Privacy concerns in Recommender System

Mani Madhukar
Technical Lead
IBM India Pvt. Ltd., Noida, India

Abstract- Privacy remains the biggest concerns with recommender systems, since users can be exposed to spam in the name of recommender systems. Recommender system provides for various recommendation methodologies and each differs in the terms of user exposure. User should be llowed fine-grained control over the type of applications or data that can be accessed from the user's digital existence.

Keywords – Recommender system, Content Filtering, Collaboration Filtering, privacy-preserving, covariance

I. INTRODUCTION

Recommender systems or **recommendation systems** are a subclass of information filtering system that seek to predict the 'rating' or 'preference' that user would give to an item. Recommender systems have become very common in recent years, and are applied in a variety of applications like movies, music, news, books, research articles, search queries, social tags, and products in general. However, there are also recommender systems for experts, jokes, restaurants, financial services, live insurances, persons dating online, and twitter followers.

Recommender systems typically produce a list of recommendations in one of two ways - through collaborative filtering or content-based filtering. Collaborative filtering approaches build a model from a user's past behavior as well as similar decisions made by other users; then use that model to predict items or ratings for items that the user may have an interest in. Content-based filtering approaches utilize a series of discrete characteristics of an item in order to recommend additional items with similar properties. These approaches are often combined.

Each type of system has its own strengths and weaknesses. Content-based filtering requires a large amount of information on a user in order to make accurate recommendations. This is an example of the cold start problem, and is common in collaborative filtering systems. While collaborative filtering needs very little information to get started, it is far more limited in scope. Recommender systems are a useful alternative to search algorithms since they help users discover items they might not have found by themselves. Interestingly enough, recommender systems are often implemented using search engines indexing non-traditional data.

II. PRIVACY-PROTECTION TECHNOLOGIES

A wide variety of privacy issues associated with recommender systems have been raised. Research from many areas could be applied to improve some of the above mentioned concerns. An overview of research areas, their mechanisms, advantages, and limitations have been discussed below.

Awareness

Research in this mainly social field aims to enhance user awareness of the privacy issues that exist within online systems. It can aid users in specifying their privacy boundaries. The Platform for Privacy Preferences (P3P) [8] is an initiative that aims to provide websites with a standardized format in which they can define their privacy policy. Visitors of the website can then, through client-side user agents (e.g. plugins for their browser or applets), easily check the details of a privacy policy and see what will happen to information they submit. This system can help to increase user awareness, but only for users that employ agents and if websites properly define their privacy policies and adhere to them. Tsai et al. [21] showed that when privacy information is shown more prominent, and users are made more aware of the privacy consequences, privacy is taken into account when shopping online.

Law and Regulations

This legal field of research aims to find proper and broad laws and regulations that protect the users' privacy, while not greatly hindering businesses. It also focuses on compliance of both users and service providers to established laws and social conducts. Laws and regulations form an important and much needed tool. For example, the Article 29 Working Party has been working towards regulations for online behavioral advertising [14]. This legal approach runs after the technology, as specific laws dealing with personal information as related to the internet often take long to be developed. Also, laws are generally used to solve matters after things go wrong, whereas most technical solutions attempt to prevent violations.

Anonymization

Service providers may try to remove the privacy issues associated with data sales, by obscuring the link between users and data sold [20]. This can be done through anonymization, which involves removing any identifiable information from the data, while preserving other structures of interest in the data. As mentioned before, the information published by Netflix as part of their recommender systems prize, though anonymized, allowed for reidentification [16]. This mainly stems from the fact that information can only be partially removed or obfuscated, while other parts must be kept intact for the dataset to remain useful. In the real world, it is difficult to predict which external sources of information may become available, allowing pieces of data to be combined into identifiable information.

When looking at anonymization during recommendation, Ciss´ee and Albayrak [7] utilized trusted agents (essentially moving the trust around) to act as a relay and filter the information that is sent. This way the user can interact (through the agent) with the recommender system in an anonymous way. The user hides his personal information from the service provider, and is safe from the service provider linking his rating information to a person.

Randomization and Differential Privacy

Similar to anonymization is randomization. In randomization (sometimes referred to as perturbation), the information fed into the system is altered to add a degree of uncertainty. Polat and Du [17] proposed a singular value decomposition predictor based on random perturbation of data. The user's data is perturbed by adding a random value (from a fixed distribution) to each of the ratings; unknown ratings are filled in with the mean rating. They go on to show the impact on privacy and accuracy, and their inherent trade-off due to perturbation. In later work [18] their setting is different. A user wants two companies to collaboratively compute recommendations for him. This user acts as a relay for the two companies. The user's privacy is based on randomizing values. Berkovsky et al. [4] proposed to combine random perturbation with a peer-to-peer structure to create a form of dynamic random perturbation. For each request, the user can decide what data to reveal and how much protection is put on the data. Different perturbation strategies are compared based on accuracy and perceived privacy. Shokri et al. [19] added privacy by aggregating user information instead of perturbing. Aggregation occurs between users, without interaction with the recommender system. Thus, the recommender system cannot identify which information is part of the original user information and what is added by aggregation. A degree of uncertainty is added to the user's information similar to randomization.

The field of randomization is shifting towards differential privacy [9], which aims to obscure the link between single users' information in the input (the user's information) and output (the recommendation). This is accomplished by making users in released data computationally indistinguishable from most of the other users in that data set. This is typically accomplished by adding noise to the inputs or output, to hide small changes that arise from a single user's contribution. The required level of noise depends on how and how often the data will be used, and typically involves a balancing act between accuracy of the output and privacy of the input. Such in-distinguish-ability also applies strongly to collaborative recommender systems, where a user should be unable to identify individual peers' ratings in the output he receives. As each recommendation leaks a little bit of information about the input (even with noise), with a larger number of recommendations, the added noise should be greater to provide the same level of privacy. McSherry and Mironov [15] proposed collaborative filtering algorithms in the differential privacy framework. Noise is added to the item covariance matrix (for item similarity). Since the item covariance matrix is smaller than the user covariance matrix, less noise needs to be added and more accuracy is preserved.

Privacy-Preserving Cryptographic Protocols

Among the tools for privacy-preserving cryptographic protocols [12] are secure multi-party computations, secret sharing, homomorphic encryption, and zero-knowledge proofs.

Secure multi-party computations are a class of protocols that allow two or more parties to collaboratively compute a function based on input held by each of them. The output of this function can be given to one of the parties or all of them. Any function can be computed, but the complexity of the protocol depends on the function. For example, multiplication and integer comparison.

Secret sharing distributes a number of shares of a value among different parties. The shares of a fixed number of parties need to be combined in order to reconstruct the original value. With less than the fixed number of shares, no information about the value can be obtained. Some secret sharing schemes allow basic operations (such as addition) to be performed.

Homomorphic encryption allows one or more operation (for example addition or multiplication) on the encrypted values, by performing a corresponding operation on the ciphertexts. This allows anyone to compute a (basic) function on the encrypted values, without knowledge of the actual values. Decryption is then required to get the result of the function. Zero-knowledge proofs allow a user to prove a property about a value, without revealing that value. For example, that a value is in a given range of possible values. To do this, the user first sends a commitment to the verifier. Then the verifier ask the user to open the commitment in a certain way. The commitment can only be opened correctly when the property of the value holds. With a certain probability the user can correctly open the commitment even if the property does not hold. However, by running multiple zero-knowledge proofs this percentage can be reduced.

Privacy-Preserving Cryptographic Protocols without Server

Privacy-preserving cryptographic protocols without a central server aim to remove the trust that is placed in service providers by removing them from the picture. Secure multi-party computations protect the privacy of users against each other. Canny [5, 6] used a combination of secure multi-party computation, homomorphic encryption, and zero-knowledge proofs to create a privacy-preserving recommender protocols without a central server. The users collaborate to privately compute intermediate values of the collaborative filtering process. These intermediate values (based on all users) are then made public. In the next step the users perform singular value decomposition and factor analysis, which leads to a model for recommendations. This model is made publicly available and can be used by each user independently to compute recommendations for them.

The system proposed by Hoens et al. [13] allowed trusted friends to collaboratively compute recommendations with each other. They rely on Facebook for retrieving friendship information and a server to facilitate asynchronous messaging. Homomorphic cryptography and secure multi-party protocols are used to compute the actual recommendations for a given item. Because a decentralized structure works strongly towards taking power away from the service provider, it is contrary to existing business models. This means that existing companies are not likely to adopt such a structure, or aid its development. Another drawback is the involvement of many users, that is required to make the recommendations. These users need to interact with each other, but not all users will be available at the same time. This can lead to considerable delays, or a loss of accuracy.

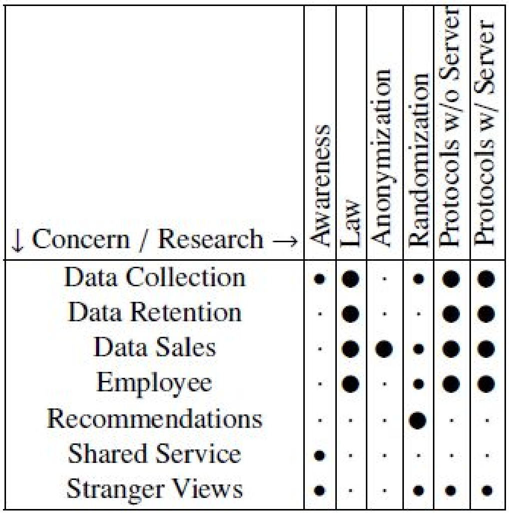
Privacy-Preserving Cryptographic Protocols with Server

Privacy-preserving cryptographic protocols with a central server, aim to make use of the centralization offered by the service provider, while using secure two-party computation and encryption to ensure the privacy of the users. Good motivations for the service provider would be a reduced liability for the data collected, an increased perception of security among users (and thus, a competitive advantage), and adherence to possible stricter future laws. A¨ımeur et al. [1] provided a framework for collaborative filtering, where user information is separately stored over two parties. An agent has access to ratings and the company has access to the items, so that they together can generate recommendations for the user. The centralized structure is preserved, but neither the agent nor the company can link the user's ratings to the items. Erkin et al. [10, 11] proposed a collaborative filtering algorithm based on homomorphic cryptosystems. In their framework, a central server acts as a mediator between the users and is in charge of combining the results given by different users. When desiring a recommendation, a user sends an encrypted request to the central server. The server distributes this request to other users that can work on the request by using the homomorphic properties of the cryptosystem. A secure two-party computation then determines for each user if their information should be included in the recommendation or not. The central server then combines the results to generate the recommendation. Basu et al. [2, 3] proposed a privacy preserving version of the slope one predictor for collaborative filtering. The assumption is that different parties hold different parts of the information; this essentially allows multiple companies to collaborate. They pre-compute the deviation and cardinality matrices under encryption and make the cardinality matrix public. Then the prediction for a single item can be computed under encryption and all parties collaborate to decrypt the result. The drawback of these schemes (that add a layer of encryption) is efficiency. The homomorphic operations and secure two-party computations are always more expensive than their unprotected counterparts. In fact, the discrepancy is often huge. This results in poor efficiency and scalability for these protocols.

III. REVIEW ON PRIVACY-PROTECTION TECHNOLOGIES

Refer to the Table 1 shown below in which research areas contribute to address which privacy concern is related to which. The majority of the research areas focus on protecting the user's information from the service provider. The privacy concerns related to the service provider have a high privacy impact. None of the research areas mentioned in this section can offer complete user privacy for all recommender systems. Privacy is multi-faceted, as are the domains in which recommender systems are applied. Several areas will likely need to be combined to develop proper privacy-protection techniques for a given application. In addition, service providers should be encouraged or required to implement such solutions, and users need to be made aware of the benefits of using them.

Table 1. Privacy concerns and relevant research areas.



↓ Concern / Research →	Awareness	Law	Anonymization	Randomization	Protocols w/o Server	Protocols w/ Server
Data Collection	•	●	·	•	●	●
Data Retention	·	●	·	·	●	●
Data Sales	·	●	●	•	●	●
Employee	·	●	·	•	●	●
Recommendations	·	·	·	●	·	·
Shared Service	•	·	·	·	·	·
Stranger Views	•	·	·	•	•	•

Recommender systems play an important role in the online experience of millions of people. While accuracy has been the focus in recommender system development, we argue that privacy should not be overlooked. We have seen that depending on the type of information utilized by a recommender system, various privacy concerns exist. The fact that trust in the service provider is not always justified further complicates matters. With increased information-sharing, users must weigh the advantages of getting (more accurate) recommendations against the privacy risks, and should more often be given the choice to opt-in or opt-out of data collection.

As commonly known, in the technical solutions there is an inherent trade-off between privacy, accuracy, and efficiency. Randomization techniques increase privacy by lowering accuracy, and leaving efficiency the same.

DIFFERENTIALLY PRIVATE RECOMMENDER SYSTEMS

We base our choice of algorithms on leading solutions to the Netflix Prize [25, 26, 27]. We adapt algorithms exemplifying two approaches that emerged as main components of Netflix Prize contenders: factor models and neighborhood models. Algorithms such as k-means clustering, perceptron classification, association rule mining, decision tree induction, and low rank approximation are all shown to have differentially private analogues, with theoretical bounds on their accuracy [28]. The wholesale release of data with anonymized user identities by Neflix has been shown to have far-reaching privacy implications [36], establishing, in particular, that most rows can be identified with near certainty based on as few as a dozen partially known data points. Although a commercial recommender system is unlikely to willingly disclose all or substantial fraction of its underlying data, a recent work by Calandrino et al. [29] demonstrates that passive observations of Amazon.com’s recommendations are sufficient to make valid inferences about individuals’ purchase histories. The focus of prior work on cryptographic solutions to the problem of secure recommender systems is on removing the single trusted party having access to everyone’s data [30, 31, 23, 24]. It does not attempt to limit amount of information leaked through the system’s recommendations in the course of its normal execution. Our solution can be combined with the modular approach of the Alambic framework [23]. It is important to distinguish our approach, of privacy preserving computation, from much prior work on privacy studying the release of anonymized records. One could imagine building a recommender system, or any machine learning technology, on top of anonymized data, drawing privacy properties from the anonymization rather than reproducing them itself. However, especially for rich, high-dimensional data, most anonymization techniques appear to cripple the utility of the data [8, 22]. By integrating the privacy guarantees into the application, we can provide it with unfettered access to the raw data, under the condition that its ultimate output—substantially less information that an entire data set—respect the privacy criteria.

IV. RECOMMENDATION ALGORITHMS

The setting we consider has both users and items, with ratings for a subset of the (user, item) pairs. Given such a partial set of ratings, the goal is to predict certain held out values at specified (user, item) locations.

Global Effects. A common first step in these systems is to center the ratings by computing and subtracting average ratings for users and for items. To stabilize this computation, the average is often computed including an additional number of fictitious ratings at the global average; users and movies with many ratings drift to their correct average, but averages with small support are not allowed to overfit. **Covariance.** Having factored first order effects, derived from properties of the ratings themselves, it is very common to look at correlations between items.1 A common approach is to look at the covariance matrix of the items, whose (i, j) entry is the average product of ratings for items i and j across all users. Of course, relatively few users have actually rated both i and j, and so the average is taken across only those users who have rated both items.

Geometric Recommendation Algorithms. Oversimplifying tremendously, to a first approximation once we have computed the covariance matrix of the items we have enough information at hand to apply a large number of advanced learning and prediction algorithms. The covariance matrix encodes the complete geometric description of the items, and any geometric algorithm (e.g.: latent factor analysis, nearest neighbor approaches, geometric clustering, etc) can be deployed at this point. Importantly, from our perspective, these approaches can be applied for each user using only the covariance information and the user's collection of ratings. If the covariance measurement can be conducted privately, any algorithm that does not need to return to the raw data of other users can be deployed at this point with privacy guarantees.

A NON-PRIVATE APPROACH

We will formalize the previous sketch into an algorithm that is non-private, but will form the skeleton of our privacy preserving approach. The steps in the algorithm may appear especially pedantic, but writing them in a simplistic form will allow us to adapt them easily to their private forms. Following [25] we use r to refer to a collection of ratings, with the notation r_{ui} for the rating of user u for movie i and r_u for the vector of ratings for user u. We use the notation e_{ui} and e_u for the binary elements and vectors indicating the presence of ratings (allowing us to distinguish from reported zero values). We start by subtracting the movie averages from each movie, where the average is dampened by a number β of ratings with the global average.

Movie Effects

1. For each item i, compute totals and counts:

(a) Let $MSum_i = \Sigma_u r_{ui}$

(b) Let $MCnt_i = \Sigma_u e_{ui}$

2. Compute global average $G = \Sigma_i MSum_{i/}\Sigma_i MCnt_i$

3. For each item i, compute the stabilized average:

(a) Let $MAvg_i = (MSum_i + \beta G)/(MCnt_i + \beta)$.

4. For each rating r_{ui}, subtract the appropriate average:

(a) Set $r_{ui} = r_{ui} - MAvg_i$.

We perform exactly the same operation for the users, computing stabilized averages and subtracting the appropriate averages from each rating.

B. DIFFERENTIAL PRIVACY

Differential privacy [32], surveyed in [33], is a relatively recent privacy definition based on the principle that the output of a computation should not allow inference about any record's presence in or absence from the computation's input. Formally, it requires that for any outcome of a randomized computation, that outcome should be nearly equally likely with and without any one record. We say two data sets A and B are adjacent, written A ≈ B, if there is exactly one record in one but not in the other.

Definition 1. A randomized computation M satisfies ∈-differential privacy if for any adjacent data sets A and B, and any subset S of possible outcomes Range(M),

$\mathbf{Pr}[M(A) \in S] \leq \exp(\_) \times \mathbf{Pr}[M(B) \in S]$.

One interpretation of the guarantee differential privacy provides is that it bounds the ability to infer from any output event S, whether the input to the computation was A or B. From an arbitrary prior p(A) and p(B), we see that

$p(A|S) /p(B|S) = p(A)/ p(B)\times p(S|A)/ p(S|B)$

.

When A ≈ B, differential privacy bounds the update to the prior by a factor of exp(∈), limiting the degree of inference possible about slight differences in the input data sets. Specifically, inference about the presence or absence (and consequently the value of) any single record is bounded by a factor of exp(∈).

C. APPROXIMATE DIFFERENTIAL PRIVACY

We will also consider a relaxed form of differential privacy that permits an additive term in the bound, as well as the multiplicative term, introduced in [33].

Definition 2. A randomized computation M satisfies (∊, δ)- Differential privacy if for any adjacent data sets A and B, and any subset S of possible outcomes Range(M),

Pr[M(A) ∈S] ≤ exp(_) × **Pr**[M(B) ∈S] + δ .

One interpretation of this guarantee is that the outcomes of the computation M are unlikely to provide much more information than for _-differential privacy, but it is possible. For any γ > ∊, take Sγ to be the set of outcomes x for which p(x|A)/p(x|B) > exp(γ).

Combining this constraint with the definition of (∊, δ)-differential privacy, we can conclude that such outputs are unlikely:

p(Sγ|B) ≤ p(Sγ|A) ≤ δ /(1 − exp(_ − γ))

While γ much larger than _ is possible, the probability is effectively bounded by δ. Moreover, for the privacy mechanisms we use, there will always be a trade-off between _ and δ; one can decrease either arbitrarily, at the expense of increasing the other. In a sense, the amount of information released (measured as the ratio of the two probabilities) is a random variable which is most likely to be small.

D. NOISE AND SENSITITIVITY

The simplest approach to differential privacy when computing numerical measurements is to apply random noise to the measurement, and argue that this masks the possible influence of a single record on the outcome. If we aim to compute a function f : Dn → Rd, the following results describe prior privacy results achieved through the addition of noise [34].

E. COUNTS, AVERAGES AND CO-VARIANCES

There are relatively few statistics we will need to measure from the data to begin adapting recommendation algorithms from previous work. Global effects, such as per-movie averages and per-user averages, play an important role in prediction. Additionally, the movie-movie covariance matrix forms the basis of many geometric algorithms, and specifically the SVD factorization approaches and the kNN geometric distance approaches. Before continuing to the specifics of our approach, we see how these quantities can be measured in the previously described frameworks. Consequently, we can report counts of arbitrary partitions of the records (our interest is in ratings per movie) with appropriate additive noise providing privacy.

F. ALGORITHM AND ANALYSIS

Our algorithm consists of several steps, measuring (with noise) progressively more challenging aspects of the data before feeding the measurements to appropriately parameterized variants of the currently top learning algorithms.

Global effects. We start with the noisy measurement of and baseline correction for various global effects. We first measure and publish the sum and count across all ratings to derive a global average. We then measure and publish, for each movie, the number and sum of ratings for that movie. We use these two quantities to produce a per-movie average, stabilized by including a number βm of ratings at the global average. Finally, we remeasure the global average, as above, for upcoming use in centering each user's ratings. We next invest some effort in preparing each user's ratings for covariance measurement. We do not want to release per-user statistics, such as the average rating for each user, as to do so with sufficient accuracy to be useful for learning would demolish our privacy guarantees. Instead, we will apply several transformations to a user's ratings before measurement, and argue that the transformations are such that privacy guarantees made of their outputs propagate to their inputs. Our specific operations include the centering of each user's ratings, again including a number of fictitious ratings at the global average, as well as a clamping of the resulting value to a more compact interval (increasing privacy, at the expense of error in outlying values).

EVALUATION

We evaluate our approach on the Netflix Prize data set that consists of roughly 100M ratings of 17770 movies contributed by 480K people. By adjusting the parameters of
the noise distributions we use, our computation will provide varying differential privacy guarantees, and its output will have measurable accuracy properties. The accuracy is measured by the root mean squared error (RMSE) on the qualifying set (3M ratings) and can be self-tested on the probe set with similar characteristics (1.5M ratings).

The Privacy v. Accuracy Tradeoff

While it is natural to parameterize differential privacy using a variety of (∊, δ) pairs, we simplify to a single parameter. For each measurement fi, we will parameterize the magnitude of the noise we use as

$$\sigma_i \quad = \quad \max_{A \approx B} \|f_i(A) - f_i(B)\| / \theta_i \ ,$$

where the θi are required to sum to a pre-specified value θ.In fact, we will take each θi to be a fixed fraction of θ, whose value we will take and vary as our single parameter. Using Laplace noise, measurement i provides ∈i-differential privacy for

$$\epsilon_i \;=\; \theta_i.$$

By Theorem 3, using Gaussian noise, measurement i provides (∈i, δi)-differential privacy for

$$\epsilon_i \;=\; \theta_i\sqrt{2\ln(2/\delta_i)}.$$

As Theorem 1 tells us that _ and δ values add, our final
guarantees have the form (for || · ||1 and || · ||2 respectively)

$$\epsilon = \sum_i \theta_i \quad \text{and} \quad \epsilon = \sum_i \theta_i\sqrt{2\ln(2/\delta_i)}$$

By taking a common value of δi, we can see that θ =Σ_i θi scales the value of _ linearly. By varying θ, and thus the θi, we can add more or less noise to our measurements and provide more or less privacy, respectively. From any θ, we can reconstruct a range of (∈, δ) pairs.

5.2 Privacy v. Accuracy over Time

Our algorithms (like most differentially-private computations) introduce a fixed amount of error to any measurement, that is increasingly dominated by actual data records as the size of the data sets increase. With more and more users and ratings, we expect the additive error we introduce for any fixed value of θ to eventually vanish. To explore how the loss due to the privacy-preserving property of the recommender mechanism decreases with the amount of available data (for the fixed value of θ = 0.15), we simulated the data gathering process at different times between 2000 and 2006 (including the peculiar property of including users with fewer than 20 ratings). Consistently with the Netflix Prize data set, the probe set was the 9 most recent ratings for each user chosen with probability 1/3 each. Fig. 1 plots the difference in RMSE (as percentage points) between privacy-preserving k-NN (after scaling) and the same algorithm without privacy guards.

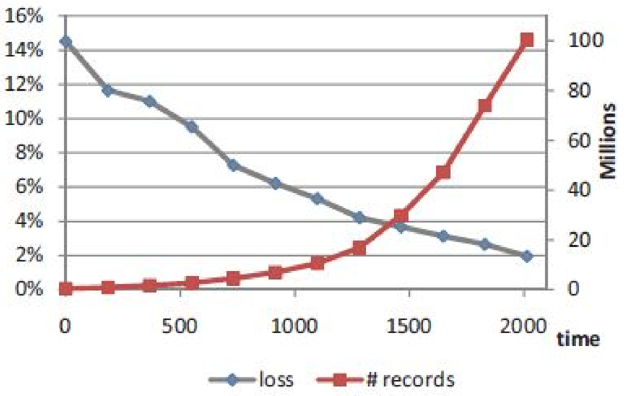


Fig 1: Left scale—accuracy loss, right scale—the number of records. The x-axis is the number of days elapsed. 7/1/2000.

V. CONCLUSIONS AND FUTUREWORK

We conclude that a recommendation system with differential privacy guarantees is feasible without taking significant hit in the recommendations accuracy. The loss in accuracy (for a fixed value of the privacy parameter) decreases as more data becomes available. In our experiments we fixed several parameters that had the potential to vary freely, and it is natural to expect that more in-depth experimentation could lead to noticeably improved prediction accuracy. The chosen dimensionalities, smoothing weights, and distribution of "accuracy" θi between the measurements could be adjusted and possibly improved. Directions for future work include efficient methods for direct privacy-preserving computations of latent factors and incorporation in the differential privacy framework of advanced methods for collaborative filtering that do not immediately admit factorization into two phases such as the integrated model of [35].

REFERENCES

1. Esma A¨ımeur, Gilles Brassard, Jos Fernandez, and Flavien Mani Onana. Alambic: a privacy-preserving recommender system for electronic commerce. International Journal of Information Security, 7(5):307–334, 2008.

2. Anirban Basu, Hiroaki Kikuchi, and Jaideep Vaidya. Privacy-preserving weighted slope one predictor for item-based collaborative filtering. In Proceedings of the international workshop on Trust and Privacy in Distributed Information Processing, 2011.
3. Anirban Basu, Jaideep Vaidya, and Hiroaki Kikuchi. E_cient privacy-preserving collaborative filtering based on the weighted slope one predictor. Journal of Internet Services and Information Security (JISIS), 1(4):26–46, 11 2011.
4. Shlomo Berkovsky, Yaniv Eytani, Tsvi Kuflik, and Francesco Ricci. Enhancing privacy and preserving accuracy of a distributed collaborative filtering. In Proceedings of the
2007 ACM conference on Recommender systems, pages 9–16, 2007.
5. John Canny. Collaborative filtering with privacy. In IEEE Symposium on Security and Privacy, pages 45–57, 2002.
6. John Canny. Collaborative filtering with privacy via factor analysis. In Proceedings of the 25th annual international conference on Research and development in information retrieval, pages 238–245, 2002.
7. Richard Ciss´ee and Sahin Albayrak. An agent-based approach for privacy-preserving recommender systems. In Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems, AAMAS ’07, pages 182:1–182:8, New York, NY, USA, 2007. ACM.
8. Lorrie Cranor, Marc Langheinrich, Massimo Marchiori, Martin Presler-Marshall, and Joseph Reagle. The platform for privacy preferences 1.0 (p3p1.0) specification. online.
http://www.w3.org/TR/P3P/.
9. Cynthia Dwork. Di_erential privacy. In Automata, Languages and Programming, 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II, pages 1–12, 2006.
10. Zekeriya Erkin, Michael Beye, Thijs Veugen, and R.L. Lagendijk. Privacy enhanced recommender system. In Thirty-first Symposium on Information Theory in the Benelux, pages 35–42, 2010.
11. Zekeriya Erkin, Michael Beye, Thijs Veugen, and R.L. Lagendijk. E_ciently computing private recommendations. In International Conference on Acoustic, Speech and Signal
Processing-ICASSP, pages 5864–5867, 2011.
12. Oded Goldreich. Foundations of cryptography: a primer. Foundations and Trends in Theoretical Computer Science, 1:1–116, April 2005.
13. T. Ryan Hoens, Marina Blanton, and Nitesh V. Chawla. A private and reliable recommendation system for social networks. In Social Computing (SocialCom), 2010 IEEE Second International Conference on, pages 816–825, 8 2010.
14. Jacob Kohnstamm. Opinion 2/2010 on online behavioural advertising. Technical Report 00909/10/EN WP 171, Article 29 Data Protection Working Party, 6 2010. http://ec.europa.eu/justice/policies/privacy/docs/wpdocs/2010/wp171 en.pdf.
15. Frank McSherry and Ilya Mironov. Di_erentially private recommender systems: building privacy into the netflix prize contenders. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 627–636, 2009.
16. Arvind Narayanan and Vitaly Shmatikov. How to break anonymity of the netflix prize dataset. CoRR: Computing Research Repository, pages 1–24, 2006.
17. Huseyin Polat and Wenliang Du. Svd-based collaborative filtering with privacy. In Proceedings of the 2005 ACM symposium on Applied computing, pages 791–795, 2005.
18. Huseyin Polat andWenliang Du. Privacy-preserving top-n recommendation on distributed data. Journal of the American Society for Information Science and Technology, 59:1093–1108, 2008.
19. Reza Shokri, Pedram Pedarsani, George Theodorakopoulos, and Jean-Pierre Hubaux. Preserving privacy in collaborative filtering through distributed aggregation of o_ine profiles. In Proceedings of the third ACM conference on Recommender systems, RecSys ’09,
pages 157–164, New York, NY, USA, 2009. ACM.
20. Latanya Sweeney. k-anonymity: “A model for protecting privacy.” IEEE Security And Privacy, 10(5):557–570, 2002.
21. Janice Y. Tsai, Serge Egelman, Lorrie Cranor, and Alessandro Acquisti. The effect of online privacy information on purchasing behavior: An experimental study. Information Systems Research, 22:254–268, June 2011.
22. C. C. Aggarwal. On k-anonymity and the curse of dimensionality. In K. B¨ohm, C. S. Jensen, L. M. Haas, M. L. Kersten, P.-˚A. Larson, and B. C. Ooi, editors, VLDB, pages 901–909. ACM, 2005.
23. E. A¨ımeur, G. Brassard, J. M. Fernandez, and F. S. M. Onana. Alambic: a privacy-preserving recommender system for electronic commerce. Int. J. Information Security, 7(5):307–334, 2008.
24. E. A¨ımeur, G. Brassard, J. M. Fernandez, F. S. M. Onana, and Z. Rakowski. Experimental demonstration of a hybrid privacy-preserving recommender system. In ARES, pages 161–170. IEEE Computer Society, 2008.
25. R. M. Bell and Y. Koren. Scalable collaborative filtering with jointly derived neighborhood interpolation weights. In ICDM, pages 43–52. IEEE Computer Society, 2007.
26. R. M. Bell, Y. Koren, and C. Volinsky. The BellKor solution to the Netflix Prize. Available fromhttp://www.netflixprize.com, 2007.

27. R. M. Bell, Y. Koren, and C. Volinsky. The BellKor 2008 solution to the Netflix Prize. Available from http://www.netflixprize.com, 2008.
28. A. Blum, C. Dwork, F. McSherry, and K. Nissim. Practical privacy: the SuLQ framework. In C. Li, editor, PODS, pages 128–138. ACM, 2005.
29. J. Calandrino, A. Narayanan, E. Felten, and V. Shmatikov. Don't review that book: Privacy risks of collaborative filtering. Manuscript, 2009.
30. J. F. Canny. Collaborative filtering with privacy. In IEEE Symposium on Security and Privacy, pages 45–57, 2002.
31. J. F. Canny. Collaborative filtering with privacy via factor analysis. In SIGIR, pages 238–245. ACM, 2002.
32. C. Dwork. Differential privacy. Invited talk. In Automata, Languages and Programming—ICALP (2), volume 4052 of Lecture Notes in Computer Science, pages 1–12. Springer, 2006.
33. C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor. Our data, ourselves: Privacy via distributed noise generation. In S. Vaudenay, editor, Advances in Cryptology—EUROCRYPT 2006, volume 4004 of Lecture Notes in Computer Science, pages 486–503. Springer, May 2006.
34. C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In S. Halevi and T. Rabin, editors, Theory of Cryptography Conference—TCC 2006, volume 3876 of Lecture Notes in Computer Science, pages 265–284. Springer, 2006.
35. Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In Li et al. [20], pages 426–434.
36. A. Narayanan and V. Shmatikov. Robust de-anonymization of large sparse datasets. In IEEE Symposium on Security and Privacy, pages 111–125. IEEE Computer Society, 2008.