $u_i \in U$, if he/she has ever watched movie m_{ovj} ($\in MOV$), then we represent the corresponding entry (denoted by $e_{i,j}$) with Boolean value 1

Otherwise, $e_{i,j} = 0$. This way, we can transform u_i's ever-watch movie set with an n-dimensional Boolean vector, i.e., ($e_{i,1}$, …, $e_{i,n}$), where $e_{i,j} = 0$ or 1. As the number of movies is often large (i.e., n is large), in order to save computational time, we transform the high-dimensional vector ($e_{i,1}$, …, $e_{i,n}$) into a corresponding low-dimensional user index $H(u_i)$ approximately, based on SimHash technique.

Next, we introduce the concrete transformation process. Let parameter $r = \lceil \log_2^n \rceil$. Then each movie can be represented by an r-dimensional vector consisting of r Boolean values, based on the movie number. For example, suppose there are totally 100 movies and user u_i watched the first ten movies (i.e., mov_1, …, mov_{10}) in the past, i.e., $n = 100$, then $r = 7$ holds and the following transformations are obtained.

$mov_1 = (0, 0, 0, 0, 0, 0, 1)$

$mov_2 = (0, 0, 0, 0, 0, 1, 0)$

…

$mov_{10} = (0, 0, 0, 1, 0, 1, 0)$

Afterwards, for the above ten 7-dimensional vectors corresponding to user u_i, we replace the element '0' by '–1' so that we can obtain the following ten new 7-dimensional vectors.

$mov_1 = (-1, -1, -1, -1, -1, -1, -1)$

$mov_2 = (-1, -1, -1, -1, -1, 1, -1)$

…

$mov_{10} = (-1, -1, -1, 1, -1, 1, -1)$

Then we count the sum of each column of the matrix constituted by the above ten 7-dimensional vectors. After that, a 7-dimensional sum vector V in (2) is obtained. We replace the negative elements and positive elements in V_1 by '0' and '1', respectively. Then a new vector V' in (3) is obtained. In our proposal, V' is the index of u_i, i.e., $H(u_i) = V'$ holds. This way, we successfully transform the original 100-dimensional vector for u_i to be a short user index $H(u_i)$ which consists of only seven dimensions. Therefore, on one hand, the search efficiency of similar friends is improved significantly; on the other hand, as the less sensitive user index $H(u_i)$, instead of the real movie-watching records, is recruited for further movie recommendation, the private information of users are protected very well.

$$V = (-10, -10, -10, -4, -2, 0, 0) \tag{2}$$

$$V' = (0, 0, 0, 0, 0, 0, 0) \tag{3}$$

As the historical movie list ever-watched by each user $u_i (\in U)$ is already known by a certain platform, the index for u_i, i.e., $H(u_i)$ is generated offline before a target user input his/her movie recommendation request. This way, the recommendation process can be accelerated.

2 Step 2: Find the neighbours of target user u_{target}.

The indices for all the users in set U, including the index for u_{target}, i.e., $H(u_{target})$ have already been produced in the previous step. Next, we measure the Hamming distance (it depicts the difference of 0/1 bits between two multi-dimensional vectors) between $H(u_{target})$ and $H(u_i)$ $(1 \le i \le m)$, denoted by $D_{ham}(H(u_{target}), H(u_i))$.

Concretely, as $H(u_{target})$ and $H(u_i)$ are both r-dimensional vectors, we assume $H(u_{target}) = (p_1, …, p_r)$ and $H(u_i) = (q_1, …, q_r)$ where p_x and q_x $(1 \le x \le r)$ are both Boolean values. Next, the Hamming distance $D_{ham}(H(u_{target}), H(u_i))$ between u_{target} and ui can be derived by (4), where symbol '⊕' means the XOR operation between two values. Afterwards, according to the theory of SimHash (Rong et al., 2014), if condition $D_{ham}(H(u_{target}), H(u_i)) < 3$ holds, we can concluded that the ever-watched movie list of u_{target} is approximately the same as that of u_i; namely, u_{target} and u_i are possible friends. In this situation, we can make movie recommendations to u_{target} based on the ever-watched movie list of u_i, which will be discussed in detail in the next section.

$$\begin{aligned} &D_{ham}\left(H\left(u_{t\arg et}\right), H\left(u_i\right)\right) \\ &= \left(p_1 \oplus q_1, p_2 \oplus q_2, \ldots, p_r \oplus q_r\right) \end{aligned} \tag{4}$$

3 Step-3: Movie recommendation to the target user.

In Step-2, the neighbouring users of u_{target} have been obtained, denoted by set $Friend\_set(u_{target})$. In this step, we make appropriate movie recommendation to u_{target} based on the similar friends in $Friend\_set(u_{target})$. Concretely, for each movie $mov_j (\in MOV)$ never watched by u_{target}, we can predict u_{target}'s rating on mov_j, denoted as $Rating_{target,j}$ by (5) where set $U_j = \{u_i \mid u_i \in Friend\_set(u_{target})$ and u_i has ever watched movie $mov_j\}$, $Rating_{i,j}$ means u_i's rating on mov_j. Afterwards, we rank the movies $mov_j (\in MOV)$ in descending order according to their $Rating_{target,j}$ values and finally return the optimal movies to the target user.

$$Rating_{target,j} = \frac{1}{|U_j|} * \sum_{u_i \in U_j} Rating_{i,j} \tag{5}$$

5 Experiments

5.1 Experiment dataset and environment

A set of experiments are designed and tested based on a real movie rating dataset, i.e., *Movielens* (http://grouplens.org/

datasets/movielens/100k/) to prove the effectiveness of the approach. To validate the advantages of our proposed $MR_{SimHash}$ approach, we compare $MR_{SimHash}$ with three related approaches, i.e., UPCC (Breese et al., 1998), P-UIPCC (Zhu et al., 2015) and PPICF (Li et al., 2016). In the experiments, both time complexity and recommendation accuracy [i.e., MAE (Uddin et al., 2011; Sood and Loguinov, 2011), smaller is better] are measured by different methods. We randomly delete 5% ratings from *Movielens* dataset and utilise the rest 95% known ratings to predict the missing 5% ratings; afterwards, through comparing the predicted 5% rating data with the real 5% rating data, we can measure the MAE values of different competitive methods.

Experiment environment is as below: 2.80 GHz processor, 8.0 GB RAM, Windows 10 and Java 8. We repeat each test process 10 times and the average values are adopted finally.

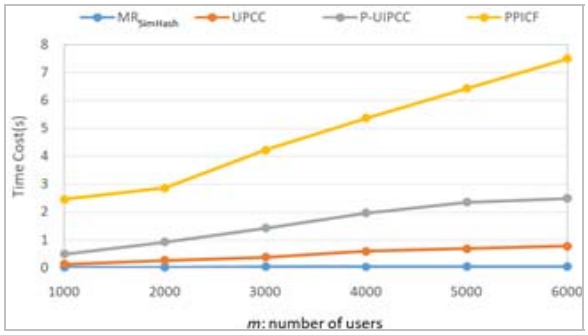
5.2 Experiment results

In this subsection, the performances of different methods are compared. Concretely, four test profiles are compared and analysed; m, n represent the size of user set and the size of movie set recruited in the experiment dataset *Movielens*, respectively.

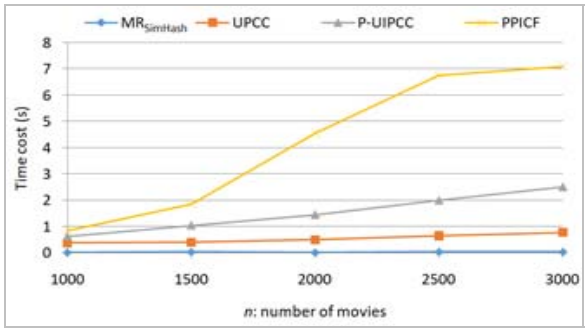
Profile 1: Time cost of four approaches.

The parameter information recruited in this test profile is as follows: m = {1,000, …, 6,000}, n = {1,000, …, 3,000}. Running results are presented in Figure 2.

Figure 2 Time costs of four approaches, (a) n = 3,000 (b) m = 6,000 (see online version for colours)



(a)



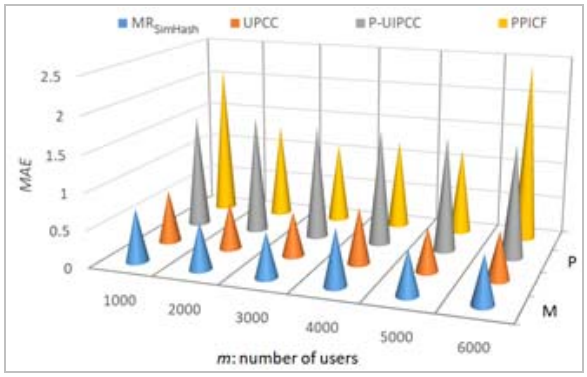
(b)

As Figure 2(a) (n = 3,000 holds) and Figure 2(b) (m = 6,000 holds) show, the recommendation efficiency of UPCC, P-UIPCC and PPICF all decrease when m or n rises, due to the fact that each user and each movie should take part in the movie recommendation process. While in $MR_{SimHash}$, the job of user indices building can be done offline before the recommender system begins to make movie recommendation. Therefore, the movie recommendation efficiency of our approach outperforms that of the rest three approaches greatly, which indicates that $MR_{SimHash}$ can deliver a good recommendation speed especially in the big data environment.

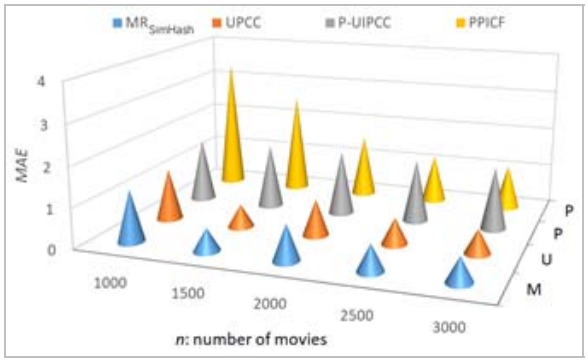
Profile 2: MAE comparison.

Accuracy is a popular measurement criterion for evaluating the performance of a recommender s_{ystem}. Therefore, to prove the effectiveness of MRSimHash, we need to test and compare the MAE values of different competitive methods. The parameter information recruited in this test profile is as follows: m = {1,000, …, 6,000}, n = {1,000, …, 3,000}. Running results are presented in Figure 3.

Figure 3 MAE comparison results, (a) n = 3,000 (b) m = 6,000 (see online version for colours)



(a)



(b)

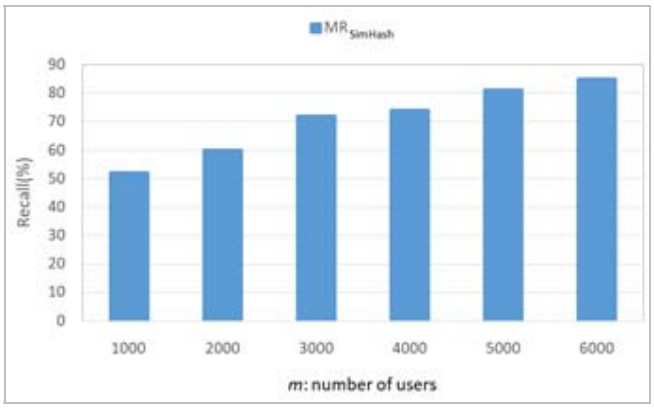
Figure 3(a) (n = 3,000 holds) and Figure 3(b) (m = 6,000 holds) show that the movie recommendation accuracy values of P-UIPCC and PPICF are relatively not high (i.e., *MAE* value is large), which is due to the fact that approximate operations are taken to realise privacy-preservation. Although these approximation techniques can protect user privacy, they decrease the movie

recommendation accuracy to some extent. In our $MR_{SimHash}$ approach, the inherent characteristics of SimHash can guarantee that the most similar friends of a target user can be found and recruited to make further movie recommendation. Therefore, the recommendation accuracy of our approach is improved significantly and stays approximately the same as the benchmark approach UPCC. This means that the recommendation accuracy of $MR_{SimHash}$ is acceptable.

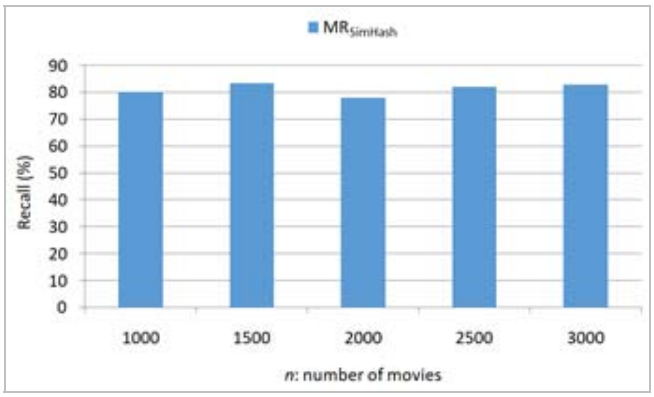
Profile 3: Recommendation recall of $MR_{SimHash}$ with respect to m and n.

Recall is an important criterion to evaluate the performance of a recommender system (Cremonesi et al., 201; Weng and Chang, 2008; Yu et al., 2006; Huang et al., 2007). Therefore, in this profile, we measure the recall values of our proposed movie recommendation approach $MR_{SimHash}$ when parameters m and n vary. Here, the recall value is calculated by the ratio of the number of recommended movies with 4* or 5* and the number of movies preferred (with ratings of 4* or 5*) by the target user. The parameter information recruited in this test profile is as follows: m = {1,000, …, 6,000}, n = {1,000, …, 3,000}. Running results are presented in Figure 4.

Figure 4 Recall values of $MR_{SimHash}$ with respect to m and n, (a) n = 3,000 (b) m = 6,000 (see online version for colours)



(a)



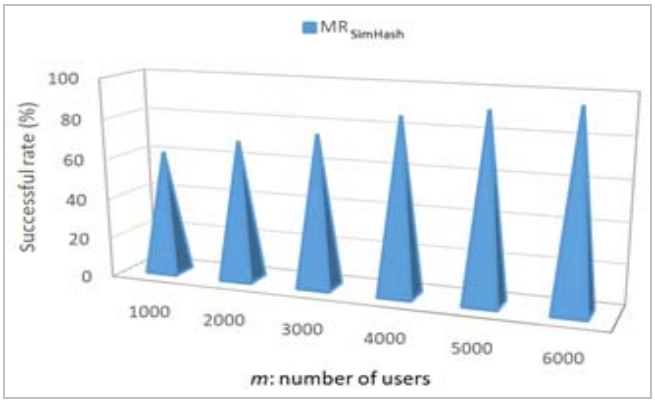
(b)

As indicated in Figure 4(a), the recommendation recall value of suggested $MR_{SimHash}$ approach rises when m grows linearly approximately; this is because all the m users should be compared with the target user to determine the neighbouring users of u_{target}. However, as shown in Figure 4(b), the recommendation recall value stays approximately stable when n rises as the users' ever-watched movie list information about the n movies is transformed into a r-dimensional vector already. Therefore, the recommendation recall value of $MR_{SimHash}$ is not related to n directly.

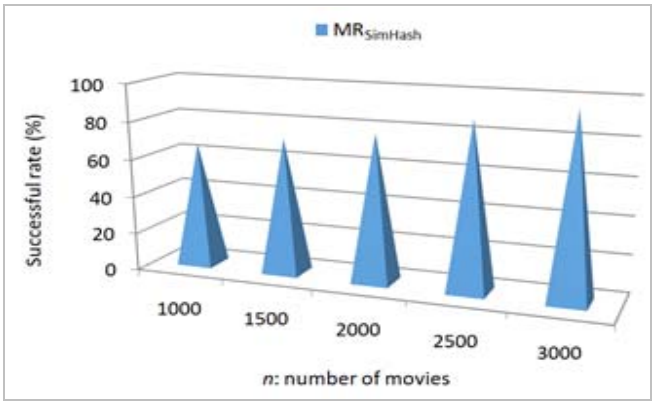
Profile 4: Recommendation successful rate of $MR_{SimHash}$ with respect to m and n.

As SimHash is actually a probability-based neighbour search technique, $MR_{SimHash}$ cannot always guarantee to return the target user a satisfying recommended result. In view of this observation, here we test the successful rate of our approach with m and n, respectively. Here, the successful rate is defined as the ratio between the number of successful recommendation times and the total recommendation times. The parameter information recruited in this test profile is as follows: m = {1,000, …, 6,000}, n = {1,000, …, 3,000}. Running results are presented in Figure 5.

Figure 5 Successful rate of $MR_{SimHash}$ with respect to m and n, (a) n = 3,000 (b) m = 6,000 (see online version for colours)



(a)



(b)

As shown in Figure 5, the recommendation successful rate of $MR_{SimHash}$ rises when m or n grows. This is due to the fact that when the user-movie watch matrix [see (1)] becomes denser (i.e., when m or n becomes larger), more neighbours of u_{target} can be found and utilised for further movie recommendation process and therefore, the successful rate of $MR_{SimHash}$ is improved accordingly.

6 Conclusions and future work

Movie recommendation is an important activity in people's daily entertainment. Typically, through analysing the users' ever-watched movie list, a movie recommender system can find out the neighbours of a target user and then recommend appropriate new movies to the target user. However, traditional movie recommendation methods often face the challenges of low efficiency and privacy leakage. In view of these challenges, we propose an efficient and privacy-aware online movie recommendation approach based on SimHash technique, i.e., $MR_{SimHash}$. Through extensive experiments deployed on well-known *MovieLens* dataset, the effectiveness and efficiency of $MR_{SimHash}$ are demonstrated. However, in this paper, we only recruit the ever-watched movie list data for recommendation decisions, without considering the concrete user rating data as used in work (Qi et al., 2017; Xu et al., 2017; Xu et al., 2018; Qi et al., 2018). Therefore, in the future work, we will discuss how to integrate these two different types of data effectively to pursue more accurate and comprehensive recommended results.

Acknowledgements

This paper is partially supported by Research on Data Mining Technology for the Healthcare Big Data (No. JY2016KJ034Y).

References

Breese, J.S., Heckerman, D. and Kadie, C. (1998) 'Empirical analysis of predictive algorithms for collaborative filtering', *The Fourteenth conference on Uncertainty in Artificial Intelligence*, pp.43–52.

Caitlin, S. and Levin, G. (2007) *Simhash: Hash-based Similarity Detection* [online] https//www.googlecode.com/sun/trunk/paper/SimHashwithBib.pdf (accessed 1 May 2018).

Chung, K., Lee, D. and Kim, K.J. (2014) 'Categorization for grouping associative items using data mining in item-based collaborative filtering', *Multimedia Tools and Applications*, Vol. 71, No. 2, pp.889–904.

Cremonesi, P., Yehud, K. and Roberto, T. (2010) 'Performance of recommender algorithms on top-n recommendation tasks', *Proceedings of the fourth ACM conference on Recommender systems*.

Dou, K., Guo, B. and Kuang, L. (2017) 'A privacy-preserving multimedia recommendation in the context of social network based on weighted noise injection', *Multimedia Tools and Applications*, pp.1–20.

Dou, W., Zhang, X., Liu, J. and Chen, J. (2015) 'HireSome-II: towards privacy-aware cross-cloud service composition for big data applications', *IEEE Transactions on Parallel and Distributed Systems*, Vol. 26, No. 2, pp.455–466.

Fletcher, K.K. and Liu, X.F. (2015) 'A collaborative filtering method for personalized preference-based service recommendation', *IEEE International Conference on Web Services*, pp.400–407.

Fu, Z., Shu, J., Wang, J., Liu, Y. and Lee, S. (2015) 'Privacy-preserving smart similarity search based on simhash over encrypted data in cloud computing', *Journal of Internet Technology*, Vol. 16, No. 3, pp.453–460.

Huang, Z., Zeng, D. and Chen, H. (2007) 'A comparison of collaborative-filtering recommendation algorithms for e-commerce', *IEEE Intelligent Systems*, Vol. 22, No. 5, pp.68–78.

Jiang, C., Duan, R., Jain, H.K., Liu, S. and Liang, K. (2015) 'Hybrid collaborative filtering for high-involvement products: a solution to opinion sparsity and dynamics', *Decision Support Systems*, Vol. 79, pp.195–208.

Li, C., Palanisamy, B. and Joshi J. (2017) 'Differentially private trajectory analysis for points-of-interest recommendation', *IEEE International Congress on Big Data*.

Li, D., Chen, C., Lv, Q., Shang, L., Zhao, Y., Lu, T. and Gu, N. (2016) 'An algorithm for efficient privacy-preserving item-based collaborative filtering', *Future Generation Computer Systems*, Vol. 55, pp.311–320.

Ma, T., Zhang, Y., Cao, J., Shen, J., Tang, M., Tian, Y., Al-Dhelaan, A. and Al-Rodhaan, M. (2015) 'KDVEM: a k-degree anonymity with vertex and edge modification algorithm', *Computing*, Vol. 70, No. 6, pp.1336–1344.

Naim, H., Aznag, M., Quafafou, M. and Durand, N. (2016) 'Probabilistic approach for diversifying web services discovery and composition', *IEEE International Conference on Web Services*, pp.73–80.

Qi, L., Wang, R., Li, S., He, Q., Xu, X. and Hu, C. (2018) 'Time-aware distributed service recommendation with privacy-preservation', *Information Sciences*, DOI: 10.1016/j.ins.2018.11.030.

Qi, L., Xiang, H., Dou, W., Yang, C., Qin, Y. and Zhang, X. (2017) 'Privacy-preserving distributed service recommendation based on locality-sensitive hashing', *IEEE International Conference on Web Services*, pp.49–56.

Rong, H. et al. (2014) 'User similarity-based collaborative filtering recommendation algorithm', *Journal on Communications*, Vol. 35, No. 2, pp.16–24.

Sood, S. and Loguinov, D. (2011) 'Probabilistic near-duplicate detection using simhash', *International Conference on Information and Knowledge Management*, pp.1117–1126.

Uddin, M.S., Roy, C.K., Schneider, K.A. and Hindle, A. (2011) 'On the effectiveness of simhash for detecting near-miss clones in large scale software systems', *18th Working Conference on Reverse Engineering*, pp.13–22.

Wang, X. et al. (2016) 'A spatial-temporal QoS prediction approach for time-aware web service recommendation', *ACM Transactions on the Web*, Vol. 10, No. 1, pp.1–25.

Weng, S.S. and Chang, H.L. (2008) 'Using ontology network analysis for research document recommendation', *Expert Systems with Applications*, Vol. 34, No. 3, pp.1857–1869.

Xu, X., Fu, S., Qi, L., Zhang, X., Liu, Q., He, Q. and Li, S. (2018) 'An IoT-oriented data placement method with privacy preservation in cloud environment', *Journal of Network and Computer Applications*, Vol. 124, pp.148–157.

Xu, Y., Qi, L., Dou, W. and Yu, J. (2017) 'Privacy-preserving and scalable service recommendation based on SimHash in a distributed cloud environment', *Complexity*, Vol. 2017, Article ID 3437854, 9pp.

Yu, C. and Huang, L. (2016) 'A web service qoS prediction approach based on time- and location-aware collaborative filtering', *Service Oriented Computing and Applications*, Vol. 10, No. 2, pp.135–149.

Yu, Z., Zhou, X., Hao, Y. and Gu, Y. (2006) 'TV program recommendation for multiple viewers based on user profile merging', *User Modeling and User-adapted Interaction*, Vol. 16, No. 1, pp.63–82.

Zhang, J. et al. (2017a) 'Hybrid computation offloading for smart home automation in mobile cloud computing', *Personal and Ubiquitous Computing*, Vol. 22, No. 1, pp.121–134.

Zhang, X. et al. (2017b) 'MRMondrian: scalable multidimensional anonymisation for big data privacy preservation', *IEEE Transactions on Big Data*, DOI: 10.1109/TBDATA. 2017.2787661.

Zheng, X., Cai, Z., Li, J. and Gao, H. (2017) 'Location-privacy-aware review publication mechanism for local business service systems', *IEEE International Conference on Computer Communications*, pp.1–9.

Zhou, P., Zhou, Y., Wu, D. and Jin, H. (2016) 'Differentially private online learning for cloud-based video recommendation with multimedia big data in social networks', *IEEE Transactions on Multimedia*, Vol. 18, No. 6, pp.1217–1229.

Zhu, J., He, P., Zheng, Z. and Lyu, M.R. (2015) 'A privacy-preserving QoS prediction framework for web service recommendation', *IEEE International Conference on Web Services*, pp.241–248.

Zhu, T., Li, G., Zhou, W., Xiong, P. and Yuan, C. (2017) 'Privacy-preserving topic model for tagging recommender systems', *Knowledge and Information Systems*, Vol. 46, No. 1, pp.33–58.