Privacy-preserving movie recommendation mechanism: recommendations with PNCF

Mingyang Sun
College of Arts and Science, Emory University
mingyang.sun@emory.edu

Abstract

Collaborative Filtering (CF) has been one of the most popular techniques in recommender systems because of its interpretability and fast nature. In recent years, the privacy-preserving mechanisms applied to this algorithm have drawn more attention. Especially for neighborhood-based CF, inappropriate privacy-preserving methods could be not powerful enough in protecting crucial information. In this case, a Private Neighbor Collaborative Filtering (PNCF) algorithm has been proposed by Zhu and et al. [1], which includes two operations: private neighbor selection and recommendation-aware sensitivity based on the notion of differential privacy. The utility and privacy guarantee have been mathematically analyzed in the paper. In this project I implemented the algorithm and modified it: in the paper, the perturbation stage adds Laplace noise to the similarities, yet I also implemented another perturbation operation which adds Laplace noise to the ratings. I compared the results between these two perturbation techniques.

I. Motivation

Recommender systems are widely applicable in e-commerce worldwide. Netflix and YouTube make use of users' activity history to provide personalized streaming service, and it's not a secret that Facebook, Google, Amazon and Taobao also conduct recommendation services by using users' browsing history or cookies. In this context, concerns with privacy issues have been more controversial recently: Netflix had to cancel the second Netflix Prize Competition due to lawsuit and users keep complaining about companies like Facebook acquiring their activity logs and worrying about their personal privacy. Several noteworthy approaches have been proposed in this circumstance, one of which is the PNCF algorithm. Interested in examining the powerfulness of this mechanism, I've implemented and tested it with a subset of Netflix dataset [2]. The results are illustrated in the evaluation section.

II. Related Work

A) *Privacy Preserving Techniques in Recommender Systems*: Several papers have focused on privacy violations. One of the most widely discussed attacks is claimed by Narayanan et al. [3] who re-identified users in published Netflix by linking with data from IMDB dataset. Another privacy violation discovered by Calandrino et al. [4] is the possibility to infer a user's rating history by observing the temporal differences in the outputs. Specifically, they issued KNN attack to neighbor-based CF methods. Traditional privacy preserving approaches include cryptographic, perturbation and obfuscation. Nevertheless these traditional methods are either costly in time or can harm utility with uncontrollable noise.

B) *Differential Privacy in Recommender Systems*: As differential privacy has been a formalized notion, researchers start exploring the efficient ways to incorporate its strength into the recommender systems. The very first significant work was conducted by McSherry et al. [5], who introduced differential privacy to the recommender systems by adding Laplace noise to the covariance matrix. Yet it has been proven that this method adds extra noise and fails to fight against KNN attack [4].

C) *Private Neighbor Collaborative Filtering*: This algorithm proposed by Zhu et al. [1] is specifically targeted to solving the KNN attack issues. By developing an innovative approach to select neighbors privately and adding perturbation noise wisely, this approach could be mathematically proven to be strongly resistant to privacy information leakage and offer significant utility guarantee.

III. Approach & Methodology

3.1 Anatomy of the Approach

In the paper, researchers summarized the pitfalls of traditional DP approaches in two aspects. 1) They are not able to hide similar neighbors. Suppose some malicious attacker has partial information of a target user's rating history, then the attacker is able to create k Sybil users. When using neighbor based CF, it's highly possible that retrieved k similar users include the other k – 1 Sybil users and the target user. In this case by carefully evaluating the recommendation results, attacker could infer the target user's rating history. 2) These algorithms ignore the unique characteristics of recommender systems, thus adding extra

noise to highly harm the utility of recommendations, and further exacerbating this shortcoming with naïve mechanisms.

Aiming at the weakness of classical DP approaches, they constructed PNCF algorithm with two private operations: the first stage is private neighbor selection, in which exponential mechanism, based on an improved sensitivity notion specifically targeting recommender systems, is used; and the second stage is perturbation, used to mask the result by simply adding naïve result. The general approach is illustrated in the following figure.

Algorithm 1 Private Neighbor Collaborative Filtering(*PNCF*)

Input: **R**, privacy parameter ϵ, truncated parameter w, number of neighbors k, u_a, t_i, I

Output: $\hat{r}_{ai}$

1. Compute item to item similarity Matrix S;
2. Private Neighbor Selection: select k neighbors $N_k(t_i)$ from I;
3. Perturbation: Perturb the similarity in $N_k(t_i)$ by adding $Lap(\frac{2k \cdot LS(i,\cdot)}{\epsilon})$ noise;
4. Predict $\hat{r}_{ai}$;

Besides implementing the original algorithm, I also developed the second version which in the second stage, perturbs the data by adding noise to ratings instead of similarities, and compared them in the evaluation section.

A) *Private Neighbor Selection*. This stage is the most crucial part in the overall algorithm, as it prevents the inferring on the neighbors. Apparently, naïve exponential mechanism cannot strongly satisfy the utility requirement of recommendations if it considers the global sensitivity. Therefore, a specific formula, *Recommender-Aware Sensitivity with Smooth Bound B*, is formulated. And then, the notion of truncated similarity is used to further reduce the noise. As global sensitivity considers the maximal change in the score function, it is independent of the input data. Recommender-aware sensitivity, however, considers the maximal change in the real similarity score of two users/items if any record is deleted.

$$RS(i,j) = \max_{i,j \in I} ||s(i,j) - s'(i,j)||_1.$$

To reduce noise, this sensitivity is smoothed with a bound B.

$$B(RS(i,j)) = exp(-\beta) \cdot RS(i,j)$$

Even with recommendation-aware sensitivity the naïve mechanism could still yield low prediction accuracy. A new notion, *truncated similarity*, is applied to enhance the quality of selected neighbors. Specifically, for any user/item in the list whose similarity is lower than the bound $s_k(i, \cdot) - w$, in which $s_k(i, \cdot)$ is the similarity score between user/item and the kth user/item, and w is the punishing term, then its similarity is truncated to $s_k(i, \cdot) - w$.

The candidate list is divided into two sets: C_1, which consists of users/items whose similarities are larger than the low score bound, and C_0, which includes the rest. Users/Items in C_1 are selected with exponential mechanism based on the similarity scores, while those in C_0 are considered as one candidate, whose probability is proportional to the following score.

$$\exp\left(\frac{\epsilon \cdot \hat{s}(i,j)}{4k \cdot RS(i,j)}\right)$$

This operation guarantees that no selected item in the final neighbor set has similarity score lower than the low bound $s_k(i, \cdot) - w$.

The private neighbor selection stage can be summarized as the following figure.

Algorithm 2 Private Neighbor Selection

Input: ϵ, k, w, t_i, I ,$\mathbf{s}(i)$
Output: $N_k(t_i)$

Sort the vector $\mathbf{s}(i)$;
$C_1 = [t_j | s(i,j) \geq s_k(i,j) - w, t_j \in I]$,
$C_0 = [t_j | s(i,j) < s_k(i,j) - w, t_j \in I]$,
for N=1:k **do**
  for each item t_j in $\mathbf{t}_i$ **do**
    Allocate probability as:

$$\frac{\exp\left(\frac{\epsilon \cdot \hat{s}(i,j)}{4k \cdot RS(i,j)}\right)}{\sum_{j \in C_1} \exp\left(\frac{\epsilon \cdot \hat{s}(i,j)}{4k \cdot RS(i,j)}\right) + |C_0| \cdot \exp\left(\frac{\epsilon \cdot \hat{s}(i,j)}{4k \cdot RS(i,j)}\right)}.$$

  end for
  Sample an element t from C_1 and C_0 without replacement according to probability;
  $N_k(t_i) = N_k(t_i) + t$;
end for

B) *Perturbation*. In this stage, the Laplace noise is added to the N_k (t) set following the distribution:

$$Lap\left(\frac{2 \cdot RS(i,j)}{\epsilon}\right)$$

In the paper, the perturbation noise is only added to the similarity score in the set, yet I also tried another approach: add the noise to the rating instead.

3.2 Privacy & Utility Analysis

A) *Privacy Analysis*. There are two private operations in the PNCF algorithm. For the first operation, *private neighbor selection*, each round preserves (ε/2k)-differential privacy. According to the composition theorem, this step guarantees (ε/2)-differential privacy.

The *perturbation* operation adds Laplace noise to the set N_k (t) following the distribution above. According to the definition, this step guarantees (ε/2)-differential privacy, too.

Based on the short proof above, we can claim that the whole mechanism preserves ε-differential privacy by composition theorem.

B) *Utility Analysis*. This algorithm is claimed to preserve the quality of the neighbors in two aspects: every user/item whose true similarity is greater than $s_k(i, \cdot) + w$ is selected and that whose true similarity is less than $s_k(i, \cdot) – w$ isn't selected, where true similarity refers to the similarity score without any perturbation. The detailed math proof is well explained in the original paper.

IIII. Evaluation

In the original paper they used PPC and MAE as metrices for comparing the PNCF with the traditional DP approach. For my implementation, I chose a subset of publicly available Netflix dataset [2] as my training/testing dataset. Due to the highly costly computation, I selected the first 20 records and considered only a user-based approach, which could be easily generalized to an item-based approach. To save computation time, the fixed parameters I chose are: w = 0.01, eps = 3, B = 1. I calculated the MAE metrics for similarity-perturbed approach and rating-perturbed approach, and also the MAE for naïve neighbor-based CF without any privacy-preserving operations.

The results are shown in the following table.

	K = 5	K = 7	K = 9
Similarity-perturbation PNCF	1.585	1.530	1.578
Rating-perturbation PNCF	1.528	1.592	1.723
Neighbor-based KNNCF	1.170	1.027	0.971

We can see that similarity-perturbation PNCF and rating-perturbation PNCF outperforms each other in different situation. One important discovery is that compared to rating-perturbation PNCF, similarity-perturbation PNCF has a more stable performance and similar performance to naïve neighbor-based KNN collaborative filtering.

IV. Lessons & Future Work

Through this project I have a deeper understanding of differential privacy, especially the notions such as sensitivity, exponential mechanism and Laplace mechanism. The implementation of the algorithm guides me to think thoroughly on the mathematical analysis of the whole approach and further understands the significance of this algorithm. From this point, I'd like to see how differential privacy could be applied in information retrieval process to provide personalized services while offering more powerful privacy preservation.

References

[1] Zhu, T., Li, G., Ren, Y., Zhou, W., and Xiong, P. (2013). Differential privacy for neighborhood-based collaborative ltering. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM 13, pages 752–759, New York, NY, USA. ACM.

[2] R. M. Bell, Y. Koren, and C. Volinsky. The BellKor solution to the Netflix Prize. Available from http://www.netflixprize.com, 2007.

[3] A. Narayanan and V. Shmatikov, “Robust de-anonymization of large sparse datasets,” in *Proceedings of the 2008 IEEE Symposium on Security and Privacy*, ser. SP ’08. Washington, DC, USA: IEEE Computer Society, 2008, pp. 111–125.

[4] J. A. Calandrino, A. Kilzer, A. Narayanan, E. W. Felten, and V. Shmatikov, ““you might also like: ” privacy risks of collaborative filtering,” ser. SP ’11, Washington, DC, USA, 2011, pp. 231–246.

[5] F. McSherry and I. Mironov, “Differentially private recommender systems: Building privacy into the netflix prize contenders,” ser. KDD ’09. New York, NY, USA: ACM, 2009, pp. 627–636.