Differential Privacy for Collaborative Filtering Recommender Algorithm

Xue Zhu *
Department of Software Engineering, Shandong University
Department of Computer science, The university of Hong Kong
xuezhu26@cs.hku.hk

Yuqing Sun†
Department of Computer Science
Shandong University
sun_yuqing@sdu.edu.cn

ABSTRACT

Collaborative filtering plays an essential role in a recommender system, which recommends a list of items to a user by learning behavior patterns from user rating matrix. However, if an attacker has some auxiliary knowledge about a user purchase history, he/she can infer more information about this user. This brings great threats to user privacy. Some methods adopt differential privacy algorithms in collaborative filtering by adding noises to a rating matrix. Although they provide theoretically private results, the influence on recommendation accuracy are not discussed. In this paper, we solve the privacy problem in recommender system in a different way by applying the differential privacy method into the procedure of recommendation. We design two differentially private recommender algorithms with sampling, named Differentially Private Item Based Recommendation with sampling (DP-IR for short) and Differentially Private User Based Recommendation with sampling(DP-UR for short). Both algorithms are based on the exponential mechanism with a carefully designed quality function. Theoretical analyses on privacy of these algorithms are presented. We also investigate the accuracy of the proposed method and give theoretical results. Experiments are performed on real datasets to verify our methods.

Keywords

Recommendation, Collaborative Filtering, Inference Attack, Differential Privacy

1. INTRODUCTION

A recommender system (RS) is designed to provide suggestions on items to users, which is widely adopted in web based applications. For example, e-commerce systems, such as Amazon or Alibaba, often recommend goods to users for commercial purposes. The core technology of these recommendation systems is the collaborative filtering algorithm, which recommends a list of items to a user by learning patterns from user behaviors, which are stored in a *rating matrix*.

However, if an attacker has some auxiliary knowledge about a user purchase history, he/she can infer other information about this user. For example, Dwork proposes three types of inference attacks on user purchase records [2]. Such attack brings great threats to user privacy.

Privacy problem in recommender systems has attracted much attention from both academia and industry. Recently, the differential privacy method is introduced into recommendation algorithms and gets acknowledgement due to its solid theoretical results in [3, 8].

In this paper, we solve the privacy problem in recommender system in a different way by applying the differential privacy method into the process of recommendation. We consider some representative recommendation algorithms and design two algorithms: Differentially Private Item-based Recommendation (DP-IR for short) and Differentially Private User-based Recommendation(DP-UR for short), respectively. Since the similarity measurement is important in collaborative filtering, we investigate what kind of measurement is suitable for differential privacy mechanism. Compared with the previous work, we design a low sensitivity metric to measure the similarity between both items and users.

In DP-IR, for each item, it first computes a list of related items based on a rating matrix. Then for a target user, it computes a list of top related items according to his/her purchase history. We apply the exponential mechanism into the selection of related items based on a carefully designed similarity measurement with low sensitivity. Such item list satisfies differential privacy and is released to the user as recommendation. In DP-UR, for a specific user, it first computes the similarity between users against their purchase history in rating matrix and select a list of top related users. Then it computes a score for each item by the sum of

*The author was a Bachlor student at Shandong University when she did this work and now is a PhD candidate at the University of Hong Kong.

†The corresponding author.Email: sun_yuqing@sdu.edu.cn.
This work is supported by NSF China (61173140), SAICT Experts Program, Independent Innovation & Achievements Transformation Program (2014ZZCX03301) and Science & Technology Development Program of Shandong Province (2014GGX101046)

CODASPY New Orleans, LA March 9-11, 2016

DOI: http://dx.doi.org/10.1145/2875475.2875483

weighed scores rated by the related users. The items with high scores are selected as the recommendation list to the user. We employ the exponential mechanism in the process of choosing the related items.

Furthermore, we present theoretical analyses on privacy of these algorithms. Motivated by the recommendation purpose, we introduce a quantitative metric to verify the quality of each protection mechanism, which evaluates the quality of a recommended item with a score. Based on this quality function, we investigate the accuracy of the proposed method and give theoretical results. Experiments are performed on two real datasets to verify our methods.

The remainder of this is organized as follows. Section 1.1 presents related works and Section 2 introduces some basic notions and theorems used in this work. The two proposed differential privacy algorithms and theoretical analysis are presented in Section 3 and 4, respectively. Section 5 discusses the experiments. Finally, we conclude and discuss the future work.

1.1 Related Work

The most related work is the differential privacy method. Differential privacy, coined by Dwork [2], quantifies the privacy with the principle that an algorithm output should prevent any inference about the presence or the absence of a record in the algorithm input. It requires that for any random computation, the outcome should be nearly equal to the results no matter a record is inside a database or not. Taking the recommendation problem, a differentially private recommended result is not sensitive to any single user record. Thus, it is a perfect notion to prevent the inference attack on a recommender system. If a user's once purchase of an item does not cause an obvious change of a recommended list to that item, an attacker could not guess the purchased item with a high confidence just by observing the change of the output. There are two mechanisms to design a differential privacy algorithm [5], Laplace Noise Mechanism and Exponential Mechanism [3],[8].

McSherry et al [3] address the privacy issue in recommender systems using Laplace noise. They add Laplace noise to the movie average rating, user average rating and covariance matrix. Then the noisy matrixes are released and used in the current recommender algorithms. In order to reduce the influence of one user record changing, a weight is assigned to each user, say $w_u = 1/|e_u|$ for user u, where e_u is the number of the rated items by u. Thus the covariance matrix is computed as $Cov_{ij} = \sum_u w_u r_{ui} r_{uj}$, where r_{ui} means the rating score on item i by user u. However, it seems unreasonable since the more 'purchase activities' a user performs, the fewer contribution to the covariance matrix.

Moritz Hardt et al. [8] convert the recommendation problem into the *Matrix Completion* problem, which tries to recover the missing entries of a matrix under a partially given matrix with a subset of the entries are randomly sampled [8]. They give an (ϵ, δ) differential privacy approach to compute the low rank approximations of large matrices that contain sensitivity information about individuals. Each entry of the matrix is independently perturbed by a noise. A low rank approximation is then computed against the resulting matrix.

Zhu et at. propose a truncated similarity function in private neighbour selection so as to achieve differential privacy for neighbourhood-based collaborative filtering [14]. For any item i and two predefined parameters $w \in (0, 1)$ and $k \in N^+$, the other items are divided into two sets C_1 and C_2 according to their general similarities with i. Each item in C_1 has a larger similarity with i than $s_k(i, .) - w$ and each item in C_0 has a smaller similarity than $s_k(i, .) - w$, where $s_k(i, .)$ denotes the similarity between i and the k^{th} similar item. Then they adopt the exponential mechanism to select the top k similar items with i under different possibilities. For C_1, the items are directly selected based on their similarities. Differently, for C_0, all items are regarded as a unit one and when it was selected, every item in C_0 has the same probability to be selected as the representative of C_0. However they do not give a detailed analysis about the privacy, especially the relationship between privacy and the parameter w. This would high influence the result since privacy depends on the setting of w. Also in the experiments they do not consider the relationship between error and ϵ, as well as the relationship between error and k.

Differently with the existing works, we employ the differential privacy method in the process of recommendation rather than on the data. An advantage of such choice is that it does not generate accumulative error. Another difference is that we present theoretical results on both privacy and accuracy.

2. PRELIMINARIES

In this section, we introduce the basic notions and theories used in this paper. Some of them are borrowed from [5],[2],[4]. Let M denote a $|U| \times |I|$ user rating matrix, where U is the user set and I is the item set in a recommender system, $M_{ui} \in \{1, 2, ..., R\}, R \in N^+$ is the rating score on item $i \in I$ given by user $u \in U$.

2.1 Differential Privacy

Differential Privacy is a privacy notion by 'hiding' one user's impact in the database [2] such that it can resist the inference attack on individual privacy based on some background knowledge. Suppose there is a query applied on database D, a recommendation algorithm satisfying differential privacy principle guarantees that if one user record is deleted or changed, the change of query result can be bounded by a predefined parameter. So an attacker can not make some inference by observing the change of released recommended item list even he knows some information .

Let $\mathcal{N}$ denote the set of $|U| \times |I|$ matrices, in which nonzero values are only allowed to occur in one row and its Euclidean norm is at most R. Formally, $\mathcal{N} = \{P : P \subset R^{|U|\times|I|}$ s.t there exists an index $u \in [1..|U|], ||P_u||_2 \leq R$ and $||P_j||_2 = 0, \forall j \neq u\}$ [8], where P_j is the j^{th} row of P. The semantics of notion $\mathcal{N}$ is that only one user's record exist in such a matrix. Other users' records are left empty.

Definition 1. *(Neighbouring Matrix) We say two user-item rating matrices $M, M' \in R^{|U|\times|I|}$ are neighbours if*

$$(M - M') \in \mathcal{N} \tag{1}$$

Based on the semantics of $\mathcal{N}$, $M - M'$ is a matrix that takes 0 at all entries except possibly in one single row, whose Euclidean norm is bounded by R. That is to say the two neighboring matrixes are different only on one user record.

DEFINITION 2. *((ϵ, δ)-Differential Privacy) A mechanism $\mathcal{A}$ is (ϵ, δ)-differential privacy if for all pairs of neighbouring rating matrices M and $M^{'}$, and for all running events $O \in range(\mathcal{A})$:*

$$Pr(\mathcal{A}(M) \in O) \leq exp(\epsilon) Pr(\mathcal{A}(M^{'}) \in O) + \delta$$

if $\delta = 0$, $\mathcal{A}$ satisfies ϵ-differential privacy.

DEFINITION 3. *(Sensitivity) Given a quality function q: $(R^{|U|\times|I|}, I) \rightarrow \mathbb{R}$, for any pair of neighbouring matrices $M, M^{'} \in R^{|U|\times|I|}$, $q(M, i)$ denotes the quality of recommending $i \in I$ under M. Its l_1 norm sensitivity is :*

$$\Delta_q = \max_{(M, M')} ||q(M, i) - q(M^{'}, i)||_1$$

Sensitivity is used to evaluate the error of the output of a differential privacy algorithm compared to the optimal output.

2.2 Exponential Mechanism

DEFINITION 4. *(Exponential Mechanism) Given a quality function $q : (R^{|U|\times|I|}, I) \rightarrow \mathbb{R}$, an input matrix M, Δ_q is the sensitivity of the quality function. the exponential mechanism $expo(M, I, q, \epsilon)$ outputs $i \in I$ with probability:*

$$Pr[expo(M, I, q, \epsilon) = i] = \frac{\epsilon q(M, i)/(2\Delta_q)}{\sum \epsilon q(M, i)/(2\Delta_q)}$$

satisfies ϵ-differential privacy.

The following combination theorem supports the case for applying a differential privacy method several times.

THEOREM 1. *(Combination Theorem) [11] For all $\epsilon, \delta, \delta^{'}$, the class of (ϵ, δ)-differential privacy mechanisms satisfy $(\epsilon^{'}, k\delta + \delta^{'})$-differential privacy under k-fold adaptive composition for:*

$$\epsilon^{'} = \sqrt{2k \ln(1/\delta^{'})}\epsilon + k\epsilon(e^{\epsilon} - 1) \quad (2)$$

THEOREM 2. *(Accuracy of the Exponential Mechanism) [5] Given an exponential mechanism $expo(M, I, q, \epsilon)$. Let $q(M, I)_{OPT} = \max_{i \in I} q(M, i)$, $I_{OPT} = \{i \in I : q(M, i) = q(M, I)_{OPT}\}$, $i^* = expo(M, I, q, \epsilon)$ then we have:*

$$Pr[q(M, i^*) \leq q(M, I)_{OPT} - \frac{2\Delta_q}{\epsilon}(\log(\frac{|I|}{|I_{OPT}|}) + t)] \leq e^{-t}$$

3. PROTECTION FOR ITEM-BASED RECOMMENDER SYSTEM

3.1 Item-based Recommender System and Inference Attack

There are several item-based recommendation algorithms. Deshpande proposes an item-based top-N recommendation algorithm [6]. Given a user-item rating matrix M and user u with purchase record $\mathcal{I}_u = \{\mathcal{I}_{u1}, \mathcal{I}_{u2}, ..., \mathcal{I}_{uw}\}$, for each item $i \in \mathcal{I}_u$, it firstly calculates the similarity S_{ij} between i and other item $j \in I$ as equation 3. Note that M_i and M_j represent the i^{th} and j^{th} column in M, respectively.

$$S_{ij} = \frac{M_i \cdot M_j}{R^2} \quad (3)$$

Algorithm 1 $\mathcal{A}_{nonpri}(M, \mathcal{I}_u)$

1: **Input**: user-item rating matrix M, purchase record $\mathcal{I}_u$ of user u, the length m of top related list.
2: $C = \varnothing$
3: **for all** $i \in \mathcal{I}_u$ **do**
4: **for all** $j \in I$ **do**
5: Calculate S_{ij} according to equation 3
6: **end for**
7: Select the top m items according to $S_{i1} \cdots S_{i|I|}$ and put them into L_i
8: **end for**
9: union the sets of related list L_i for each item $i \in \mathcal{I}_u$ $C = \bigcup_{i \in \mathcal{I}_u} L_i$;
10: candidate item set $C = C - \mathcal{I}_u$
11: score each item j in C by $s_j = \sum_{i \in \mathcal{I}_u} S_{ij}$
12: **Output**: the top-k recommended items in C to u

We first present the basic recommendation algorithm without privacy protection as shown in **Algorithm** 1.

For each item $i \in I_u$, a recommendation algorithm publishs the related list L_i of i. For example, *LibraryThing* publishes the related list for each book. The inference attack is against the released related list. Supposing an attacker knows some auxiliary information about a target user u, usually some part of the purchasing record $\mathcal{I}_u = \{\mathcal{I}_{u1}, \mathcal{I}_{u2}, ..., \mathcal{I}_{uw}\}$. Suppose one user u interacts with the system within the time period $[t_1, t_2]$ and purchase item m, which results in m is added to I_u. The covariance between m and all items in I_u must increase. Thus, the rank of m in the related list to any $i \in I_u$ grows. Then the attacker can infer the purchasing activity of u by observing these related lists of items in I_u. If the same item m appears or moves up in the related-item lists of a sufficiently large subset of the auxiliary items, the attacker can infer that u bought m.

3.2 Quality and Sensitivity

Since the related list of an item is the essential notion in a recommender system, we design the Differential Private Item-based Recommender Algorithm with sampling (DP-IR for short) on this list to solve the privacy problem. Although many recommender algorithms are designed based on the related list (without accessing other original data), the succeeding procedures do not compromise the differential privacy. The theoretical proof would be presented.

Taking the semantics of related list into account, the exponential mechanism is adopted to the list generation. The common challenge in exponential mechanism is to design an appropriate quality metric, which should reflect the essential of the problem and be not sensitive to trivial change. So we introduce the similarity function into $DP - IR$, as equation 3, which is consistent with user behaviours as other widely adopted similarity computation but less sensitive. We will analyze the rationality of this similarity measurement in **Section** 3.6. Now we introduce a quality function.

DEFINITION 5. *(quality function and sensitivity) Given a rating matrix M, an item i, the quality of the recommended item j to i is given by:*

$$q(M, i, j) = S_{ij} \quad (4)$$

Since $S_{ij} \in [0,1]$, for any pair of neighbouring matrices $M, M^{'}$, the sensitivity of the quality function is given as: $\Delta_{q(M,i,j)} = \max_{(M,M')} |q(M,i,j) - q(M^{'},i,j)| = 1$

From the above discussion, we can see that the similarity changes at most 1 when one user updates his purchase record.

3.3 Item-based Differential Privacy Recommender Algorithm

Based on the above notions, we design an Item-based Differential Privacy Recommender Algorithm with sampling (DP-IR for short). It consists of two steps:

- **Step 1:** Given a user u with purchase record $\mathcal{I}_u = \{\mathcal{I}_{u1}, \mathcal{I}_{u2}, ..., \mathcal{I}_{uw}\}$, we sample each user with probability p in M, and get the new user-item rating matrix T. Then we design a differential privacy algorithm $\mathcal{DP} - \mathcal{A}_1(T, c, \delta_0, \mathcal{I}_u) \rightarrow \mathcal{L}^{|\mathcal{I}_u|}$ to compute a series of related list $\mathcal{L}_i$ for each item $i \in \mathcal{I}_u$ based on T, which satisfies (c, δ_0)-differential privacy. The whole process is algorithm $\mathcal{DP} - \mathcal{A}'_1(M, c, \delta_0, \mathcal{I}_u, \epsilon)$ (based on the original data M) satisfying (ϵ, δ)-differential privacy,where $\delta = \epsilon\delta_0/2$.
- **Step 2:** We recommend k items to user u according to the related lists $\mathcal{L}_i$ got by step 1 and the purchase record $\mathcal{I}_u$ of user u, denoted by $\mathcal{R}(\mathcal{L}, \mathcal{I}_u)$.

We will show that the whole algorithm DP-IR:$R^{|U|\times|I|} \rightarrow I^k$ satisfies (ϵ, δ)-differential privacy, where $\delta = \epsilon\delta_0/2$.

3.3.1 Differential Privacy Related Items Recommender Algorithm

Firstly we present the differential privacy algorithm for related lists recommendation **Algorithm** $\mathcal{DP} - \mathcal{A}_1(M, c, \delta_0, \mathcal{I}_u)$, shown in **Algorithm 2**. Given a user u with purchase record $\mathcal{I}_u = \{\mathcal{I}_{u1}, \mathcal{I}_{u2}, ..., \mathcal{I}_{uw}\}$, we recommend m related items to u. Algorithm $\mathcal{DP} - \mathcal{A}_1$ applies exponential mechanism $m|\mathcal{I}_u|$ times. Suppose that we expect a (c, δ) differential privacy, in each fold we should set $\epsilon^{'} = \frac{c}{2\sqrt{2m|\mathcal{I}_u|\ln(1/\delta_0)}}$ according to **Theorem 1**.

3.3.2 Item-based Differential Privacy Recommendation with Sampling

Since the number of users is very large in a dataset, to reduce the complexity of computing item similarity $S(i,j)$, we sample users for calculation. Then the recommendation is based on this part of users as well as their purchase records. The sampling process plays an important role in our experiment based on the Netflix data set.

We construct an algorithm $\mathcal{DP} - \mathcal{A}'_1$ to recommend the related lists to user u based on purchase history $\mathcal{I}_u$, given by **Algorithm 3**. We sample each user with probability $p = \frac{\epsilon}{2}$ to get the new user set T and return $\mathcal{DP} - \mathcal{A}_1(T)$. We will show that $\mathcal{DP} - \mathcal{A}'_1$ satisfies $(\epsilon, \epsilon \cdot \delta_0/2)$-differential privacy.

After $\mathcal{DP} - \mathcal{A}^{'}$, we apply Algorithm $R(\mathcal{L}, \mathcal{I}_u) : \mathcal{L}^{|\mathcal{I}_u|} \rightarrow I^k$ given by **Algorithm 4** to recommend top k related items to u based on $\mathcal{L}$.

3.4 Analysis of Privacy

In this section, we analyse the privacy of DP-IR. Firstly we give some theorems.

Algorithm 2 $\mathcal{DP} - \mathcal{A}_1(M, c, \delta_0, \mathcal{I}_u)$

input: user u's purchase record $\mathcal{I}_u$, user-item rating matrix $M^{|U|\times|I|}$, $M_{ui} \in \{1, 2, .., R\}$, privacy paramaters (c, δ_0), parameter m for the number of items in each related list
for all $i \in \mathcal{I}_u$ **do**
  for all $j \in I$ **do**
    $S(i,j) = \frac{\sum_{u\in U} M_{ui} \times M_{uj}}{R^2}$
  end for
  Initialised: $\mathcal{L}_i = \emptyset, \mathbb{I} = I$
  for all $t = 1 : m$ **do**
    Sample an item $r \in \mathbb{I}$ with probability:

$$\frac{\exp\frac{\epsilon^{'} S(i,r)}{2\Delta}}{\sum_{j\in\mathbb{I}} \exp\frac{\epsilon^{'} S(i,j)}{2\Delta}}$$

    , where $\epsilon^{'} = \frac{c}{2\sqrt{2m|\mathcal{I}_u|\ln(1/\delta_0)}}$, $\Delta = 1$
    $\mathcal{L}_i = \mathcal{L}_i \cup r, \mathbb{I} = \mathbb{I} - \{r\}$
  end for
end for
Output:$\mathcal{L}_1, \mathcal{L}_2, ..., \mathcal{L}_{|\mathcal{I}_u|}$

LEMMA 1 (PRIVACY OF SAMPLING). *[9] Given a database D, suppose $A(D)$ is a $(1, \delta_0)$-differential privacy, we design an algorithm $A^{'}$ as: Firstly, we construct a set $V \subset D$ by selecting each element from D independently with probability ϵ^*; then return $A(V)$. So for any $\epsilon^* \in (0,1)$,$\mathcal{A}^{'}(\epsilon, D)$ satisfies $(2\epsilon^*, \epsilon^* \cdot \delta_0)$ differential privacy.*

Then the following theorem can be derived directly :

THEOREM 3 (THE PRIVACY OF ALGORITHM $\mathcal{DP} - \mathcal{A}'_1$). *Algorithm $\mathcal{DP} - \mathcal{A}'_1$ is $(\epsilon, \epsilon\delta_0/2)$-differential privacy.*

And we apply $\mathcal{R}(\mathcal{L}, \mathcal{I}_u)$ to the output of $\mathcal{DP} - \mathcal{A}'_1$, we will show that the Post-Processing doesn't make it less differentially private.

THEOREM 4 (PRIVACY OF POST-PROCESSING). *[12] Let $\mathcal{A}' : R^{|U||I|} \rightarrow L^{|\mathcal{I}_u|}$ be (ϵ, δ)-differential privacy and let $f : L^{|\mathcal{I}_u|} \rightarrow I^k$ be an arbitrary function.Then:*

$$f \circ \mathcal{A}^{'} : R^{|U||I|} \rightarrow \mathcal{I}^k$$

is (ϵ, δ)-differential privacy

It indicates that if the later procedure does not access the matrix M and just be applied on the output $\mathcal{L}$ derived from the former procedure, then it reserves differentially private.. That's to say even we apply algorithm $\mathcal{R}$ to the output of $\mathcal{DP} - \mathcal{A}^{'}$, we don't make the whole algorithm less differentially private. Finally, we get the following theorem:

THEOREM 5 (PRIVACY OF DP-IR). *DP-IR satisfis (ϵ, δ)-differential privacy, where $\delta = \epsilon\delta_0/2$*

Thus **Theorem 5** can be derived directly from **Theorem 3** and **Theorem 4**.

3.5 Analysis of Error

Now we analyze the accuracy of DP-IR. The error comes from two parts: sampling ($error_1$) and the exponential mechanism ($error_2$). Firstly we give the definition of (τ, θ)-accuracy.

Algorithm 3 $\mathcal{DP}-\mathcal{A}_1^{'}(M,c,\delta_0,\mathcal{I}_u,\epsilon)$

input: user u's purchase recoding $\mathcal{I}_u$, user-item rating matrix $M^{|U|\times|I|}$, $M_{ui}\in\{1,2,..,R\}$, privacy paramater (c,δ_0) where $c=1$, parameter ϵ, and parameter m for the number of items in each related list.
Sample each user with probability $p=\epsilon/2$ and get the new user set V.
for all $i\in\mathcal{I}_u$ **do**
  for all $j\in I$ **do**
    calculate

$$S^{'}(i,j)=\frac{\sum_{u\in V}M_{ui}\times M_{uj}}{R^2}\times\frac{1}{p}$$

  end for
  Initialised: $\mathcal{L}_i=\emptyset,\mathbb{I}=I$
  for all $t=1:m$ **do**
    Sample an item $r\in\mathbb{I}$ with probability:

$$\frac{\exp\frac{\epsilon^{'}S^{'}(i,r)}{2\Delta}}{\sum_{j\in\mathbb{I}}\exp\frac{\epsilon^{'}S^{'}(i,j)}{2\Delta}}$$

    . where $\epsilon^{'}=\frac{c}{2\sqrt{2m|\mathcal{I}_u|\ln(1/\delta_0)}}$, $\Delta=\frac{1}{p}$
    $\mathcal{L}_i=\mathcal{L}_i\cup r,\mathbb{I}=\mathbb{I}-\{r\}$
  end for
end for
Output:$\mathcal{L}=\{\mathcal{L}_1,\mathcal{L}_2,...,\mathcal{L}_{|\mathcal{I}_u|}\}$.

Algorithm 4 $R(\mathcal{L},\mathcal{I}_u)$

input: user $u's$ purchasing record $\mathcal{I}_u$, and the published related lists $\mathcal{L}=\{\mathcal{L}_1,\mathcal{L}_2,...,\mathcal{L}_{|\mathcal{I}_u|}\}\in L^{|I|}$,parameter k for the number of recommended items
output: the recommended item list $l\in L$ for u
For every item $t\in\mathcal{I}_u$, put all items $j\in L_t$ to C
Remove the items u already bought from C
For every item $j\in C$, count the frequency that it appears in $\mathcal{L}$ and order the items in descending order according to the frequency.
Select the top k items in C as the recommended item list l for u

Definition 6 (Error of Algorithm). *For an item i, suppose $\mathcal{DP}-\mathcal{A}_1^{'}$ recommends item r^* in the t^{th} inner- recommendation. The error of the recommendation is defined as:*

$$error=|S(i,i^*)-S(i,i_{OPT})| \tag{5}$$

The error of algorithm $\mathcal{DP}-\mathcal{A}'_1$ is defined as the difference between the final output and the optimal output. Algorithm $\mathcal{DP}-\mathcal{A}'_1$ is defined (τ,θ)-accuracy, if the error satisfies:

$$Pr[error\geq\tau]\leq\theta \tag{6}$$

Lemma 2. *(Additive Chernoff Bound) [10]*
Let $X_1,X_2,...,X_n$ be independent random variables, where $X_i\in(0,1),\forall i\in[1..n]$. Let $\bar{X}=\frac{1}{n}\sum_i^n X_i$ denote their mean, and let μ denote their expected mean. Then

$$Pr[\bar{X}-\mu\geq\alpha]\leq\exp(-2\alpha^2 n)$$

and also

$$Pr[\mu-\bar{X}\geq\alpha]\leq\exp(-2\alpha^2 n)$$

The error of an algorithm evaluates the distance with the optimal result. Based on this definition and the lemma, we propose the following theorem.

Theorem 6 (Accuracy of DP-IR). *For an item i, when algorithm $\mathcal{DP}-\mathcal{A}_1^{'}$ recommends an item to it, DP-IR is $(\mathcal{O}(\frac{\sqrt{m|\mathcal{I}_u|\ln(1/\delta_0)}(\log|I|+t)+\sqrt{|U|\ln|U|}}{\epsilon}),\frac{1}{|U|^2}+e^{-t})$-accuracy*

Proof. Let $Y=\{Y_1,Y_2,...,Y_{|U|}\}$, denote the variable with probability distribution:

$$Y_u=\begin{cases}\frac{M_{ui}\times M_{uj}}{R^2} & \text{p}\\ 0 & \text{1-p}\end{cases}$$

M_{ui} is the rating of u to i, $Y_u\in[0,1]$. In $\mathcal{DP}-\mathcal{A}_1^{'}$ we sample each user with probability p and get the user set V, then calculate

$$S^{'}(i,j)=\frac{\sum_{u\in V}M_{ui}\times M_{uj}}{R^2}\times\frac{1}{p}$$

, also $S^{'}(i,j)$ can be addressed as $\frac{1}{p}\sum_{u\in U}Y_u$. Let $E(Y)$ denote the expected mean. We have

$$E(Y)=p\frac{\sum_{u\in U}M_{ui}\times M_{uj}}{R^2|U|}=\frac{pS(i,j)}{|U|}$$

Then we have

$$Pr[\sum_{u\in U}Y_u/|U|-E(Y)\geq\alpha]\leq\exp(-2\alpha^2|U|) \tag{7}$$

and Pluging in $S^{'}(i,j)$ and $S(i,j)$, we get

$$Pr[S^{'}(i,j)-S(i,j)\geq\frac{|U|\alpha}{p}]\leq\exp(-2\alpha^2|U|) \tag{8}$$

In general, $|U|$ is large, thus if we let $\exp(-2\alpha^2|U|)\leq\frac{1}{|U|^2}$, we can make sure that the error is bound with great probability, and we get $\alpha=\sqrt{\frac{\ln|U|}{|U|}}$. Thus the error is:

$error_1=\frac{|U|\alpha}{P}=\frac{\sqrt{|U|\ln|U|}}{p}=\frac{2\sqrt{|U|\ln|U|}}{\epsilon}$

We have: $Pr[S(i,j)-S^{'}(i,j)\geq\frac{2\sqrt{|U|\ln|U|}}{\epsilon}]\leq\frac{1}{|U|^2}$ Here $S^{'}(i)_{OPT}=\max_{j\in I}S^{'}(i,j)$, $i_{OPT}=\{j\in I:S^{'}(i,j)=S^{'}(i)_{OPT}\}$. Note that $\Delta_q=1/p=2/\epsilon$, and $\epsilon^{'}=\frac{1}{2\sqrt{2m|\mathcal{I}_u|\ln(1/\delta_0)}}$, then according to **Theorem 2**, we get:

$$Pr[s^{'}(i,r^*)\leq S(i)^{'}_{OPT}-\frac{8\sqrt{2m|\mathcal{I}_u|\ln(1/\delta_0)}}{\epsilon}(\log(|I|)+t)]\leq e^{-t} \tag{9}$$

Note that we should define the error as:

$$error=|S(i,i^*)-S(i,i_{OPT})| \tag{10}$$

i^* is the item got by algorithm $\mathcal{DP}-\mathcal{A}_1^{'}$. Note that it's possible that $S^{'}(i,r)=S(i,r)+error_1$ and $S^{'}_{OPT}=S(i)_{OPT}-error_1$. The total error is the sum of the error produced by sampling($error_1$) and exponential mechanism($error_2$).

$$error=\frac{8\sqrt{2m|\mathcal{I}_u|\ln(1/\delta_0)}+4\sqrt{|U|\ln|U|}}{\epsilon}$$

And one of the two errors exceeds the bound the total error will exceed *error*. Thus,

$$\begin{aligned}Pr[E_1>error_1\vee E_2>error_2]&\leq Pr[E_1>error_1]+Pr[E_2>error_2]\\&=\frac{1}{|U|^2}+e^{-t}\end{aligned} \tag{11}$$

Thus we have :
let $\Gamma = S(i, i_{opt}) - S(i, i^*)$

$$Pr[\Gamma > \frac{8\sqrt{2m|\mathcal{I}_u| \ln(1/\delta_0)}(\log|I| + t) + 4\sqrt{|U| \ln|U|}}{\epsilon}] \leq \frac{1}{|U|^2} + e^{-t} \tag{12}$$

Note that the main part of the error is $\frac{4\sqrt{|U| \ln|U|}}{\epsilon}$, compared with the range $[0, |U|]$, $\sqrt{\frac{\ln|U|}{|U|}} \times \frac{4}{\epsilon}$ will be less than 1 when $\epsilon > 4\sqrt{\frac{\ln|U|}{|U|}}$.

3.6 Discussion on Quality Function

One key point in Item-based collaborative filtering is the item similarity measurement. There are three aspects should be considered when choosing a quality function. Firstly, a good similarity metric should reflect the essential of item correlations, which should be learned from user behaviours. In equation 3, the molecular reflects the fact that the more users who purchase both items i and j and give a high rating, the more similar i and j. Secondly, a good metric should not be sensitive to trivial change. Because in a differential privacy method, the accuracy is related to the sensitivity of the function. More details can be got from **Theorem** 2. Considering our similarity function, the biggest difference is only $1/|U|$, which is very small since the number of users is very large. Thirdly, the sensitivity is expected to decrease along with the increasing number of users. There are some other popular measurements to evaluate item similarity [7]. We would make some comparisons.

- The simplest and most common one is the Euclidean distance:

$$S(i, j) = (\sum_{u=1}^{|U|}(M_{ui} - M_{uj})^2)^{1/2} \tag{13}$$

  Here, $S(i, j) \in [0, \sqrt{|U|}R]$ and its most change is R on one change of a user, which derives the $(1/\sqrt{|U|})$-sensitivity. Thus we can define

$$q(M, i, j) = S(i, j) = \frac{(\sum_{u=1}^{|U|}(M_{ui} - M_{uj})^2)^{1/2}}{\sqrt{|U|}R} \tag{14}$$

- Another common approach is the cosine similarity:

$$cos(i, j) = \frac{M_i \cdot M_j}{||M_i||||M_j||} \tag{15}$$

  $cos(i, j) \in [0, 1]$. If one user changes his input , the quality score changes 1 at most, which derives the 1-sensitivity. Cosine similarity is not a feasible quality function in differential privacy. Firstly, according to **Theorem 2**, the error will be at almost $\frac{2}{\epsilon}\log|I|$, which is over the max value of its range; Secondly, its sensitivity can not decrease with the increasing of the user number. And the weakness of Pearson Coefficient is similar to cosine similarity.

  In our experiments , we compare some comparison with other similarity. And, we find that the Dot Similarity in equation 3 performs better in the recall rate.

4. PROTECTION FOR USER-BASED RECOMMENDER SYSTEM

4.1 User Based Recommender System and Inference Attack

The user based recommendation is another representative algorithm, which can be formalised as two steps [7]. For some user u, it firstly finds the k most similar users set $\mathcal{K}_u = \{U_{ui}, U_{u2}, ..., U_{uk}\}$ according to some similarity metric (e.g., the Pearson correlation coefficient or cosine similarity). The similarity matrix between users is represented as S, where each entry S_{uv} denotes the similarity between users u and v. Then for each user u, it predicates u's rating on item $i \in I$ according to purchase histories of these k-nearest neighbours. The predicting score of user u for item i can be computed by $r_{ui} = \frac{\sum_{n\in\mathcal{K}} S_{uv}M_{vi}}{\sum_{n\in\mathcal{K}} S_{uv}}$. Items are ranked in descending order according to the predicating score. The higher the score, the more user u is preferred.

Suppose that an attacker already has some background about a target user $u's$ purchase history, namely the n items and ratings by user u recorded in $I_u = \{I_{u1}, I_{u2}, ..., I_{uw}\}$. The purpose of this attack is to obtain knowledge about other unknown items that were also bought by u. The attack procedure includes two steps. Firstly, the attacker creates k sybil users and populates each sybil user's purchase history with his background information about u. Then the attacker tries to infer u's other purchasing record by playing as one of the sybil users and inspecting the recommended list. Any item in the ahead of this list might be bought by u with high probability.

4.2 Preference and Sensitivity

Given a user u with purchase record $\mathcal{I}_u = \{\mathcal{I}_{u1}, \mathcal{I}_{u2}, ..., \mathcal{I}_{uw}\}$, we recommend k items to u based on the neighbours of u. Firstly we define the similarity between two users. A user u vector $r_u \in \{1, 2, ..., R\}^{|I|}$ is chosen from the rating matrix, where $r_{ui} = M_{ui}$ if user u has rated item i, and 0 otherwise. Obviously, $r_{ui} \in \{0, 1, ..., R\}$. Let $\hat{r}_u = \sum_{i\in I} r_{ui}/|I|$ be the average rating of u; Let $\mathcal{I}_{uv} = \mathcal{I}_u \bigcap \mathcal{I}_v$. The similarity between two users u and v is computed as

$$S(u, v) = \frac{\sum_{i\in\mathcal{I}_{uv}}(r_{ui} - \hat{r}_u)(r_{vi} - \hat{r}_v)}{\sqrt{\sum_{i\in\mathcal{I}_{uv}}(r_{ui} - \hat{r}_u)^2 \sum_{i\in\mathcal{I}_{uv}}(r_{vi} - \hat{r}_v)^2}} \tag{16}$$

Based on this similarity, we introduce the user preference function $q(M, u, i)$, which evaluates how much a user prefers an item.

DEFINITION 7. *(preference function and sensitivity) Given a user set U, an item set I, a user similarity matrix S and a rating matrix M ($M_{ui} \in \{0..R\}, R \in N^+$), for a user $u \in U$, $u's$ preference on item $i \in I$ is computed by:*

$$q(M, u, i) = \frac{\sum_{v\in U} S(u, v)M_{vi}}{R} \tag{17}$$

Generally speaking, the higher this score , the more user u prefered. For any pair of the neighbouring matrix M and $M^{'}$, the sensitivity of user preference function is defined,

$$\Delta_q = \max_{(M,M')} |q(M, u, i) - q(M^{'}, u, i)| = 1 \tag{18}$$

Algorithm 5 $\mathcal{DP}-\mathcal{A}_2^{'}(M,c,\delta_0,\mathcal{I}_u,k,\epsilon)$

1: **input**: user-to-item rating matrix M, user u's purchase record $\mathcal{I}_u$, item set I ,the privacy parameter (c,δ) where $c=1$, k is the number of recommended items and ϵ.
2: Sample each user with probability $p=\epsilon/2$ and get the new user set V.
3: **for all** $v\in V$ **do**
4: Calculate the similarity between u and v by

$$S^{'}(u,v)=\frac{\sum_{i\in\mathcal{I}_{uv}}(r_{ui}-\hat{r}_u)(r_{vi}-\hat{r}_v)}{\sqrt{\sum_{i\in\mathcal{I}_{uv}}(r_{ui}-\hat{r}_u)^2\sum_{i\in\mathcal{I}_{uv}}(r_{vi}-\hat{r}_v)^2}}$$

5: **end for**
6: **for all** $i\in\mathcal{I}_u$ **do**
7: predict the rating of u to i by

$$q^{'}(M,u,i)=\frac{\sum_{v\in V}S^{'}(u,v)M_{vi}}{R}\times\frac{1}{p}$$

8: **end for**
9: **Initialised**: $l_u=\emptyset,\mathbb{I}=I$
10: **for all** $t=1,2,....k$ **do**
11: select one item $r\in\mathbb{I}$ with probability

$$\frac{\exp(\epsilon^{'}q^{'}(M,u,r)/(2\Delta_{q^{'}}))}{\sum_{i\in\mathbb{I}}\exp(\epsilon^{'}q^{'}(M,u,i)/(2\Delta_{q^{'}}))}$$

where $\epsilon^{'}=\frac{c}{2\sqrt{2k\ln(1/\delta_0)}},\Delta_{q^{'}}=1/p$
12: $l_u=l_u\cup r,\mathbb{I}=\mathbb{I}-\{r\}$
13: **end for**
14: **output** the recommended item list l_u;

4.3 Differential Privacy User-based Recommender Algorithm

Also there are a large number of users, thus when we predict a rating we must sum the product of the similarity and rating by $|U|$ times. We sample each user with probability p recorded in V, then, we apply an exponential algorithm to get the recommended item list. The recommender algorithm based on sampling $\mathcal{DP}-\mathcal{A}_2^{'}(M,c,\delta_0,\mathcal{I}_u,\epsilon)$ is addressed as **Algorithm 5**.

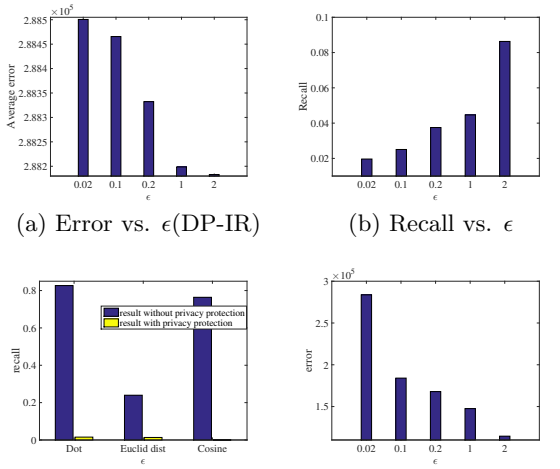
THEOREM 7. *$\mathcal{DP}-\mathcal{A}_2^{'}(M,c,\mathcal{I}_u,\epsilon)$ is $(\epsilon,\epsilon\delta_0/2)$-differential privacy.*

PROOF. : It can be derived directly from **Theorem 1** and **Lemma 1**

We can regard the variable as $Y=\{Y_1,Y_2,...,Y_{|U|}\}$,$Y_i\in[0,1]$ with probability distribution

$$Y_v=\begin{cases}\frac{S(u,v)\times M_{vi}}{R} & \text{p}\\ 0 & \text{1-p}\end{cases}$$

Suppose $\mathcal{DP}-\mathcal{A}_2^{'}$ recommends item i^* in t^{th} fold, and i_{OPT} be the optimal item in t^{th} fold. And $q(M,u,i)$ can be described as $q(M,u,i)=\sum_{u\in U}Y_u$. Then according to the **Additive Chernoff Bound**, the error deduced by sampling is $Pr[q^{'}(M,u,i)-q(M,u,i)\geq\frac{\alpha|U|}{p}]\leq\exp(-2\alpha^2|U|)$ Let $\exp(-2\alpha^2|U|)\leq\frac{1}{|U|^2}$ we get $error_1=\frac{\sqrt{|U|\ln|U|}}{p}=\frac{2\sqrt{|U|\ln|U|}}{\epsilon}$ And the error deduced by exponential privacy is



(a) Error vs. ϵ(DP-IR) (b) Recall vs. ϵ

(c) Dot Sim vs. Euclidean and Cosine (d) Error vs. ϵ(DP-UR)

Figure 1: Experiment result

$error_2=\frac{8\sqrt{2k\ln(1/\delta_0)}}{\epsilon}$ The total error will be

$$error=\frac{4\sqrt{|U|\ln|U|}+8\sqrt{2k\ln(1/\delta_0)}}{\epsilon}$$

Similarly we have :$\Gamma=S(i,i_{opt})-S(i,i^*)$

$$Pr[\Gamma>\frac{8\sqrt{2k\ln(1/\delta_0)}(\log|I|+t)+4\sqrt{|U|\ln|U|}}{\epsilon}]\leq\frac{1}{|U|^2}+e^{-t} \quad (19)$$

THEOREM 8 (ACCURACY OF DP-UR). *DP-UR is $(\mathcal{O}(\frac{\sqrt{k\ln(1/\delta_0)}(\log|I|+t)+\sqrt{|U|\ln|U|}}{\epsilon}),\frac{1}{|U|^2}+e^{-t})$-accuracy*

5. EXPERIMENT

5.1 The Data Set

The data set is from 'Alibaba Big Data Competition', which has 182881 records with 9531 items and 884 users. We divide the data set into training set and test set. The test set contains the purchase records of dozens of the users randomly sampled from the data set. Each record in the dataset is in the form of $(uid,iid,operation,time)$. The operations include *browse*, *collect* and *buy*, which are mapped to the rating score of a user as 1, 2 and 3. The another data set comes from the Netflix data set. $training-set.tar$ is a tar of a directory containing 17770 files, one per movie.

5.2 Experiment for DP-IR

In this section, we verify the effectiveness of DP-IR from three aspects. One is to quantitatively examine the recommended results against the metric of quality score and verify the relationship between the error and ϵ . And in order to show the quality of recommender result directly, we also calculate recall and show the relationship between recall and ϵ. We make comparison between Dot similarity and another representative similarity measurement since the adoption of similarity metric has great influence on sensitivity.

Recommendation Error vs. ϵ. This experiment tests the relationship between the recommendation error and the

privacy parameter ϵ, since the ϵ is determined by the sampling probability p, thus also, it's the relationship between error and the number of users . The parameter ϵ is selected from $\{2, 1, 0.2, 0.1, 0.02\}$ respectively , accordingly p is selected from $\{1, 0.5, 0.1, 0.05, 0.01\}$ respectively. The parameter setting in this experiment is $m = 50$. Given a ϵ, for each user u in the test set TU, each item $i \in \mathcal{I}_u$, the related list generated by $\mathcal{DP} - \mathcal{A}_1^{'}$ is denoted by L_i. The quality of one related list L_i is calculated by $\sum_{j \in L_i} S_{ij}$, and $score_1$ is the mean of these quality ,$score_1 = \frac{\sum_{u \in TU} \sum_{i \in \mathcal{I}_u} \sum_{j \in L_i} S(i,j)}{\sum_{u \in TU} |\mathcal{I}_u|}$. Also we get the optimal related list l_i^{opt} for each $i \in \mathcal{I}_u$ by Algorithm 1 $\mathcal{A}_{nonpri}(M, \mathcal{I}_u)$.

And $score_1 = \frac{\sum_{u \in TU} \sum_{i \in \mathcal{I}_u} \sum_{j \in L_{opt}} S(i,j)}{\sum_{u \in TU} |\mathcal{I}_u|}$ Then for this ϵ, the error is computed as $error = score_1 - score_2$.

The result based on Netflix dataset, as depicted in **Figure** 1(a), shows the average error decreases with the increasing size of the training set. This illustrates that a larger population bring more accuracy on recommendation.

Recall vs. ϵ. In order to understand the recommended result directly, the *recall* score is adopted here as an evaluation. Let S_1 be the item set purchased by u actually , S_2 be the item set that we recommend to u, and $S_3 = S_1 \cap S_2$, $recall = \frac{S_3}{S_1}$. Obviously the smaller the error the higher the recall. As shown in **Figure** 1(b) , the recall increases with the increasing of ϵ.

Dot Similarity vs. Euclidean Distance-based Similarity and Cosine Similarity. To verify our similarity metric, some comparisons are made against the Euclidean Distance-based Similarity and Cosine Similarity based on the 'alibaba' data set. The recall score is adopted here as an evaluation. The parameters are set $\epsilon = 2$, accordingly $p = 1$, respectively. The results, depicted in **Figure** 1(c) , show that the recall under our method is better than with Euclidean Similarity and Cosine Similarity and **Algorithm** $\mathcal{DP} - \mathcal{A}_1^{'}$, though the result based on Cosine Similarity in **Algorithm** $\mathcal{A}_{nonpri}$ is the best, however it's recall is 0 in **Algorithm** $\mathcal{DP} - \mathcal{A}_1^{'}$.

5.3 Experiments for DP-UR

In this section, we verify the effectiveness of DP-UR. We show the relationship between the average quality score of the recommended item list to user u and ϵ .

This experiments verify the relationship between the competitive parameters accuracy and privacy. For each user u in the test set we recommend 200 items in list L^* by DP-UR, and calculate the quality score of the list we recommend to u similarly to that in the last section The quality error is calculated as $error = score_1 - score_2$. From the results in **Figure** 1(d), we can see that the average error scales inversely with the increasing ϵ. This illustrates that the quality of the recommender result can be improved with the increasing number of user.

6. CONCLUSION

We solve the inference attack problem in recommender system by applying the differential privacy method into the procedure of recommendation. We design two differentially private recommender algorithms, named DP-IR and DP-UR for item based recommender system and user based based recommender system, respectively. Algorithms are based on the exponential mechanism with a carefully designed quality function with sampling. And we discuss what kind of similarity measurement are appropriate in differential privacy. Theoretical analyses on privacy of these algorithms are presented. We also investigated the accuracy about the proposed method and give theoretical results. Experiments are performed on real dataset to verify our methods, which show that the recommended results get improved with the increasing of the number of users.

7. REFERENCES

[1] Calandrino J A, Kilzer A, Narayanan A, et al. " You Might Also Like:" Privacy Risks of Collaborative Filtering[C]//Security and Privacy (SP), 2011 IEEE Symposium on. IEEE, 2011: 231-246.

[2] Dwork C. Differential privacy[M]//Encyclopedia of Cryptography and Security. Springer US, 2011: 338-340.

[3] McSherry F, Mironov I. Differentially private recommender systems: building privacy into the net[C]//Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2009: 627-636.

[4] Dwork C, Rothblum G N, Vadhan S. Boosting and differential privacy[C]//Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on. IEEE, 2010: 51-60.

[5] McSherry F, Talwar K. Mechanism design via differential privacy[C]//Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium on. IEEE, 2007: 94-103.

[6] Deshpande M, Karypis G. Item-based top-n recommendation algorithms[J]. ACM Transactions on Information Systems (TOIS), 2004, 22(1): 143-177.

[7] Rokach L, Shapira B, Kantor P B. Recommender systems handbook[M]. New York: Springer, 2011.

[8] Hardt M, Roth A. Beating randomized response on incoherent matrices[C]//Proceedings of the forty-fourth annual ACM symposium on Theory of computing. ACM, 2012: 1255-1268.

[9] Adam Smith, Differential privacy and the secrecy of the sample, 2009, https://adamdsmith.wordpress.com/2009/09/02/sample-secrecy/

[10] Hoeffding W. Probability inequalities for sums of bounded random variables[J]. Journal of the American statistical association, 1963, 58(301): 13-30.

[11] Oh S, Viswanath P. The composition theorem for differential privacy[J]. arXiv preprint arXiv:1311.0776, 2013.

[12] Li C, Hay M, Rastogi V, et al. Optimizing linear counting queries under differential privacy[C]//Proceedings of the twenty-ninth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems. ACM, 2010: 123-134.

[13] Dwork C, Roth A. The algorithmic foundations of differential privacy[J]. Theoretical Computer Science, 2013, 9(3-4): 211-407.

[14] Zhu, Tianqing, et al. Differential privacy for neighborhood-based collaborative filtering. Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. ACM, 2013.