PrivacyCanary: Privacy-Aware Recommenders with Adaptive Input Obfuscation

Thivya Kandappu †*, Arik Friedman *, Roksana Boreli *, Vijay Sivaraman †
† School of Electrical Engineering & Telecommunications, UNSW, Sydney, Australia.
∗ National ICT Australia, Sydney, Australia.
Emails: {*t.kandappu@student.unsw.edu.au, arik.friedman@nicta.com.au, roksana.boreli@nicta.com.au, vijay@unsw.edu.au*}

***Abstract*—Recommender systems are widely used by online retailers to promote products and content that are most likely to be of interest to a specific customer. In such systems, users often implicitly or explicitly rate products they have consumed, and some form of collaborative filtering is used to find other users with similar tastes to whom the products can be recommended. While users can benefit from more targeted and relevant recommendations, they are also exposed to greater risks of privacy loss, which can lead to undesirable financial and social consequences. The use of obfuscation techniques to preserve the privacy of user ratings is well studied in the literature. However, works on obfuscation typically assume that all users uniformly apply the same level of obfuscation. In a heterogeneous environment, in which users adopt different levels of obfuscation based on their comfort level, the different levels of obfuscation may impact the users in the system in a different way. In this work we consider such a situation and make the following contributions: (a) using an offline dataset, we evaluate the privacy-utility trade-off in a system where a varying portion of users adopt the privacy preserving technique. Our study highlights the effects that each user's choices have, not only on their own experience but also on the utility that other users will gain from the system; and (b) we propose PrivacyCanary, an interactive system that enables users to directly control the privacy-utility trade-off of the recommender system to achieve a desired accuracy while maximizing privacy protection, by probing the system via a private (i.e., undisclosed to the system) set of items. We evaluate the performance of our system with an off-line recommendations dataset, and show its effectiveness in balancing a target recommender accuracy with user privacy, compared to approaches that focus on a fixed privacy level.**

I. Introduction

Online retailers and content providers deliver a huge variety of offerings, and matching consumers with the most relevant products is pivotal to both user satisfaction and the service providers' revenues. Consequently, these service providers have made recommender systems a salient part of their Web sites, as they analyze patterns of user interest in products to offer personalized recommendations to individual users. The use of recommender systems has in fact become widespread in online services, as exemplified by large retail sites like Amazon[1] or eBay[2] and content providers like Netflix[3].

Despite the benefits of personalized recommendations, there is also a negative side to the use of such systems. Recommendations are based on personal data (e.g., purchase or viewing history and ratings of items) and can therefore lead to privacy loss. For example, although recommendations in online services are derived from aggregated users' information, they can be used to infer information about specific users, aided by only a limited amount of background information [1], [2]. Such works stress the importance of incorporating privacy-enhancing mechanisms into recommender systems.

Proposed solutions to this problem rely on cryptographic techniques [3], privacy enhancement via a centralized trusted entity [4], or local (client side) data obfuscation. In this paper we focus on local obfuscation of users' ratings to protect privacy against either the service provider or third party inference of users' private information, where the obfuscation can be performed by a user agent (e.g., a browser plugin or other client-side software). Client side data obfuscation has been studied extensively in the context of data mining [5], [6] and it has the advantage of providing privacy protection without requiring the users to trust the online system, but in turn it reduces the accuracy of the recommendations. Unlike the majority of previous work, which targets a specific level of privacy and aims to maximize the utility (i.e., accuracy of the recommender system) within this bound, we consider a user-acceptable utility and propose a mechanism that will achieve this.

To measure and control the level of privacy obtained through obfuscation, we rely on differential privacy [7], a rigorous framework that enables quantification of privacy loss under arbitrary adversary conditions. We study the resulting trade-off between privacy and utility in the context of recommender systems, where users obfuscate their ratings by adding a selected (according to the desired privacy setting) level of obfuscation noise. We provide the following contributions:

- Using an off-line movie ratings dataset, we evaluate the trade-off between privacy and accuracy of recommendations when a varying portion of users in the system obfuscate the reported ratings. We evaluate the cumulative effect of the obfuscation conducted by different users, and show how the rating **prediction accuracy for each user** is affected both by **this user's privacy choices** and by **other users' choices**. We show that the **impact of noise introduced directly by a user**, on their rating prediction accuracy, is **significantly greater than** the resulting **impact of noise added by other system users**. For example, a target prediction accuracy of 0.946 for

[1] http://www.amazon.com/
[2] http://www.ebay.com.au/
[3] https://www.netflix.com/global

a specific user could be maintained either when the user added a *low* level of noise directly to his ratings (while other system users contributed non-noisy ratings), or when 20% of the users in the system (188 in our experiments) obfuscated their data with the same noise level (while the target user contributed non-noisy ratings).
- We propose PrivacyCanary, an **adaptive obfuscation mechanism** that allows users to toggle the obfuscation level based on the prediction accuracy estimated using a reference set of *canary items*[4] and targeting a **desired accuracy of the predictions**. The canary set consists of a small number of user ratings that are not disclosed to the recommender system. We evaluate our algorithm using the same dataset of movie ratings, and demonstrate that this algorithm can **effectively maintain the desired accuracy level**. We further show that, compared to random selection of *canary items*, choosing movies from the extreme (most or least liked) user's items leads to a lower deviation from the target accuracy, with an error lowered by a factor of close to 2.

Whereas prior studies of obfuscation techniques assume a uniform policy employed by all users, we believe our work is the first to study how the utility provided by the system changes as different fractions of the user community adopt privacy-enhancing technologies. In addition, the proposed interactive obfuscation mechanism allows the user agent to change the level of obfuscation to obtain the desired level of accuracy while taking into account the dynamically changing influence of inputs from other system users. While our proposal provides a best-effort level of privacy (as each user item can have a specific level of obfuscation and a corresponding privacy guarantee), we believe that a target accuracy level is a more meaningful metric for the user. We argue that the primary purpose of the recommender system should not be compromised as the system would otherwise have no meaningful purpose. Regardless, we note that this approach can be easily modified to track the cumulative loss of privacy, and take action if losing privacy beyond some bound warrants quitting use of the system.

The remainder of the paper is organized as follows. Existing privacy preserving solutions for recommender systems are reviewed in Section II. We present our system setup and background in Section III, and evaluate the privacy-utility trade-off using the MovieLens dataset in Section IV. Section V presents our adaptive obfuscation mechanism, and demonstrates its effectiveness in achieving the desired level of accuracy. We discuss our results in Section VI.

II. Related Work

Privacy-aware recommender systems were studied in a number of prior works, including approaches such as homomorphic encryption of user data [3], [8]; decentralizing storage and operating the recommender system in a distributed manner [9], [10]; and data modification techniques (most closely related to this paper) to limit the profiling accuracy of the recommender system.

Data modification techniques include approaches such as data obfuscation [11], [12], data randomization [13], and differentially private recommenders [4], [14]. They allow users to modify their ratings by adding random noise to prevent the recommender system from deriving the users' actual ratings. Polat and Du [13] aimed to find a perturbation algorithm that imposes the smallest error on recommendations while ensuring that users will get a high level of privacy. Berkovsky et al. [11] evaluated experimentally the impact of data obfuscation on the accuracy of the generated predictions. While we adopt the same data obfuscation technique to perturb the users' answers, we take a different perspective that captures the performance of the system when different fractions of the user community apply the privacy mechanism.

Differential privacy [7], [15] provides provable guarantees of privacy under arbitrary adversary conditions. Works on differentially private recommender systems [4], [14] focused on event-level privacy, where privacy is guaranteed for each rating (rather than with respect to each user), and studied how a central entity could generate differentially private recommendations from the collected data. In this work we rely on event-level differential privacy obtained through input perturbation, where the noise is added by the users before submitting the responses to the system. We perform experimental evaluation of the impact that user response obfuscation has on the recommender accuracy, and we study how the recommender accuracy changes depending on the number of privacy-conscious users that obfuscate their inputs.

Weinsberg et al. [2] studied obfuscating user ratings, with the goal of obtaining accurate recommendations while preventing inferences on user demographics. E.g., the privacy gain obtained by obfuscation is captured via the reduced performance in inference of gender based on the user ratings.

Chen et al. [16] studied how the adoption of privacy policies by different proportions of users affects inference attacks in online social networks. In contrast, we consider their effect on prediction accuracy in the context of recommendation systems.

III. System Setup

We consider a generic system architecture, where the users access an online service that also includes recommender system functionality (within the system, or provided externally, by an analytics service), as shown in Figure 1.

A. Entities and System Components

Users utilize the *online (shopping or content provider) service* and also use the associated *recommendation system*. For the latter, they provide ratings for the items purchased or viewed using a client-side software, e.g., a browser add-on or a mobile app. The software obfuscates the users' ratings before they are sent to the recommender system by adding noise, where the magnitude of noise is calibrated according to the level of privacy chosen by the user. We note that the privacy

[4]Canary birds were used in coal mining as an early warning system.

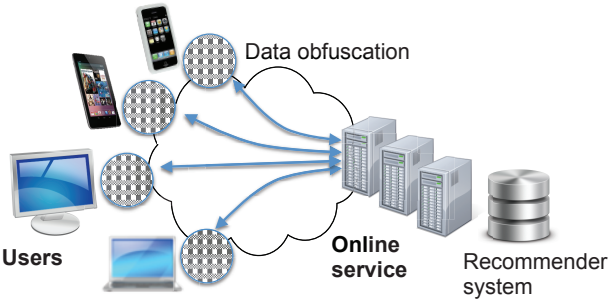


Fig. 1. System architecture

level needs to be determined by considering two contradicting objectives: users' privacy preferences, and system utility (receiving meaningful recommendations on relevant products).

The recommender system acquires rating and transaction data from users and generates personalized recommendations. The objective of the recommender system is to accurately predict user preferences for future viewings or purchases.

For our study, we utilize matrix factorization [17], the state-of-the-art technique in recommender systems. The input to the recommendation process is a set R of ratings of n users over m items, such that each element $r_{ui} \in R$ is a rating that user u assigned to item i. The matrix factorization process derives two low-rank matrices with d latent factors: $P_{n\times d}$ for users and $Q_{m\times d}$ for items, where each row p_u in P pertains to a user, and each row q_i in Q pertains to an item. The matrices are constructed such that they provide an accurate approximation for the known ratings, i.e., for all the known ratings, the predicted rating of user u for item i, $\hat{r}_{ui}$, maintains $\hat{r}_{ui} \approx r_{ui}$. To this end, the following loss function is minimized:

$$J_S(P,Q) := \sum_{r_{ui}\in R} [(r_{ui} - p_u q_i^\intercal)^2 + \gamma(\|p_u\|^2 + \|q_i\|^2)] \quad , \quad (1)$$

where γ is a regularization parameter that prevents overfitting. A predicted rating of user u for item i is then given by $\hat{r}_{ui} = p_u \cdot q_i^\intercal$. In this work we use matrix factorization as a black box, and a similar evaluation could be conducted with any other collaborative filtering algorithm.

B. Data Obfuscation Using Differential Privacy

To measure the privacy protection resulting from data obfuscation (perturbation of the recommender system input), we rely on the differential privacy framework [7]. Differential privacy sets strict bounds on the sensitivity of the outcome of the computation to any particular record in the input. We say that two datasets A and B are neighbors if they differ only in a single record. I.e., $\exists r_{ui}, r'_{ui}$ such that $A = B \setminus \{r_{ui}\} \cup \{r'_{ui}\}$. A mechanism M provides (ϵ,δ)-differential privacy if for any neighboring datasets A and B, and for all outcomes $S \subseteq Range(M)$:

$$\Pr[M(A) \in S] \leq exp(\epsilon) \times \Pr[M(B) \in S] + \delta \quad . \quad (2)$$

Differential privacy guarantees that an adversary will not be able to infer any particular rating (i.e., the difference between the datasets A and B) from the outcome of the computation regardless of the computational power or the background knowledge available to the adversary. The parameter ϵ controls the level of privacy, where smaller values of ϵ provide stricter bounds on the influence of any particular input record on the outcome, and therefore provide better privacy. The parameter δ allows relaxing the condition in Equation (2) from the stricter definition of ϵ-differential privacy [7], in which $\delta = 0$.

The privacy options considered by users should be simple enough for the lay user to understand, so we adopt a simple set of four privacy levels: *no*, *low*, *medium* and *high*. In the experimental evaluation, we use two different noise distributions, namely Laplace and Gaussian, to generate differentially private noise to obfuscate the users' responses.

The Laplace mechanism [7] can be used to perturb the user response while conforming to ϵ-differential privacy. A Laplace distribution with a scale parameter b (and variance $\sigma^2 = 2b^2$) has probability density function $\Pr(x|b) = \frac{1}{2b}\exp(-|x|/b)$. Given the range Δ of the user's possible answers, the Laplace mechanism obtains ϵ-differential privacy by adding to each input noise sampled from the Laplace distribution, with its scale b calibrated through the relation:

$$\epsilon = \frac{\Delta}{b} \quad . \quad (3)$$

Similarly, (ϵ,δ)-differential privacy can be obtained by perturbing the inputs with Gaussian noise $\mathcal{N}(0, \sigma^2)$ [18], calibrated through the relation:

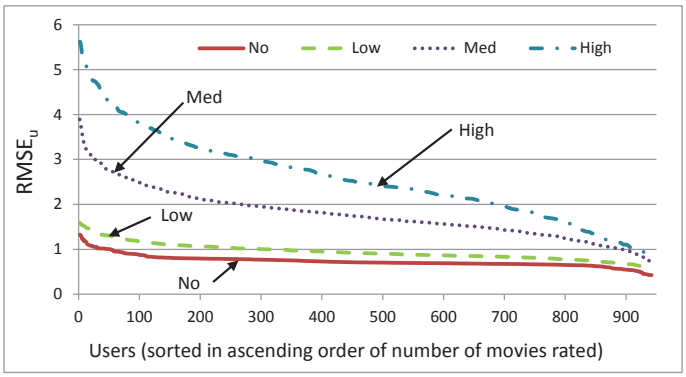
$$\frac{\epsilon\sigma^2}{2\Delta^2} + \ln(\epsilon\sigma^2) \geq \ln\frac{1}{\delta} \quad . \quad (4)$$

Example 1: Consider a 5-point rating scale commonly used in movie ratings, with ratings between 1 (very poor) and 5 (excellent). We use noise with standard deviation $\sigma = 0$ for no privacy, $\sigma = \sqrt{2}$ for low privacy, $\sigma = 4\sqrt{2}$ for medium privacy, and $\sigma = 8\sqrt{2}$ for high privacy (note that the obfuscated responses will consequently be real-valued rather than integers) for both Laplace and Gaussian distributions. With $\Delta = 5 - 1 = 4$ and $\delta = 0$, the *low*, *medium* and *high* privacy settings correspond to $\epsilon = 4$, $\epsilon = 1$ and $\epsilon = 0.5$ respectively for Laplace noise. Setting $\delta = 0.01$, the *low*, *medium* and *high* privacy settings correspond to $\epsilon = 17.1$, $\epsilon = 1.07$ and $\epsilon = 0.26$ respectively for Gaussian noise.

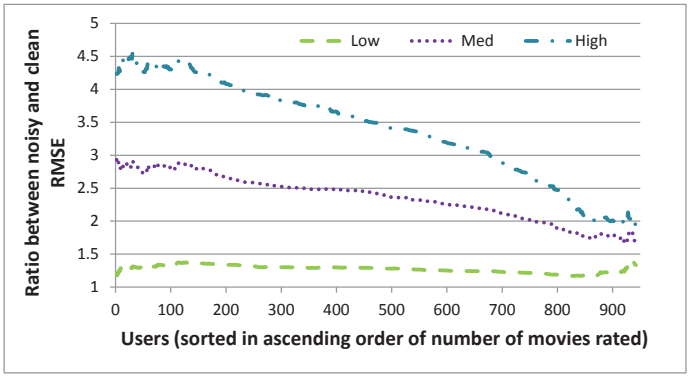
IV. Experimental Evaluation of Client-Side Obfuscation

To evaluate the impact of client-side obfuscation on the accuracy of the resulting recommendations, we conducted an experimental evaluation using the MyMediaLite [19] recommender system library (version 3.04) and the MovieLens $100K$ dataset [20]. The dataset contains $100K$ ratings from 943 users, who rated 1682 movies. Each user rated 106.04 movies on average.

Table I summarizes the parameter values that were used in the matrix factorization algorithm throughout the experiments.



(a) Impact of Laplace noise on the accuracy of the recommendations



(b) Ratio between noisy and clean RMSE across users – Laplace noise

Fig. 2. The impact of Laplace noise on rating prediction accuracy

We conducted a separate evaluation to tune the parameters so that the prediction error of a non-private recommender over the MovieLens $100K$ dataset is minimized.

TABLE I
THE PARAMETERS USED FOR MATRIX FACTORIZATION

Parameter	Value
Number of factors	10
Regularization Parameter	0.1
Learning rate	0.1
Number of iterations	40

Section IV-A outlines the evaluation criteria in our experiments. In Section IV-B we consider the direct impact of the user's privacy choices and the level of obfuscation on the resulting prediction accuracy of this user's ratings. Section IV-A extends this evaluation to assess how this accuracy is influenced by the obfuscation conducted by other users in the system. Finally, we present in Section IV-D a game-theoretic interpretation of our results and its implication on user choices.

A. Evaluation Criteria

Enhancing privacy through obfuscation of user ratings imposes some error on the accuracy of the recommendation system. We evaluate the cost (in accuracy) of obfuscation by comparing the prediction error in our system to that in a system without privacy protection. In both cases we compute the prediction error based on the root mean square error (RMSE). Let $r_{ui} \in \{1,2,3,4,5\}$ be the rating that user u assigned to item i. We denote by $\hat{r}_{ui}$ the system's prediction for that rating. Let $R_u = \{r_{vi} \in R | v = u\}$ be the ratings of user u. To evaluate the RMSE for a particular user u in rating prediction, we use "leave-one-out" cross validation. The data set is split into training and test data, where a data point r_{ui} is taken as the test data, and the remaining ratings $R \setminus \{r_{ui}\}$ are taken as the training data used to predict the rating $\hat{r}_{ui}$. The prediction error $r_{ui} - \hat{r}_{ui}$ is then calculated for that particular rating. The process is repeated over all the ratings $r_{ui} \in R_u$ and the squared prediction error is averaged to yield the RMSE of that particular user u, given by:

$$RMSE_u = \sqrt{\frac{\sum_{r_{ui} \in R_u} (r_{ui} - \hat{r}_{ui})^2}{|R_u|}} \quad . \tag{5}$$

The overall RMSE of the recommender system $RMSE_S$ is then evaluated by:

$$RMSE_S = \sum_{u \in U} RMSE_u \Big/ |U| \quad , \tag{6}$$

where U is the set of users in the dataset.

We note that this measure is slightly different from the typical RMSE measurement used to evaluate overall system accuracy, in which the mean value is evaluated over the ratings, rather than over the user averages:

$$RMSE^* = \sqrt{\frac{\sum_{u \in U, r_{ui} \in R_u} (r_{ui} - \hat{r}_{ui})^2}{|R|}} \quad . \tag{7}$$

We believe the measurement in Equation 6 is more suitable in this context, as we investigate the accuracy for each of the users, rather than the per-rating prediction accuracy. However, we also report for reference the RMSE measurement per Equation 7.

We evaluate $RMSE_u$ and $RMSE_S$ both for the system with no input perturbation (*no* privacy), and for the system with different privacy levels (*low*, *medium* and *high*).

B. Obfuscation by a Single User

In this section, we evaluate the accuracy of the predictions for the target user when that user obfuscates their ratings. The noise settings (i.e., Laplace and Gaussian Noise) for which we conducted our experiments are described next.

To provide privacy, Laplace noise with mean 0 and scale b (calibrated using Eq. (3)) is added locally to the user's ratings to perturb them before they are sent to the recommender system. In this experiment only the ratings of the target user are perturbed, while the ratings of other users are used without any noise added. The root mean square error for the target user u ($RMSE_u$) is calculated by averaging the prediction error

TABLE II
PRIVACY AND ACCURACY TRADE-OFFS

	Parameters	Low	Medium	High
Laplace noise	Std.dev of the noise	$\sqrt{2}$	$4\sqrt{2}$	$8\sqrt{2}$
	Privacy budget ϵ	4	1	0.5
	$RMSE_S$	0.9460	1.7791	2.5841
	Std. dev $of RMSE_S$	0.0885	0.0963	0.1032
	$RMSE^*$	0.8442	0.9659	1.5041
Gaussian noise	Std.dev of the noise	$\sqrt{2}$	$4\sqrt{2}$	$8\sqrt{2}$
	Privacy budget ϵ	17.1	1.07	0.26
	$RMSE_S$	0.9216	1.7104	2.2255
	Std. dev $of RMSE_S$	0.0791	0.0845	0.0978
	$RMSE^*$	0.8095	0.9034	1.2568

over all of target user u's ratings. This evaluation is conducted over all 943 users of the MovieLens $100K$ dataset to estimate $RMSE_S$. The same experiment is repeated also using Gaussian noise (where the ϵ and δ parameters are calibrated using Equation (4)), using the same standard deviation used for Laplace noise.

Fig. 2(a) shows the average RMSE in prediction across all the users in the dataset when the data is obfuscated with Laplace noise. The X-axis depicts the users sorted in ascending order of the number of movies rated by each user. Each point in this figure represents $RMSE_u$ of a particular user, averaged over 100 simulation runs. Different lines in the plot represent the different noise settings chosen by the user (see Table II). This plot demonstrates the impact of the user's obfuscation on the prediction accuracy. The error in prediction is lower for users who rated more movies (i.e., users on the right). Comparing to the prediction error without the presence of noise, addition of noise will incur some cost in prediction accuracy. Unsurprisingly, the figure illustrates that as users reveal more about their preferences to the system, the system can predict the user ratings more accurately, and this holds also for the noisy ratings. Moreover, Figure 2(b) shows the ratio between the $RMSE_u$ obtained by the noisy inputs and that obtained by the non-private variant, and demonstrates that the relative cost in accuracy also diminishes as users rate more items.

Similar results were observed when the user responses are obfuscated with Gaussian noise and are omitted for brevity.

Table II summarizes the overall privacy-accuracy trade-off across different privacy settings (*no*, *low*, *medium* and *high*), when the data is obfuscated with Laplace or Gaussian noise. The RMSE values for each privacy level are obtained by averaging the $RMSE_u$ of all the users (Equation 6). We note that the average RMSE value for the baseline case, when no noise is added by users, is 0.84. We can observe that the overall error in prediction (i.e., $RMSE_S$) increases as users add more noise. Consider, for example, a user who wants to get good (i.e., accurate) recommendations while enhancing privacy, so that the prediction accuracy stays within an RMSE of 1.0. In other words, the user is willing to sacrifice an increase of almost 20% in the RMSE for better privacy. In that case, the average user should settle for the *low* privacy option (equivalent to $\epsilon = 4$), as any higher privacy level would result in a higher loss of accuracy than would be acceptable by the user. In general, by changing the noise level, the user can trade-off the accuracy of the recommendations with the desired level of privacy.

C. Impact of the Noise Added by Multiple Users

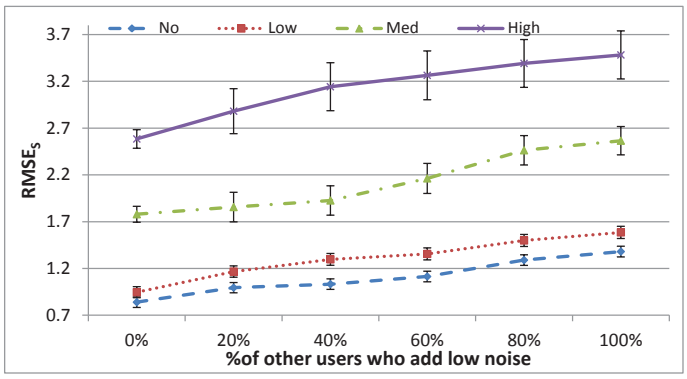
If many users in the system are privacy-conscious, and obfuscate their responses to protect their privacy, we can expect that the added noise would have a significant impact on the overall accuracy of the system, affecting also the recommendations for users who did not perturb their inputs. In this section we investigate how the prediction accuracy for a specific user is influenced by input obfuscation done by other users. We consider a target user u, who can choose a privacy level $p \in \{no, low, medium, high\}$ and explore the following scenarios:

1) User u, along with a fraction of other users in the system, obfuscate their responses using the same privacy level p. The remaining users do not obfuscate their ratings.
2) User u is not concerned about privacy (i.e., the user selected the *no privacy* option), but a fraction of other users in the system obfuscate their responses, using the privacy level p. The remaining users do not obfuscate their ratings.
3) User u selected the privacy level p to obfuscate responses, and a fraction of other users in the system obfuscate their responses with a different privacy level q. The remaining users do not obfuscate their ratings.

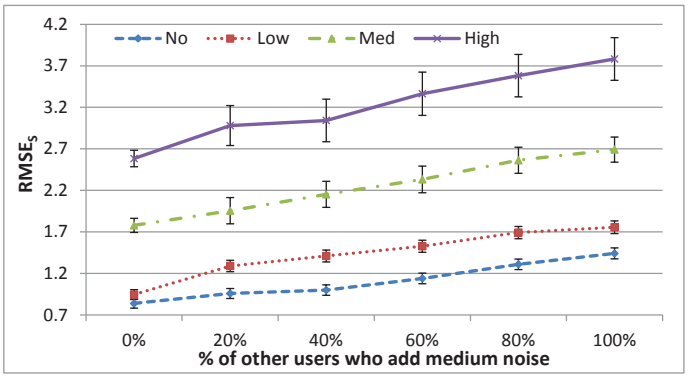
For each of those scenarios, we test the impact of the obfuscation on the prediction accuracy for the particular user u. The motivation behind the different scenarios is to study which obfuscation has more significant influence on the prediction accuracy for user u: the obfuscation done by the user, or by other users in the system.

Figures 3(a) shows the average RMSE of the generated predictions of a target user u ($RMSE_u$), while a fraction of the other users in the system obfuscate their responses with *low* privacy noise level from Laplace noise distribution. Each line in Figure 3(a) represents the privacy level chosen by the target user u. The RMSE is averaged over all the users in the dataset for whom $RMSE_u$ is evaluated (i.e., 943 experiments are conducted, each with a different target user u). For each user u, $RMSE_u$ is evaluated based on 100 simulation runs. Figures 3(b) and 3(c) reflect a similar setup, but with the fraction of other users choosing *medium* and *high* privacy noise level from Laplace noise distribution respectively. Without any obfuscation, the RMSE in prediction is around 0.84. Alongside the average RMSE, we also show error bars that depict the standard deviation of the prediction error. We make several observations from these plots.

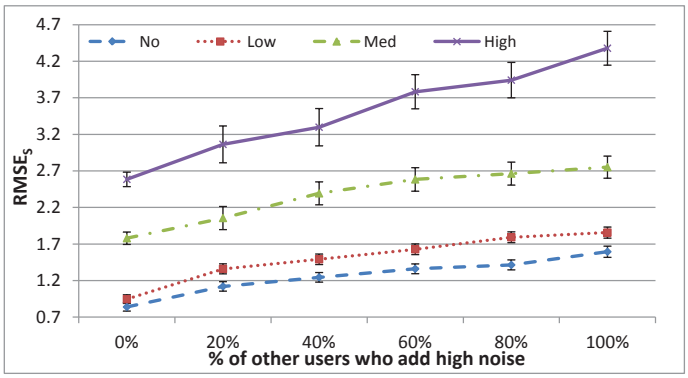
First, we observe that even when a target user does not obfuscate his input, the prediction accuracy obtained from the system decreases as more and more other users obfuscate their responses (see the no-privacy lines in the figures). As the target user starts to obfuscate his responses, the error in prediction significantly increases with the increasing fraction of other



(a) Other users add low level of Laplace noise



(b) Other users add medium level of Laplace noise



(c) Other users add high level of Laplace noise

Fig. 3. Impact of (a) low (b) medium and (b) high Laplace noise on the accuracy of the recommendations.

users who add noise. Our results indicate that the accuracy of the predictions of a target user can be increased either by lowering the privacy level (of the target user and/or of the other users in the system) or by adjusting the fraction of other users who obfuscate their data.

Second, the results show that the penalty in RMSE when both the target user and a fraction of other users in the system obfuscate the data is more than the combination of penalties imposed by the obfuscation done by the target user alone, or the other users in the system. For example, when the target user adds *low* privacy obfuscation using Laplace noise while others do not, the RMSE increases by 0.1 $(= 0.94 - 0.84)$ with respect to the non-private baseline; when 20% of the other users add *low* privacy obfuscation while the target user does not, the RMSE increases by 0.15 $(= 0.99 - 0.84)$ with respect to the non-private baseline. However, when both the target user and 20% of the users add *low* privacy obfuscation, the RMSE increases by 0.32 $(> 0.1 + 0.15)$ with respect to the baseline.

Lastly, our results indicate that the noise addition by the user incurs a higher cost in prediction accuracy than the noise added by the other users. For example, in Figure 3(a), the *low* level obfuscation done by the target user alone (while no other user adds noise) results in an average error of 0.95, which is equivalent to the RMSE resulting from the combined effect of 20% of the other users obfuscating their data (while the target user does not add noise).

D. Prisoner's Dilemma

While the experiments conducted in this section explore only a few simple settings, they offer several insights into the impact of user decisions on the system as a whole. A user's choice affects not only the accuracy of the recommendations for that user, but also the utility of the system for other users. While this effect may be negligible when only a single user obfuscates the input, the cumulative effect of many users obfuscating the input considerably brings down the accuracy of the recommendations. In fact, in some cases the options available to the user may be perceived as a (well known) game theory paradox, the prisoner's dilemma [21]. For example, consider a simple case where the users make a choice between having either a *low* level of privacy, or *no* privacy. We further assume that, to gain the privacy protection offered by obfuscation, a typical user would be willing to sacrifice a loss of accuracy of up to 0.3 in the RMSE measurement. We consider the overall utility for the user to be a combination of the recommendations accuracy (represented by $RMSE_u$) and the privacy protection, where the added value of a *low* privacy setting over the *no* privacy setting amounts to a utility gain of 0.3. In other words, the utility score is equal to $RMSE_u$ when no privacy protection is in place, and $RMSE_u - 0.3$ when a *low* level of privacy protection is in place. A lower

value indicates higher utility for the user.

Table III captures the resulting situation for a typical target user, based on the experimental outcomes reported in Figure 3(a) (note that the utility gain of 0.3 due to privacy protection results in correspondingly lower utility scores for cases where the target user chooses low privacy). We observe that, regardless of other users' choices, the target user's dominant strategy resulting in the best outcome, i.e., best utility, is to always obfuscate the input. However, if all users make this choice, the resulting outcome, for all users, will be a significantly degraded utility from the system, comparing to a system where no obfuscation is conducted.

TABLE III
UTILITY VALUES SHOWING THE PRIVACY/ACCURACY TRADE-OFF AS A PRISONER'S DILEMMA. LOWER VALUE INDICATES HIGHER UTILITY FOR THE USER

Example user chooses:	No privacy	Low Privacy
All other users choose:		
No Privacy	0.84	0.64 (=0.94-0.3)
Low Privacy	1.38	1.28 (=1.58-0.3)

The game-theoretic point of view suggests that with growing user awareness of the privacy implications of personalization, and considering the deficiencies of current recommender systems in preserving privacy, client-side privacy-enhancing solutions could become more prevalent, and could eventually harm the performance of personalized systems, to the detriment of both the users and the service providers. In the next section we will show an adaptive obfuscation mechanism that allows to mitigate this problem, and help users achieve a desired level of accuracy while maximizing obfuscation.

V. ADAPTIVE OBFUSCATION

In Section IV-B we showed how the rating prediction accuracy for a user is influenced by the cumulative effect of the privacy choices made by other users in the system. Since getting adequate prediction accuracy is a key requirement of the system, we propose a practical approach to balance privacy and accuracy in this context. The goal is to allow users to receive reasonably accurate recommendations, but without completely compromising their privacy. To this end, we propose PrivacyCanary, an interactive recommendation service, which can be probed by the users to assess the expected accuracy of the predicted ratings. This feedback mechanism allows the users to adapt obfuscation according to the current performance of the system.

In our system, users obtain predictions from the recommender system and have a pre-determined range of acceptable prediction accuracy. To control the accuracy of received recommendations and the level of privacy loss, each user agent utilizes a reference set of *canary items*. The canary set is a set of items that the user rated without disclosing the ratings to the recommender system. By probing the recommender engine to obtain the predictions for these canary items, the user agent can estimate the current recommendation accuracy and select the appropriate level of obfuscation to maintain accuracy within the acceptable range. Essentially, the user agent adapts the obfuscation level of subsequently submitted ratings according to the expected accuracy, so privacy protection can be maximized while maintaining reasonable accuracy in future predictions.

We now briefly describe the protocol workflow, which is repeated each time that a user rates a new item:

Step 1: the user probes the system: the user u keeps a set I_u of item ratings as canary set and probes the system for rating predictions for the items in the set.

Step 2: the recommender system sends the predictions for the canary items: the recommender system executes the recommendation algorithm to derive predictions for the canary set.

Step 3: the user adjusts the obfuscation level and reveals the obfuscated rating for a new item: the exact process, which takes into account the targeted accuracy level and modifies the obfuscation level according to the accuracy bounds, is described in the next section.

A. Targeting a Selected Accuracy Level

1) Evaluation Metrics: The basic premise in this work is that, rather than selecting a privacy level and evaluating the resulting accuracy, we select a desired accuracy level and adjust privacy (i.e., the level of added noise) accordingly.

We assume that each user has a set of item ratings that are not disclosed to the recommender system, and refer to them as the canary items. Let I_u denote the set of canary items for user u, and let r_{ui} be the rating that user u gave to canary item i, where $r_{ui} \in [r_{\min}, r_{\max}]$. We denote by $\hat{r}_{ui}$ the system prediction for u's rating for item i. The Root Mean Square Error (RMSE) of user u's canary set is given by:

$$RMSE_{I_u} = \sqrt{\frac{\sum_{i \in I_u} (r_{ui} - \hat{r_{ui}})^2}{|I_u|}} \quad . \tag{8}$$

We rely on the RMSE of the canary set as a measure for the expected accuracy of the recommendations.

2) Adaptive Obfuscation Algorithm: Given a set of canary items I_u of user u, Eq. 8 provides the RMSE of the canary set. To control the quality of information that the recommender system has about him, the user aims to maintain a recommendation system accuracy while obfuscating his ratings (sent sequentially to the system). The user defines the lower and upper bounds on prediction accuracy (lower error will result in accurate profiling of the user, while higher error will result in meaningless recommendations), denoted by e_l and e_u respectively, and consequently adjusts the obfuscation level if the prediction errors are not in the desired region, as detailed in Algorithm 1.

B. Offline Evaluation

In this section we apply the proposed algorithm to the MovieLens $100K$ dataset of to study the trade-off between privacy and accuracy.

Algorithm 1: Adaptive obfuscation: rating a new item

Input :
r – rating for a new item i
I_u – a canary set of items rated by user u
e_l, e_u – lower/upper bounds for canary RMSE
$p_{\min}$, $p_{\max}$ – minimum/maximum obfuscation levels
$p \in [p_{\min}, p_{\max}]$ – current obfuscation level
Output:
Obfuscated rating for the new item

1: $e \leftarrow RMSE(I_u)$
2: **if** $(e < e_l$ *and* $p < p_{max})$ **then** increment(p)
3: **else if** $(e > e_u$ *and* $p > p_{min})$ **then** decrement(p)
4: **return** *Obfuscate* (r, p)

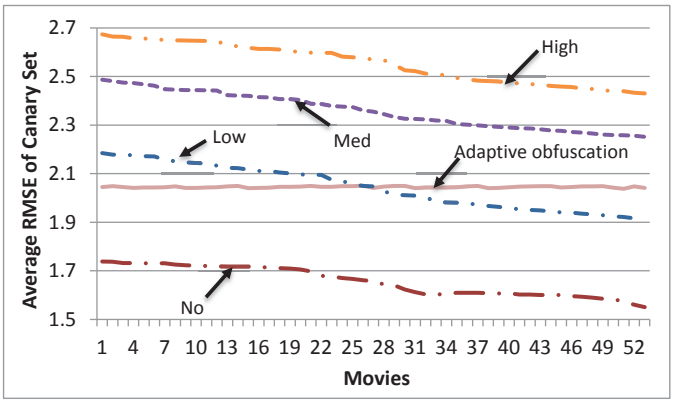


Fig. 4. Accuracy of the predictions of canary set: Randomly chosen canary set

Since the obfuscation is applied only to the user ratings, our approach can be used with any recommendation system. Without loss of generality, in the evaluation we use matrix factorization as a recommender algorithm, with the same parameters as in the previous section (see Table I).

We use Laplace noise for obfuscation of user ratings. Each user is provided with four obfuscation levels, {*none*, *low*, *medium* and *high*}. Each level is associated with Laplace noise with standard deviations $\sigma = 0, \sqrt{2}, 4, 4\sqrt{2}$ respectively.[5]

To avoid the cold start problem in the recommender algorithm, we start with a non-empty set of known ratings that are revealed to the recommender system (without any obfuscation). This allows the recommender system to initially learn about the user's preferences, and obfuscating any additional ratings provided by the user will help the user maintain privacy.

To evaluate system performance, we randomly selected 100 users who have rated at least 75 movies. We then randomly chose for each of those users a set of 10 movies as the canary set (denoted by I_u), a set of 5 movies to bootstrap the user profile (i.e., the user reveals the ratings of these five

[5]For ratings in the range $[1, 5]$, in the differential privacy framework these noise levels can be interpreted as corresponding to event-level privacy with $\epsilon = \infty$, $\epsilon = 4$, $\epsilon = 1.4$, and $\epsilon = 1$ respectively.

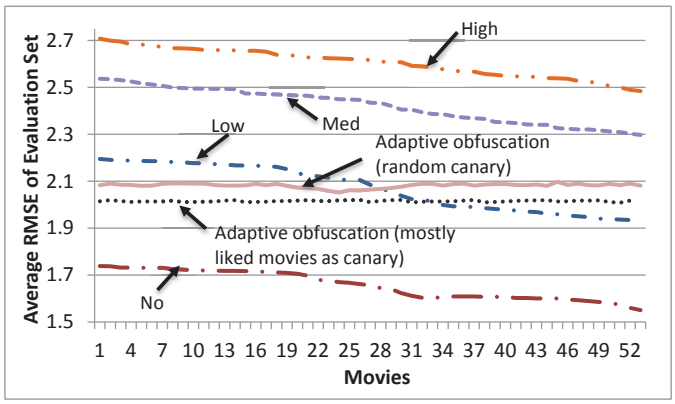


Fig. 5. Accuracy of the predictions of evaluation set for randomly chosen and movies that the users like most as canary sets

movies without obfuscation) and 10 movies as an evaluation set (denoted by E_u). We note that while the RMSE for canary movies reflects to users the expected level of rating prediction accuracy, we use the RMSE of the evaluation set to measure the actual accuracy of recommendations given to the user based on the obfuscated ratings, i.e., the utility of the system. Since the canary set was chosen as a random sample of the available ratings, we expect that the RMSE of the evaluation set would follow the RMSE for the canary set. We chose $e_l = 2$ and $e_u = 2.1$ as the user's expected lower and upper accuracy bounds for the canary set.

Fig. 4 shows the average RMSE in predictions evaluated for the canary set of randomly selected 100 users, for various levels of obfuscation, as each user provides additional movie ratings to the system. We introduce four baseline strategies that correspond to different fixed obfuscation levels; *no*, *low*, *medium* and *high*, in which the user samples noise from the same distribution and adds it to each rating before forwarding it to the system. Each point in the plot is obtained by averaging the RMSE over 100 simulation runs per user, and then averaging it over the 100 users.

As discussed in the previous section, the RMSE of the baselines decrease over time, despite the addition of noise, as the effect of the white noise averages out as the user rates an increasing number of items. In contrast, when using adaptive obfuscating of user ratings, the user agent aims to keep the RMSE of the canary set stable over time, within the accuracy bounds set by the user (i.e., on average the prediction error is kept at about the same level).

To evaluate the performance of the recommender system, we plot in Fig. 5 the average error in RMSE over the evaluation set over time (as the users rate more movies) for various obfuscation levels. Obfuscation imposes a penalty, increasing the error in prediction. However, similarly to the canary set RMSE, the impact of obfuscation on prediction error fades as the users keep using the same fixed obfuscation level (see the baselines corresponding to *no*, *low*, *medium* and *high* levels). In contrast, the adaptive obfuscation algorithm effectively keeps the average prediction error stable as users rate more

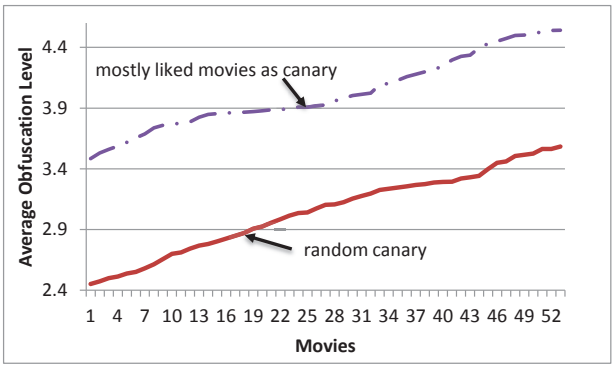


Fig. 6. Average obfuscation level of the adaptive obfuscation mechanism

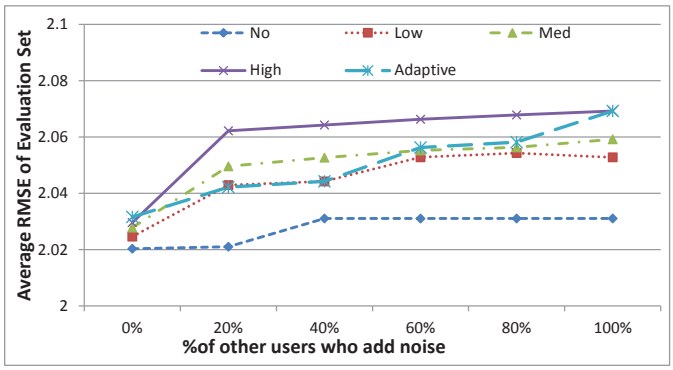


Fig. 7. Accuracy of the predictions of evaluation set when % of other users add noise

movies.

We note that the resulting privacy provided by the system is best-effort, as, considering the privacy metric (differential privacy) described in Section III-B, each individual rating item belonging to a selected user will be protected according to the corresponding level of obfuscation noise.

1) Choice of the Canary Set: To evaluate how the choice of movies for the canary set impacts the performance of the system, we further plot in Fig. 5 the RMSE of the evaluation set over time, given a canary set that contains the highest rated movies (i.e., "most liked" movies).

Since the baselines (corresponding to *no*, *low*, *medium* and *high* obfuscation levels) are independent of the canary set, they are essentially the same in both experiments. When comparing the lines corresponding to adaptive obfuscation for two different canary sets in Fig. 5, we observe that even though the average RMSE of the evaluation set stays between the boundary levels, choosing the mostly liked movies as canary set yields lower average RMSE compared to the random canary set.

To see how the obfuscation level adaptively changes to provide stable accuracy in recommendations, Fig. 6 shows the average obfuscation level over time for the experiments with the different canary sets. The user needs to add more noise (i.e., higher obfuscation levels) when the canary set is chosen as the movies most liked by the users, compared to the case where the canary set was selected at random. Albeit on average, random canary allows the user to choose lower obfuscation levels compared to other canary sets to stay within the desired accuracy bounds, it is still unclear how much each user's prediction error over the evaluation set deviates from the lower and upper bounds.

This motivates us to define a new metric called "user boundary miss" (denoted by Δ_k) of a user k who adaptively obfuscates his ratings to investigate the efficacy of the choice of canary movies. The user error Δ_k, captures how much the error in prediction of the evaluation set, i.e., $RMSE_{E_k}$, deviates from the desired accuracy bounds e_l and e_u. To quantify this, we consider the difference between $RMSE_{E_k}$ and e_l or e_u and average it over the set of movies H_k that the user k has rated using adaptive obfuscation:

$$\Delta_k = \sum_{H_k} max\{(RMSE_{E_k} - e_u), 0, (e_l - RMSE_{E_k})\}/|H_k| \tag{9}$$

By averaging this metric over the group of users S (in our case, it is 100 users) who adaptively obfuscate the ratings, we can calculate the "group boundary miss" Δ_S – how much a group of users' average error in prediction over the canary sets varies over time from the desired accuracy bounds.

$$\Delta_S = \sum_{S} \frac{\Delta_k}{|S|} \tag{10}$$

We use Δ_S over different selection strategies of canary sets (randomly chosen, movies the users like most, and movies the users like the least) to evaluate the best selection strategy for a canary set. Table IV shows Δ_S values for different canary sets. Most interestingly, choosing a random canary set causes the RMSE to deviate more from the desired bounds, yet the average still stays within the accepted boundary levels. In contrast, choosing extremely rated movies (i.e., most or least liked movies) for the canary set helps to keep the boundary misses at the lower level. This validates the point made in [11] that extreme ratings are *representative* of the users, and more important when generating predictions.

TABLE IV
CANARY SELECTION METHODS AND USER ERROR

Canary Set	Δ_S
Random movies	0.854
Most liked movies	0.412
Least liked movies	0.475

C. Adaptive Obfuscation and Heterogeneous Privacy Preferences

In this section we investigate how the prediction accuracy of a user who obfuscates the ratings adaptively, is influenced by input obfuscation done by a proportion of other users in the system. We consider a target user u, who adaptively obfuscates his ratings based on a randomly chosen canary set and explore the following scenario: a fraction of other users obfuscate their

responses with a privacy level p while the user u adaptively obfuscates his ratings based on his canary set. For this scenario we test the impact of the obfuscation conducted by other users on the prediction accuracy of the user u. The aim of this study is to see whether a user can effectively control his prediction accuracy by probing the system and changing the obfuscation level based on the feedback, regardless of other users conducting obfuscation in the system.

In the first approach, which was evaluated in Section IV-C, the user determined directly the privacy budget to spend, but did not have control on the accuracy of the recommendations, especially given the effect of other users' privacy choices. In contrast, in the approach considered in this section the user has direct control on the accuracy of the recommendations, but cannot determine in advance the privacy loss that will be incurred to maintain this level of accuracy.

To evaluate the aforementioned scenario, we use the same simulation setup described in Section V-B. The target user u adaptively obfuscates and reveals his noisy rating for a new movie i, while a proportion of the user community obfuscates their ratings of the same movie with a privacy level p. We calculate the average prediction error in the evaluation set of target user u over all the movies he rated with adaptive noise. We repeat this exercise with varying proportion of the user community and levels of privacy.

Fig. 7 shows the RMSE in predictions for the evaluation set of randomly selected 100 users. We choose $e_l = 2$ and $e_u = 2.1$ as the user's expected lower and upper accuracy bounds for the evaluation set. Each line of the plot depicts the average RMSE in predictions of a user who adaptively obfuscates while a fraction of other users in the system choose *no*, *low*, *medium* and *high* privacy. Each point in the plot is obtained by averaging RMSE over 100 simulation runs per user and then averaging it over the 100 users.

We observe that when a user adaptively obfuscates his ratings, his accuracy in predictions of the evaluation set stays stable over time, despite the obfuscation conducted by other users. In contrast to Figures 3(a), 3(b) and 3(c), adaptive obfuscation of user ratings enables the user to keep the accuracy fixed between the desired bounds regardless of the fraction of other users who obfuscate their ratings.

We are also interested in the scenario in which a target user and a fraction of other users adaptively obfuscate the ratings, and evaluate the average error in the predictions of the evaluation set of the target user. As evident in Fig. 7, adaptive obfuscation conducted by the target user allows him to keep the error in between the desired boundary levels regardless of the obfuscation method leveraged by other users in the system.

VI. Conclusion

In this paper we evaluated how the accuracy of ratings that a recommender system predicts for a user is affected both by input obfuscation conducted by that user and by obfuscation conducted by other users in the system. We showed that while the user's privacy choices would have the highest impact on the resulting accuracy, the cumulative effect of other users' choices would also have a significant influence on the prediction accuracy.

To allow a consistent performance of the recommender system, within accuracy bounds set by the user, we proposed and evaluated a privacy-aware interactive recommender system that can be probed by users to understand how well they were profiled by the system. Specifically, the system is requested to predict the ratings for a set of undisclosed items, the canary set. Obfuscation of ratings can take place on the client side, so this approach does not require a fully trusted recommender system. Users can obfuscate their inputs based on their level of comfort and based on the error in the predictions of the canary set, while receiving reasonably accurate personalized recommendations from the system. We evaluated our system off-line using a dataset of movie ratings, and showed how users can tune their noise level to receive meaningful recommendations, while maximizing obfuscation within the accuracy bounds.

References

[1] J. Calandrino, A. Kilzer, A. Narayanan, E. Felton, and V. Shamatikov, ""You Might Also Like:" Privacy Risks of Collaborative Filtering," in *IEEE Security and Privacy*, 2011.

[2] U. Weinsberg, S. Bhagat, S. Ioannidis, and N. Taft, "Blurme: inferring and obfuscating user gender based on ratings," in *RecSys*, 2012, pp. 195–202.

[3] J. Canny, "Collaborative Filtering with Privacy," in *IEEE Security and Privacy*, 2002.

[4] F. McSherry and I. Mironov, "Differentially Private Recommender Systems: Building Privacy into the Netflix Prize Contenders," in *Proc. KDD*, 2009.

[5] R. Agrawal and R. Srikant, "Privacy-preserving data mining," in *SIGMOD Conference*, 2000, pp. 439–450.

[6] D. Agrawal and C. C. Aggarwal, "On the design and quantification of privacy preserving data mining algorithms," in *PODS*, 2001.

[7] C. Dwork, F. Mcsherry, K. Nissim, and A. Smith, "Calibrating Noise to Sensitivity in Private Data Analysis," in *TOCS*, 2006.

[8] J. Canny, "Collaborative Filtering with Privacy via Factor Analysis," in *ACM SIGIR*, 2002.

[9] B. N. Miller, J. A. Konstan, and J. Riedel, "Pocketlens: Toward a Personal Recommender System," in *ACM Trans. Inf. Sys*, 2004.

[10] S. Berkovsky, Y. Eytani, T. Kuflikk, and F. Ricci, "Enhancing Privacy and Preserving Accuracy of a Distributed Collaborative Filtering," in *ACM RecSys*, 2007.

[11] S. Berkovsky, T. Kuflikk, and F. Ricci, "The Impact of Data Obfuscation on the Accuracy of Collaborative Filtering," in *Expert Systems with Applications*, 2012.

[12] R. Parameswaran and D. M. Blough, "Privacy Preserving Collaborative Filtering Using Data Obfuscation," in *IEEE Int. Conf. in Granular Computing*, 2007.

[13] H. Polat and W. Du, "Privacy Preserving Collaborative Filtering Using Randomized Perturbation Techniques," in *IEEE ICDM*, 2003.

[14] A. Machanavajjhala, A. Korolova, and A. D. Sarma, "Personalized Social Recommendations - Accurate or Private," in *VLDB*, 2011.

[15] C. Dwork, "Differential Privacy: A Survey of Results," in *TAMC*, 2008.

[16] T. Chen, R. Boreli, M. Kaafar, and A. Friedman, "An the Effectiveness of Obfuscation Techniques in Online Social Networks," in *PETS*, 2014.

[17] Y. Koren, R. Bell, and C. Volinsky, "Matrix Factorization Techniques For Recommender Systems," in *IEEE Computer Society*, 2009.

[18] C. Dwork, K. Kenthapadi, F. Mcsherry, I. Mironov, and M. Naor, "Our Data, Ourselves: Privacy via Distributed Noise Generation," in *Proc. EUROCRYPT*, 2006.

[19] Z. Gantner, S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "MyMediaLite: A free recommender system library," in *Proc. RecSys*, 2011.

[20] MovieLens. [Online]. Available: http://www.grouplens.org/node/73

[21] R. Axelrod, *The Evolution of Cooperation*. New York: Basic Books, 1984.