YANA: An Efficient Privacy-Preserving Recommender System for Online Social Communities

Dongsheng Li†, Qin Lv§, Li Shang‡, Ning Gu†

†School of Computer Science
Fudan University
Shanghai 200433 P.R.China
dongshengli@fudan.edu.cn
ninggu@fudan.edu.cn

§CS Department
University of Colorado at Boulder
Boulder, CO 80309 USA
qin.lv@colorado.edu

‡ECEE Department
University of Colorado at Boulder
Boulder, CO 80309 USA
li.shang@colorado.edu

ABSTRACT

Many recommender systems use collaborative filtering, a method that makes recommendations based on what are liked by other users with similar interests. Serious privacy issues may arise in this process, especially for online social communities, as sensitive personal information (e.g., content interests) may be collected and disclosed to other parties. In this paper, we propose *YANA* (short for "you are not alone"), a group-based content recommender system for online social communities, which protects users' interest privacy via interest-based groups and pseudo users. We have developed a prototype system on desktop and mobile devices, and evaluated it using real-world data. The results demonstrate that YANA can effectively protect users' privacy, while achieving high recommendation quality and energy efficiency.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information filtering; K.4.1 [**Public Policy Issues**]: Privacy

General Terms

Algorithms, Experimentation

Keywords

Collaborative filtering, Privacy, Efficiency

1. INTRODUCTION

Many recommender systems [3, 2, 6]) adopt *collaborative filtering* (CF), a popular recommendation method that has high accuracy, low overhead, and is generally applicable to various domains. In CF-based systems, the server collects user information and predicts a user's interest on an item based on the decisions (or ratings) of other similar users. In this process, users' personal interests are exposed to the recommender server, which raises several privacy concerns.

CIKM'11, October 24–28, 2011, Glasgow, Scotland, UK.

Recent research efforts towards privacy-preserving collaborative filtering (PPCF) have generated solutions in two main categories. One is based on secure multi-party computation (SMPC) [5, 9], and the other is based on randomization [4, 10]. SMPC-based methods require a large amount of computation and communication in order for users to jointly compute a value (e.g., overall rating for an item) without disclosing their individual values. This is inefficient for online social communities, where the number of users and items can be millions or even billions. What is more, many users tend to access online social communities via their smartphones or tablet PCs, which have limited computation capability and battery capacity. The efficiency of PPCF is thus critical. Randomization-based methods trade accuracy for privacy, which means users will receive lower-quality recommendation in order to protect their privacy.

In this paper, we propose *YANA* (short for "you are not alone"), a privacy-preserving content recommender system for online social communities. YANA can protect user privacy, while at the same achieving high recommendation quality and energy efficiency. YANA is group based – it automatically organizes users into groups with diverse interests such that each user's private interests can be hidden among a set of users. A number of pseudo users are created for each group, each representing a unique interest and the union of them covers all interests of the group. The pseudo users communicate with the recommender server on behalf of the real users. The real users can then obtain personalized recommendations based on the server's recommendations to the pseudo users, without exposing their private data to the server. To the best of our knowledge, this is the first work that targets efficient privacy-preserving collaborative filtering for recommender systems in online social communities.

2. SYSTEM OVERVIEW

Targeting the large-scale users and items in online social communities, YANA is designed to be highly efficient and scalable. As illustrated in Figure 1, YANA consists of three key components:

- **User groups**. YANA automatically organizes users into groups with diverse content interests, and individual users' content interests are hidden and aggregated within each group. Thus, user privacy is protected from the server. Inside each user group, users collaborate via privacy-preserving mechanisms, including efficient secure multi-party computation (SMPC), to protect user privacy from being inferred by other members in the group.

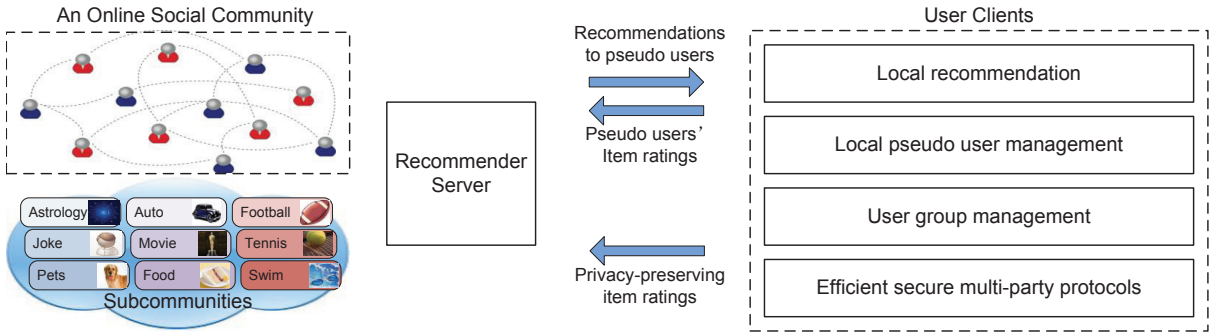

Figure 1: YANA: Group-based privacy-preserving recommendation for online social communities.

• **Pseudo users**. To obtain recommendations, the users in a group maintain a set of local pseudo users, each of which represents a unique interest liked by at least one group member. The pseudo users together cover all interests of the group members. The server makes recommendations to the pseudo users based on their interests, and the real users can re-calculate their personalized recommendations based on their own interests distribution and the server's recommendations to the pseudo users.

• **Recommendation algorithm**. To make recommendations, the server first needs to collect users' item ratings. We achieve this through efficient secure multi-party computation inside each group, and the aggregated decisions of each group are sent to the server via pseudo users. The server then runs the proposed collaborative filtering algorithm to make recommendations to the pseudo users. These recommendations to pseudo users are used by the real users to calculate their own personalized recommendations.

3. USER GROUPING

3.1 User Group Definition

We leverage *user groups* to hide individual users' content interests among a set of users with diverse interests, such that no one can associate any specific interest with a particular user. A user group is defined as following:

DEFINITION 1 (USER GROUP). *A user group g is a 4-tuple: $\{\mathcal{U}_g, \mathcal{WU}_g, \mathcal{S}_g, \mathcal{P}_g\}$, in which $\mathcal{U}_g$ is a set of users who have joined g and collaborate together to hide the privacy of each other, $\mathcal{WU}_g$ is a set of users who want to join the group (i.e., on the waiting list of g), $\mathcal{S}_g$ is the set of interests that users in $\mathcal{U}_g$ have, and $\mathcal{P}_g$ is a set of pseudo users who communicate with the server on behalf of the real users.*

3.2 User Group Construction

Given the user group definition above, we propose a method to automatically organize users into groups with diverse interests in a distributed and privacy-preserving way. The basic process of user group construction works as follows: 1) some random users are chosen to host groups; 2) other users choose the groups that are hosted/joined by their friends; 3) users in groups check if their privacy requirements (e.g., the number of users and the number of interests inside a group) are satisfied via distributed privacy-preserving computation.

Condition checking in Step 3) can be reduced to computing the sum of n real values privately held by n parties. To achieve secure n-party computation, we adopt an efficient secure multi-party summation protocol **SecureSum** [8]. The basic idea is as follows: 1) users randomly divide their local value into r parts with the property that the sum of the r parts equals their local value; 2) users randomly send/receive parts to/from other random users (i.e., random shuffling of the split parts); 3) after some rounds, each user sums its local parts and sends the sum to a host; 4) the host sums all values and returns the final result to all users.

4. PSEUDO USER MANAGEMENT

After user grouping, pseudo users are generated to protect user privacy and provide recommendations to real users. Real users interact with servers through the pseudo users, and all the information sent to the server by the pseudo users are the aggregated results of a group of users, so that the server cannot identify the information of individual real users. Each pseudo user acts as a "delegate" for a particular interest, and the recommendations to the pseudo user can be utilized by real users who have that interest.

Local pseudo users are formed based on the interests of the users inside each group. Let $\mathcal{S}_g$ be the set of interests for a given group g. Then the group members will construct $|\mathcal{P}_g| = |\mathcal{S}_g|$ pseudo users. For each pseudo user p with interest s_p, the items that p likes is the union of the items liked by users in the group who have interest s_p. One issue in maintaining local pseudo users is to update the interest profiles of the pseudo users as new items are generated in online social communities. This step is important, as we rely on the pseudo users' item decisions (or ratings) to make recommendations to pseudo users (with similar interest) in other user groups. Given a new item x, whether x is liked by pseudo user p depends on the number of real users who share interest with p and like x:

$$w_{x,p} = \sum_{u \in \mathcal{U}_g \ and \ p \in \mathcal{P}_g} w_{x,u} * \lambda(u, s_p) \qquad (1)$$

where $w_{x,u} = 1$ if u has read x, and 0 otherwise. s_p is the interest of pseudo user p, and $\lambda(u, s_p) = |X_{s_p} \cap X_u|/|X_u|$ (where X_u is the set of items that u is interested in). Equation 1 can be computed securely by the **SecureSum** protocol among a group of users. Item x is then assigned to the pseudo user with the highest non-zero $w_{x,p}$ value.

5. PRIVACY-PRESERVING CONTENT RECOMMENDATION

After the generation of local pseudo users, the server can collect the interest profiles of local pseudo users of all groups. The server can then make recommendations to the pseudo users based on the item ratings of other pseudo users. However, the use of pseudo users and user groups, while protect-

ing user interest privacy, introduces new challenges to the recommendation process:

• **Group ratings of new items.** Given a group g with K users. Each pseudo user represents a specific content interests, and some real users may have shared interests. How to measure the importance of an item within a group g is challenging. To determine the rating of an item inside a group, we need to consider two factors: 1) how the users in the group like it, and 2) the importance (or "expertise") of the users in the group for that kind of items. The first factor can be measured by $w_{x,p}$ in Equation 1, as higher value indicates that the item is more popular in the group. The second factor can be measured similarly as follows. Since the pseudo user p represents a specific interest s_p, the importance of user u with regard to s_p can be measured by the fraction of items liked by u that belong to s_p. Moreover, the weight of the users who do not like s_p will be 0, which means that the "non-experts" will not influence the rating of an item. Thus, the rating of item x by the users in group g can be computed as follows:

$$v_{x,g} = \sum_{u \in \mathcal{U}_g} \lambda(u, s_{\tilde{p}}) \tag{2}$$

where $\tilde{p} = \arg\max_{p \in \mathcal{P}_g}\{w_{x,p}\}$. The user who maintains pseudo user $\tilde{p}$ will send the pair $(w_{x,\tilde{p}}, v_{x,g})$ to the server.

• **Server-side recommendation.** The server cannot get a standard user-item matrix, which is required in the standard collaborative filtering algorithm. Thus, another challenge is how to adapt the standard collaborative filtering algorithm to work in the new form of data. The server generates the rating of item x for pseudo user q as follows:

$$\gamma(x,p) = \frac{\sum_{p' \in P} w_{x,p'} \times sim(p,p')}{\sum_{p' \in P} v_{x,g_{p'}} \times sim(p,p')} \tag{3}$$

where P is the set of all pseudo users, $g_{p'}$ is the group that pseudo user p' belongs to, $sim(p,p')$ is the Jaccard Similarity between pseudo user p and p'. $w_{x,p'}$ and $v_{x,g_{p'}}$ are obtained from the pseudo user p'. After the calculation of $\gamma(x,p)$, the server will recommend highly-rated items to pseudo user p.

• **Client-side recommendation.** In our system, the server makes recommendations only to the pseudo users, each of which stands for a unique interest. But the real users may have diverse interests. How to generate personalized recommendations for each real user based on the recommendations for the pseudo users is another challenge. As an item may have different ratings for different pseudo users, and real users also have different levels of interests for different pseudo users. We combine the ratings as follows:

$$\hat{\gamma}(x,u) = \sum_{p \in \mathcal{P}_g} \lambda(u, s_p) \times \gamma(x,p) \tag{4}$$

where $\mathcal{P}_g$ is the set of pseudo users in u's group, and s_p is the interest of pseudo user p.

6. EXPERIMENTAL RESULTS

We have developed a prototype system of YANA and conducted detailed evaluation using *Fudan* BBS [1], a popular online social community with mobile client support. YANA emphasizes high-quality, high-efficiency, and privacy-preserving content recommendation. We compare YANA with two state-of-the-art CF solutions. One is a privacy-preserving SVD-based collaborative filtering framework proposed by Canny [5]. The other is a probabilistic latent semantic indexing (PLSI) based collaborative filtering algorithm proposed by Hofmann [7], and later adopted by Google News [6].

6.1 Experimental Setup

YANA is implemented both on desktop and mobile devices using Java. The mobile client is implemented on HTC Magic smartphones. The HTC Magic runs Android 2.1 operating system with 528 MHz CPU and 288 MB Ram. It supports Wi-Fi of IEEE 802.11 b/g. And the battery capacity is 1340 mAh. The energy consumption of YANA is measured by monitoring the run-time battery capacity of the mobile phone. The recommendation server is also developed using Java on workstation, which collects pseudo users' information, clusters items, and computes item recommendations for the pseudo users.

YANA is tested with *Fudan* BBS, a popular online social forum among Chinese universities. It has over 60,000 users and supports various content-related user interactions, including posting, reading, and replying to articles and multimedia content. Everyday, there are approximately 20,000 new posts, and over 180,000 reads. *Fudan* BBS has over 100 subcommunities, and our experiments are conducted on 10 of the most popular subcommunities. We have collected data over three consecutive weeks, and divided the data into training set (Weeks 1 and 2) and testing set (Week 3). The training set is used to construct user groups and pseudo user profiles, and the testing set is used to evaluate the quality and efficiency of recommendations.

6.2 Recommendation Quality

Figure 2 shows the recommendation quality of YANA compared with the SVD and PLSI algorithms [1]. Please note that, higher precision at the same recall indicates better recommendation quality. As we can see, YANA outperforms PLSI in all the 10 subcommunities (by 23.2% on average), and outperforms SVD in 8 of the 10 subcommunities (by 16.3% on average), and achieves comparable quality in the other 2 subcommunities. The improvement is achieved through the accurate interest grouping and the novel interest based recommendation, which can accurately identify user interests, reduce false negative user decisions, and improve the overall recommendation quality of less popular items.

6.3 Recommendation Efficiency

To evaluate recommendation efficiency, we compare YANA with SVD, both of which are privacy-preserving collaborative filtering algorithms. The computation and communication complexities of SVD are both $O(k'm \log n)$ per user, where k' is the decomposition factor of SVD, m is the number of items, and n is the number of users. In contrast, the computation and communication complexities of YANA are both $O(km)$ per user. Since k' and k have similar order and n is usually big for online social communities, YANA can be much more efficient than SVD in terms of computation and communication complexities.

Latency. Figure 3 shows the overall latency of recommending each item to a user. We only consider the overall client-side latency as it would be the bottleneck of the whole system. Please note that, for the SVD solution, we cannot

[1]The number of users per group K is set to 10 in these experiments. We have also tested with different K values (from 10 to 50) and the quality results are similar.

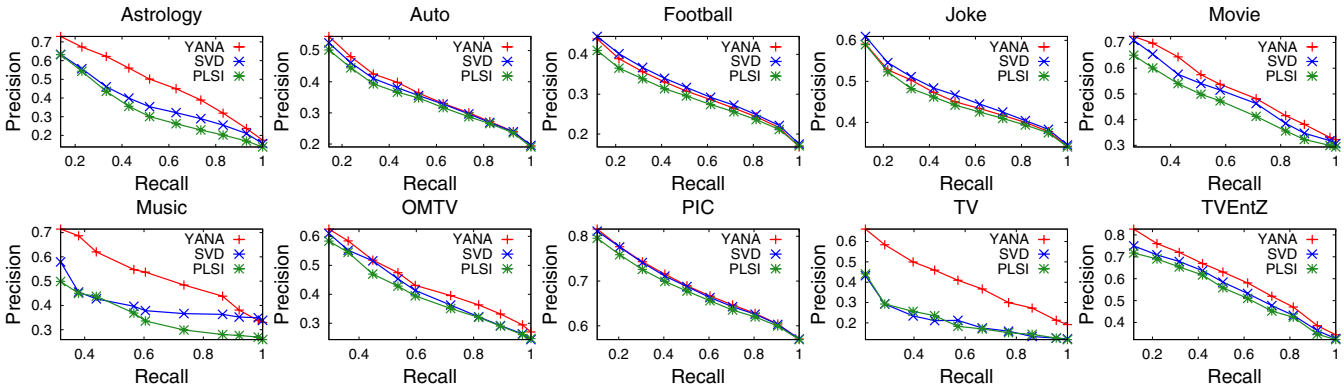

Figure 2: Recommendation quality comparison of YANA, SVD and PLSI in ten subcommunities.

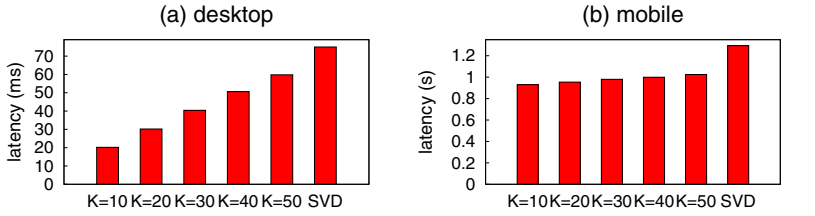

Figure 3: Comparison of recommendation latency (per user and per item) between YANA (with different group size K) and SVD.

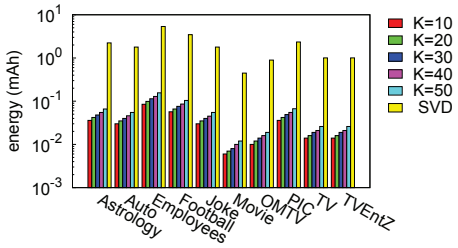

Figure 4: Daily energy consumption comparison of YANA and SVD.

measure its communication latency, as it requires a large-scale ($O(k'm \log n)$) peer-to-peer communication. Thus, we only measure the computation latency of the clients of SVD. Still, we can see that YANA outperforms SVD both on desktop and mobile devices in all the cases by at least 30%. The latency of YANA on desktop is between 20 and 60 milliseconds, which is fairly efficient. The latency on mobile devices is around 1 second, which is also reasonable and practical for users of mobile devices.

Energy Efficiency. Figure 4 shows the daily energy consumption of YANA mobile clients in the ten subcommunities, as compared with SVD. The daily energy consumption of YANA ranges from less than $0.01mAh$ to $0.16mAh$, which is really small compared with the total capacity of the battery ($1340mAh$). In contrast, the energy consumption of SVD ranges from about $0.6mAh$ to $5.36mAh$, which is over 30 times higher than that of YANA. The high energy consumption of SVD is due to its encryption and decryption operations in the singular value decomposition process, which is both time consuming and energy expensive.

7. CONCLUSION

In this work, we propose YANA, a group-based privacy-preserving content recommender system for online social communities. YANA automatically organizes users into groups with diverse content interests, which help protect individual users' private interests from the server. Inside the groups, efficient secure multi-party computation protocols are adopted to ensure privacy among group members. Pseudo users are created within each group to communicate with the server on behalf of real users, and the recommendations they receive from the server can be re-calculated locally to provide customized recommendation for individual real users. We have developed a prototype system and evaluated it using real-world traces of an online social community. The experimental results demonstrate that YANA can protect user interest privacy, achieve better recommendation quality, and is much more efficient compared against state-of-the-art collaborative filtering solutions.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant No. 60736020 and No. 60803118, the Shanghai Leading Academic Discipline Project under Grant No. B114, and the National Science Foundation of USA under Grant No. CNS–0910995.

8. REFERENCES

[1] Fudan BBS. `http://bbs.fudan.edu.cn`.
[2] NetFlix. `http://www.netflix.com/`.
[3] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. on Knowl. and Data Eng.*, 17(6):734–749, 2005.
[4] S. Berkovsky, Y. Eytani, T. Kuflik, and F. Ricci. Enhancing privacy and preserving accuracy of a distributed collaborative filtering. In *RecSys '07*, pages 9–16, 2007.
[5] J. Canny. Collaborative filtering with privacy. In *S&P '02*, pages 45–57, 2002.
[6] A. S. Das, M. Datar, A. Garg, and S. Rajaram. Google news personalization: scalable online collaborative filtering. In *WWW '07*, pages 271–280, 2007.
[7] T. Hofmann. Latent semantic models for collaborative filtering. *ACM Trans. on Info. Sys.*, 22(1):89–115, 2004.
[8] D. Li, Q. Lv, H. Xia, L. Shang, T. Lu, and N. Gu. Pistis: A privacy-preserving content recommender system for online social communities. In *WI '11*, 2011.
[9] H. Polat and W. Du. Privacy-preserving top-n recommendation on horizontally partitioned data. In *WI '05*, pages 725–731. IEEE Computer Society, 2005.
[10] S. Zhang, J. Ford, and F. Makedon. A privacy-preserving collaborative filtering scheme with two-way communication. In *EC '06*, pages 316–323, 2006.