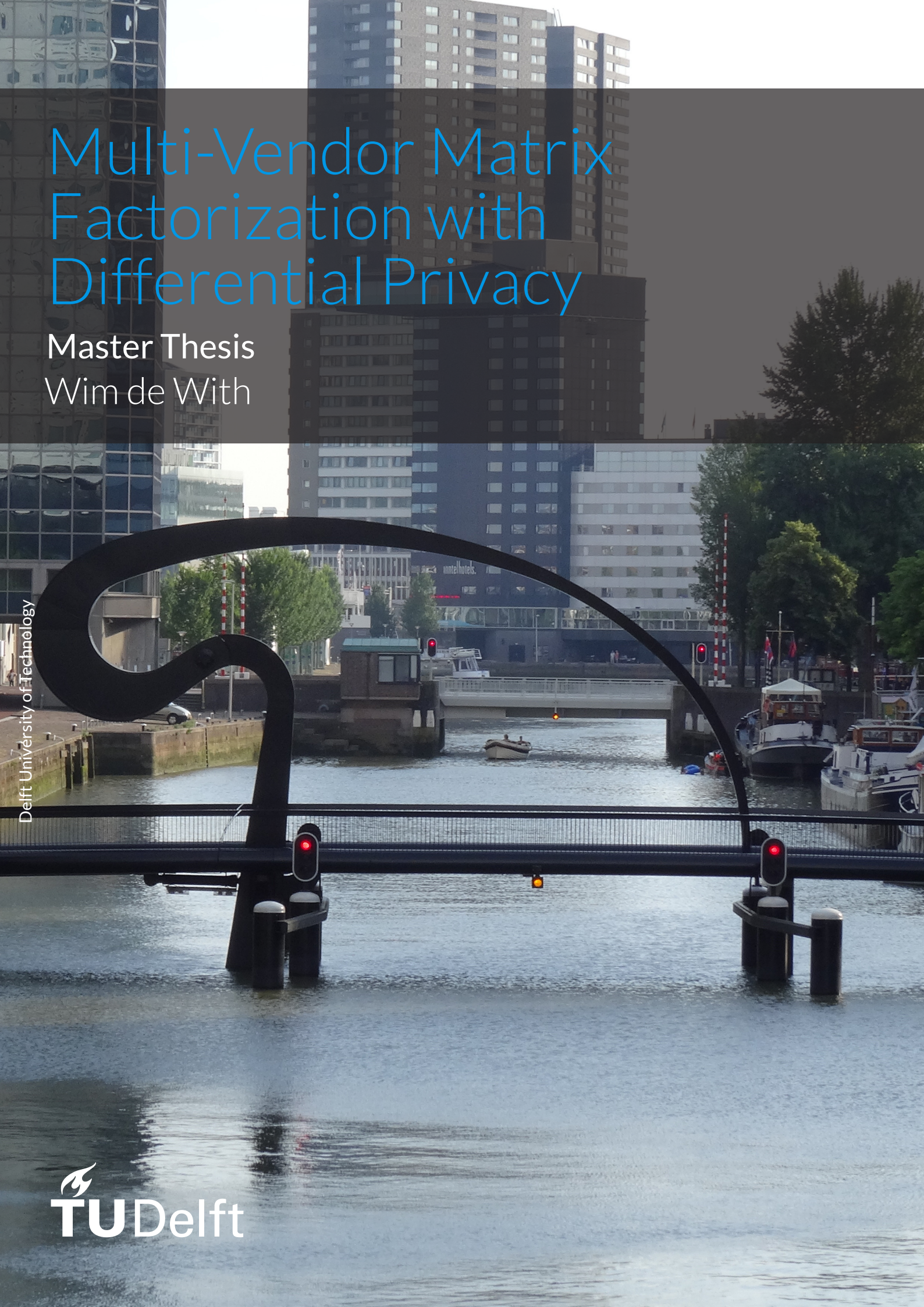


Multi-Vendor Matrix Factorization with Differential Privacy

Master Thesis

Wim de With

Delft University of Technology

TU Delft

Multi-Vendor Matrix Factorization with Differential Privacy

by

Wim de With

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Thursday, November 17, 2022 at 10:00 AM.

Student number:	4295277	
Supervisor:	dr. Zekeriya Erkin	TU Delft
Thesis committee:	dr. Julián Urbano	TU Delft

TU Delft

Preface

In front of you, you will have a copy of my master's thesis, which is the result of my work from September 2021 to November 2022. My goal for the thesis project was to do something I would not be likely to work with after finishing my master's degree. This lead me to choose a subject related to privacy-enhancing technologies, especially since privacy is very relevant for many aspects of our digital society. Privacy-enhancing technologies have very practical applications, but are based on mathematical theory, and this grey area between theory and practice really appealed to me.

When I look back to when I started, I realize I have learned so much in the past months, in terms of both technical knowledge and personal growth. It has been a journey with low lows and high highs. I'm happy and satisfied to be at the end of this journey and I hope to take all the lessons I learned with me to the future.

I want to thank dr. Zeki Erkin for being my supervisor. I wasn't always the easiest student to supervise and I really appreciated his patience with me and his words of encouragement. He motivated me to write this thesis in the form of a research paper. Writing this paper has opened my eyes on how hard it actually is to only write what is necessary, nothing more and nothing less.

Furthermore, I want to thank my brother Harm, for listening to and answering all my questions about statistics and data science. Finally, a thank you to my parents, my brothers and sisters, and my friends, whose support was indispensable for completing this project.

I hope you enjoy reading!

Wim de With
Rotterdam, November 2022

Privacy in Multi-Vendor Matrix Factorization: Any Reason to Collaborate?

Wim de With
W.F.deWith@student.tudelft.nl
Delft University of Technology
Delft, Netherlands

Zekeriya Erkin
Z.Erkin@tudelft.nl
Delft University of Technology
Delft, Netherlands

ABSTRACT

Recommender systems usually base their predictions on user-item interaction, a technique known as collaborative filtering. Vendors that utilize collaborative filtering generally exclusively use their own user-item interactions, but the accuracy of the recommendations may improve if several vendors share their data. Since user-item interaction data is typically privacy sensitive, sharing this data poses a privacy challenge for the collaborating vendors. In this work, we study the use of matrix factorization with multiple vendors under a differential privacy guarantee. Since differential privacy incurs a trade-off between privacy and utility, one obstacle is that the utility loss of the privacy-preserving measure may be greater than the utility gain of collaboration. We show that the empirical evaluation of this property in existing work is questionable, and that these works do not solve the problem. We also demonstrate that in a common experiment setup, the upper bound on the utility gain that can be achieved by collaboration is limited, which places a hard limit on the acceptable utility loss due to privacy preservation. This limit is small enough that even the utility loss in the current state of the art in differentially private matrix factorization in general exceeds it. We conclude with a number of open challenges for future work.

KEYWORDS

matrix factorization, differential privacy, federated learning

1 INTRODUCTION

In the past decades, the internet has played an increasingly important role in people's daily lives. This has led to an unprecedented increase of available information, which in turn has made it increasingly difficult for a single person to process this information to find the things they need or enjoy. Recommender systems are tools designed to manage this information and help people quickly find the items that are relevant to them, based on their preferences. One of the more commonly used methods for recommender systems is known as collaborative filtering (CF) [17]. CF uses past interactions with items of all users to find similarities in preferences, which are then used to predict the preference of a specific user. Since CF only makes use of user-item interactions, it can consider both users and items to be opaque entities, which makes it a robust technique that is insensitive to the domain in which it is used.

Parties who utilize CF, which we call vendors in this work, typically only use their own dataset of user-item interactions to generate recommendations. It has been shown that the accuracy of the recommendations may significantly improve if several vendors collaborate by sharing their user-item interaction data [8], especially if their datasets are small. However, sharing this data is complicated due to two reasons. First, user preference data is inherently privacy sensitive, and vendors require explicit consent from their users to share sensitive data with other vendors due to recent data protection laws such as the General Data Protection Regulation (GDPR). Second, vendors are usually commercial entities and will therefore be reluctant to share user preference data directly with potential competitors, as it will certainly be deemed business-sensitive information. These two complications can be avoided if the sensitive data is never shared directly, which informs our goal of finding a solution that allows vendors to collaborate so that they can improve their recommendation accuracy while preserving user privacy.

In this work, we specifically consider matrix factorization (MF) [25] as the collaborative filtering technique, since it is a popular and well-performing solution. Even though it is over a decade old, it remains competitive to this day [11, 38].

Learning an MF model with multiple vendors may not necessarily reveal direct rating information to other vendors, nonetheless, it is a known fact that model parameters [9, 12, 41], the intermediate training steps [16, 44] and the recommendations themselves [7] can expose sensitive user data. Consequently, we need stronger privacy guarantees to protect user data when vendors collaborate to train a shared model. These privacy guarantees can be formalized through *differential privacy* [13, 14].

The intuition of differential privacy is that if the contribution of a single entity to the outcome of an algorithm is indistinguishable, while trends over multiple entities can still be observed, the privacy of a single entity is preserved. The contribution of a single entity is typically disguised by perturbing the input, the intermediate state, or the output of the algorithm. However, the applied perturbations may incur utility loss in the outcome of the algorithm, so there exists a trade-off between utility and privacy.

Since our stated goal is for multiple vendors to improve their recommendation accuracy by collaborating, there is a significant constraint on the trade-off between privacy and utility. If the utility gain of collaboration is smaller than the utility loss of the privacy measure, the collaborating vendors will only impair their recommendation accuracy and are better off not collaborating at all. For this reason, it is essential that the empirical evaluation of any proposed solution shows that the accuracy of collaborating vendors improves over a well-tuned algorithm on only the vendors' local data.

Our contributions are twofold:

- We show that the existing solutions in literature for multi-vendor matrix factorization with differential privacy (DPMF)

either have experiment configurations that are highly dependent on the exact partitions of the datasets and show inconsistent results [15] or the results of the model learned by collaborating vendors can be outperformed with simple baselines or a better-tuned model on the local vendor data [29].
- We show that the maximum possible utility gain for a collaborating vendor that can be shown empirically on an existing dataset split into multiple parts is limited, which in turn limits the maximum acceptable utility loss. We also compare the utility gain of collaboration to the utility loss in the current state of the art method for single vendor DPMF.

The outline of this work is as follows. In Section 2, we show an overview of related work. In Section 3, we provide the necessary background knowledge. In Section 4, we perform experiments to give context to the reported results in existing work and we show that these results are questionable. In Section 5, we discuss the relation between single vendor DPMF and multi-vendor DPMF and we show the maximum possible utility gain on a common experiment configuration. Finally, in Section 6, we discuss challenges for future work and we conclude the work.

2 RELATED WORK

We classify related work on multi-vendor DPMF into three categories: alternative collaborative filtering techniques, single vendor DPMF, and learning a shared model in general.

Multi-vendor collaborative filtering has been studied with nearest neighborhood techniques. One approach is to apply secure multi-party computation (MPC) to allow multiple vendors to generate predictions without having to reveal both the ratings and the intermediate computation results [5, 23, 40]. This approach does not consider the privacy leakage that these predictions may incur and does not protect against the inference attacks described by Calandrino et al. [7]. The application of differential privacy to the nearest neighborhood technique to prevent these attacks was studied by Li et al. [27]. This work focuses on a scenario with two vendors where the data is shared in only one direction.

MF with differential privacy has been studied in various works [6, 10, 20, 21, 30, 35]. These methods all focus on the situation with a single vendor and only aim to protect against inference attacks based on the recommendations of the system. They do not need to protect the intermediate results of the training process and they can assume that all rating data is accessible during the training process.

The problem of learning a shared model with multiple data owners is studied under the umbrella of *federated learning* [32]. Federated learning with differential privacy for non-convex problems, such as MF, is still an open problem [22]. Current methods [33, 34] are based on adding noise to the iterative training process and composing the privacy expenditure over multiple iterations through privacy accounting, such as the *moments accountant* [1]. These methods typically require large datasets to achieve accurate models, which contrasts our scenario of vendors with smaller datasets trying to learn from each other.

Finally, to the best of our knowledge, there are two existing works in literature that propose solutions for multi-vendor DPMF. The first is a work by Ermiş and Cemgil [15], in which the authors propose a Bayesian model that exploits a natural connection between Bayesian posterior sampling and differential privacy [26, 42]. The second work is by Li et al. [29], where the authors adopt the moments accountant [1] to learn an MF model. We will consider these two works more closely in Section 4.

3 BACKGROUND

3.1 Differential Privacy

3.1.1 Standard Differential Privacy. Differential privacy [13, 14] is based on the notion of adjacent datasets, where any two datasets are called adjacent when they differ at most in a single entity. The exact definition of an entity and when they are different depends on the chosen privacy guarantee.

Definition 1 ((ϵ, δ)-differential privacy). *A randomized algorithm $\mathcal{A} : \mathcal{D} \to \mathcal{R}$ is (ϵ, δ)-differentially private if for any adjacent datasets $\mathcal{D}$ and $\mathcal{D}'$ that only differ in a single entity and for any potential outcomes $S \subseteq \mathcal{R}$, the following holds:*

$$\Pr[\mathcal{A}(\mathcal{D}) \in S] \leq e^{\epsilon} \cdot \Pr[\mathcal{A}(\mathcal{D}') \in S] + \delta.$$

If $\delta = 0$, $\mathcal{A}$ is called ϵ-differentially private.

The ϵ parameter is commonly called the privacy budget and the δ parameter is the probability that the privacy guarantee does not hold.

3.1.2 Joint Differential Privacy. The standard differential privacy definition considers all outputs of an algorithm. In the case of a recommender system, these outputs include the personalized recommendations generated for a specific user based on their past ratings, from which follows that a user can learn nothing about their past ratings from their recommendations. Consequently, recommendations can no longer be personalized, thereby defeating the purpose of a recommender system. Therefore, a slight relaxation of the differential privacy definition is needed, which allows a user to learn their own ratings from the output of the algorithm. The user's privacy is preserved as long as the user does not make their recommendations public. This relaxation is known as *joint differential privacy* and was first formalized by Kearns et al. [24]. The joint differential privacy definition is used either implicitly or explicitly in [10, 15, 21, 29, 30, 35].

Definition 2 ((ϵ, δ)-joint differential privacy). *A randomized algorithm $\mathcal{A} : \mathcal{D} \to \mathcal{R}$ is (ϵ, δ)-joint differentially private if for all users i, any adjacent datasets $\mathcal{D}$ and $\mathcal{D}'$ that only differ in entities for user i, and for any potential outcomes for all other users $S_{-i} \in \mathcal{R}$, the following holds:*

$$\Pr[\mathcal{A}(\mathcal{D}) \in S_{-i}] \leq e^{\epsilon} \cdot \Pr[\mathcal{A}(\mathcal{D}') \in S_{-i}] + \delta.$$

If $\delta = 0$, $\mathcal{A}$ is called ϵ-joint differentially private.

3.2 Matrix Factorization

3.2.1 Model. Matrix factorization for collaborative filtering [25] models the problem as follows. Let $R \in \mathbb{R}^{m \times n}$ be a matrix that represents the preferences of m users for n items. We observe a tiny fraction $\Omega \subseteq [m] \times [n]$ of R in the form of ratings, where $[n]$ denotes the set $\{1, \ldots, n\}$. We assume that R is low rank and can be factored into two smaller matrices $U \in \mathbb{R}^{m \times d}$ and $V \in \mathbb{R}^{n \times d}$

such that $R \approx UV^\intercal$, where d is the embedding dimension and $d \ll \min(m, n)$.

U and V are learned by optimizing the following model, where u_i and v_j are the i-th and j-th row vectors of U and V respectively:

$$\underset{U,V}{\operatorname{argmin}} \sum_{(i,j)\in\Omega} (R_{ij} - \langle u_i, v_j \rangle)^2 + \lambda(\|U\|_2^2 + \|V\|_2^2). \tag{1}$$

λ is a regularization parameter to prevent overfitting. This model can be extended with bias terms so that the optimization problem becomes:

$$\begin{aligned}\underset{U,V,b}{\operatorname{argmin}} \sum_{(i,j)\in\Omega} & (R_{ij} - \langle u_i, v_j \rangle - b_i^u - b_j^v - b_0)^2 \\ & +\lambda(\|U\|_2^2 + \|V\|_2^2 + b_i^u + b_j^v),\end{aligned} \tag{2}$$

where b_i^u is the user bias, b_j^v is the item bias and b_0 is the global bias.

Since solving this non-convex minimalization problem is NP-hard [18], solutions are approximated. In practice, stochastic gradient descent (SGD), Markov chain Monte Carlo (MCMC) [2, 39] and alternating least squares (ALS) [25] are popular methods that work well.

3.2.2 Differential Privacy. Mapping the joint differential privacy definition to the MF model is straightforward. A prediction of item j for user i is calculated as the dot product between the latent vectors: $\langle u_i, v_j \rangle$. Here, u_i is exclusively used to predict ratings for user i, while v_j is used for all users. Therefore, as long as differential privacy is guaranteed for the item latent vectors V, joint differential privacy is guaranteed for the entire model, and no privacy guarantees are necessary for the user latent vectors U.

There are three levels of differential privacy used in the literature. From weakest to strongest guarantee they are:

- **Rating value:** This level guarantees that the exact value of a single rating cannot be inferred from the model. Formally, let (R, Ω) and (R', Ω') be adjacent datasets with $\Omega = \Omega'$, if $\exists (i, j) \in \Omega$ for which $R_{ij} \neq R'_{ij}$. This guarantee is used in [6, 20].
- **Rating existence:** Here, the existence of a single rating is protected. Formally, let (R, Ω) and (R', Ω') be adjacent datasets if $(i, j) \in \Omega$ and $(i, j) \notin \Omega'$ or vice versa $(i, j) \notin \Omega$ and $(i, j) \in \Omega'$. This guarantee is used in [15, 29, 35, 36]. We refer to this guarantee as *rating level* privacy.
- **User existence:** The user level protection guarantees that the existence of all ratings by a single user is protected. Formally, let (R, Ω) and (R', Ω') be adjacent datasets if $(i, \cdot) \in \Omega$ and $(i, \cdot) \notin \Omega'$ or vice versa $(i, \cdot) \notin \Omega$ and $(i, \cdot) \in \Omega'$. This guarantee is used in [10, 21, 29, 30, 35]. We refer to this guarantee as *user level* privacy.

We consider the *user level* privacy guarantee to be essential for this problem because user preferences and therefore their ratings are typically very correlated. Indeed, the very purpose of collaborative filtering is to find these correlations and use them to generate recommendations. It has been shown that this aspect of collaborative filtering can be exploited to reveal user preferences [7].

3.3 Multi-Vendor Collaborative Filtering

3.3.1 Data Distribution. Collaborative filtering with multiple vendors is commonly classified into three distribution scenarios:

- **Horizontal:** In a horizontal distribution, the vendors share some or all items, but do not share users.
- **Vertical:** In a vertical distribution, vendors do not share items, but they do share some or all users.
- **Arbitrary:** The arbitrary distribution is a combination of both horizontal and vertical distribution. In this scenario, vendors share both some or all users and some or all items.

Note that *user level* privacy can only be guaranteed in the horizontal distribution scenario. The user-level privacy guarantee leads to two problems in the vertical distribution scenario. First, vendors would need to align the user identifiers across their datasets. Aligning user identifiers would already leak some private data about users, namely whether they have interacted with the other vendors or not. Second, differential privacy guarantees that one vendor cannot infer anything about a user from another vendor, *including anything that might help improve that user's recommendations.* In a horizontal distribution scenario, these problems do not occur, because only item latent features are learned collaboratively, and those features do not exclusively depend on the data of a specific user. Since the arbitrary distribution is a combination of the horizontal and vertical distributions, differential privacy can only be applied to the horizontal component, effectively turning it into a horizontal distribution.

3.3.2 Trust Model. We assume that users belonging to a specific vendor allow that vendor to use their data, but do not want their data shared beyond that boundary. Any data shared between vendors must be protected in such a way that the vendors do not need to trust each other to not learn anything about specific users beyond the users in their own dataset. This property is achieved when differential privacy is guaranteed on all shared data since there is no way for a malicious vendor to learn something that may violate user privacy in that scenario. Other malicious behaviors, such as influencing the resulting model to worsen its performance are out of the scope of this work.

3.4 Datasets and Metrics

In our experiments, we use three different variants of the MovieLens datasets [19]. The first variant is MovieLens 100k, which consists of 100,000 ratings by 943 users on 1,682 movies. The second variant is MovieLens 1M, which consists of 1,000,209 ratings by 6,040 users on 3,706 movies. The third variant is MovieLens 10M, which consists of 10,000,054 ratings by 69,878 users on 10,677 movies. The rating scale in MovieLens 100k and 1M is 1 to 5 with steps of size 1, and in MovieLens 10M, the scale is 0.5 to 5 with steps of size 0.5.

We evaluate all models by selecting a number of ratings for training as Ω_{train} and the remaining ratings for testing as Ω_{test}. Then, we use Ω_{train} to learn the model, and we report the Root Mean Squared Error (RMSE) of the predictions of the model on Ω_{test}.

$$\text{RMSE} = \sqrt{\frac{1}{|\Omega_{\text{test}}|} \sum_{(i,j)\in\Omega_{\text{test}}} (R_{ij} - \hat{R}_{ij})^2} \tag{3}$$

4 EVALUATION OF EXISTING WORK

In this section, we consider some of the baseline experiments used in existing work and we show that these experiments do not support the conclusions of the papers when compared to the results of better-tuned baselines. We consider collective matrix factorization (CMF) by Ermiş and Cemgil [15], and federated matrix factorization (FMF) by Li et al. [29]. Both of these works do not have a publicly available implementation, so we cannot reproduce their experiments. Instead, we focus on what is reported in the papers and replicate the setting of the experiment as closely as possible.

4.1 Collective Matrix Factorization

Description. Ermiş and Cemgil propose a Bayesian MF model with *rating level* differential privacy [15]. The model is learned through stochastic gradient Langevin dynamics (SGLD) [43], which is an approximation of the MCMC method. SGLD can provide differential privacy without any additional injected noise if the step size is selected small enough, although it still requires clipping the gradient norms [26, 42]. The restriction on the step size and the clipping of the gradient norms still incurs a utility loss compared to a non-private model, which the authors demonstrate in their experiments. The work only considers a horizontal distribution.

Experiment protocol. For every partition of a dataset, the authors randomly select 80% of the ratings as training data, and the remaining 20% for testing the model. In all their experiments, the authors set the embedding dimension to $d = 5$.

Unreliable comparison with related work. The authors compare CMF with the works of Hua et al. [20] and Liu et al. [30] with $\epsilon = 0.1$ on MovieLens 1M. Both of these works only consider single vendor MF, so these experiments do not involve multiple vendors. Note that the comparison with Liu et al. is not entirely valid, since Liu et al. provides *user level* differential privacy, which is much stronger than *rating level* differential privacy. We include the RMSE values for predicting the global average rating and the average rating per movie. Both the global average and the movie averages are calculated in a differentially private way, to compare using the same privacy constraints. We show both the reported results and the results of our baseline in Table 1. The first three rows are results from [15, Figure 10(b)], and the second two rows are our baselines.

Table 1: MovieLens 1M results for $\epsilon = 0.1$. Results in the first group are from [15, Figure 10(b)].

Method	RMSE
Hua et al. [20]	2.01
Liu et al. [30]	1.93
CMF [15]	1.49
DP global average	1.12
DP movie averages	0.98

It is clear that our baselines perform much better than the reported results in this setting. The comparison of the differentially private MF techniques in these experiments is very unreliable since the results do not indicate that the compared techniques are even functional recommender systems. Therefore, we cannot draw conclusions on the relative performance of the tested techniques based on these experiments.

Inconsistency between experiments. The second problem we identify is the inconsistent results between the experiments that show the impact of the privacy parameter ϵ and the impact of including multiple vendors.

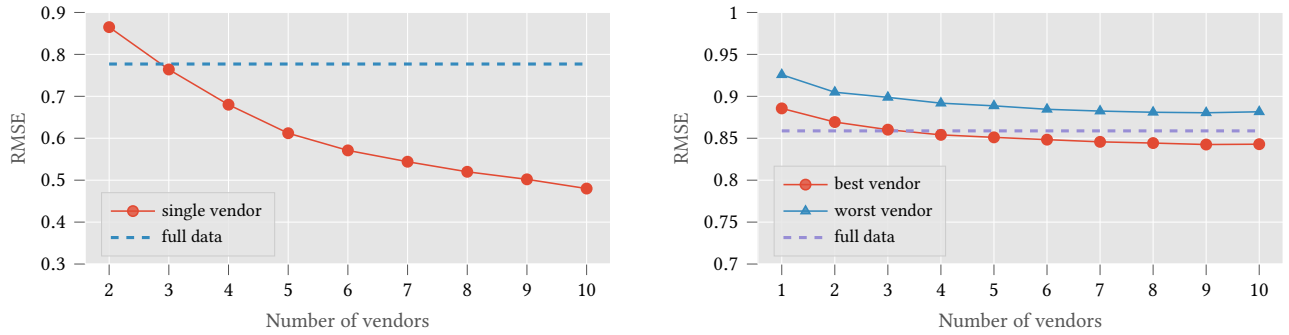
In the experiment that shows the impact of the privacy parameter ϵ, the authors divide the users in the MovieLens 1M dataset between two vendors. [1] They measure the performance of the algorithm by the RMSE for both vendors for the non-private case and for the private case with $\epsilon \in \{0.05, 0.1, 1\}$. We use the non-private RMSE values of both vendors to calculate the non-private RMSE on the full dataset. The non-private RMSE on the full dataset reflects the best-case scenario for the model performance.

For the second experiment, the authors divide the MovieLens 1M dataset by user across 10 vendors. Then, the authors report the RMSE for a single vendor while they increase the number of vendors involved in training from 2 to 10, with the privacy parameter ϵ set to 1.

We point out two problems with the second experiment. The first problem is that the RMSE of a single vendor that only uses 1/10-th of the data depends on how the data is partitioned and which partition of the data is used. The second problem is that the reported results in the second experiment are not consistent with the non-private results on the full dataset in the first experiment. The reported results for the two experiments are shown in Fig. 1a. We expect the RMSE of the single vendor to slowly converge to be close to the RMSE of the entire dataset when more vendors are included. However, the reported RMSE values for the single vendor with privacy are substantially lower than the reported RMSE values for the full dataset without privacy. This result is surprising because one would expect the RMSE values to be higher than the non-private value due to the utility loss caused by the privacy guarantee.

To show that our concerns about the reported results are justified, we perform an experiment with the same configuration without privacy. We report the results for both the single vendor with the best and the worst RMSE and the RMSE on the entire dataset in Fig. 1b. We make the following two observations when we compare our results to the results reported for CMF. First, the utility gain for the single vendor with CMF from including 6 to 10 vendors is linear and the effect of diminishing returns is minimal. In contrast, the results of our experiment indicate that there is a much stronger effect of diminishing returns on including more vendors. We argue that a degree of diminishing returns should be expected and that the reported results of the CMF technique are likely inaccurate. Second, the difference between the RMSE values between the vendors with the best and the worst results is significant. Hence, the RMSE of a single vendor has high variance, which makes the reported results in [15, Table 4] more dependent on the prior conditions than on the performance of CMF itself.

[1] The dimensions of the rating matrices reported in [15] seem to indicate that the *movies* are divided between vendors. However, since the entire work is exclusively focused on a horizontal distribution and this error does not exist in the reported dimensions of the Netflix Prize dataset, we assume it to be a typographical error.



(a) CMF **(b) Non-private replication**

Figure 1: RMSE for a single vendor in a random horizontal partition of MovieLens 1M into 10 vendors, with a varying amount of data from other vendors included.

4.2 Federated Matrix Factorization

Description. Li et al. introduce a MF model with differential privacy that is learned with SGD [29]. FMF can provide either *rating level* and *user level* privacy guarantees in both horizontal and vertical distributions. The authors provide differential privacy by bounding the latent vectors, which limits the sensitivity of a single rating, followed by using the moments accountant [1] to yield tighter privacy losses for the sequential composition of the training process. To provide *user level* privacy, the authors trim the datasets to include only a small number of ratings per user, which limits the sensitivity per user.

Experiment protocol. The authors select 90% of the ratings of every partition of a dataset for training the model, and the remaining 10% for testing. Throughout their work, the authors report the Mean Squared Error (MSE) instead of the RMSE. For consistency with the rest of our work, we convert the reported MSE values to RMSE when we compare them to our own results.[2] The authors set the embedding dimension to $d = 20$.

Vertical distribution. The authors evaluate FMF in a vertical distribution by partitioning the MovieLens 10M dataset by movie across multiple vendors and reporting the MSE values for $\epsilon \in \{0.037, 0.055, 0.086, 0.174\}$ with *rating level* privacy. They also evaluate the vertical distribution with *user level* privacy, but as noted in Section 3.3.1, vertical distribution with *user level* privacy by definition cannot lead to improved results across vendors; we will therefore ignore this configuration in our analysis. The authors include two variants of the experiment on the vertical distribution setting. In the first variant, they partition the dataset such that 18 vendors each have the ratings of all movies of a certain genre, and then they compare the results to the results of a random partition of the movies across 18 vendors. In the second variant, they compare the results for a random partition of movies across 2, 5, and 10 vendors.

We compare the reported values of both experiment variants with a baseline that recommends the average rating per movie with differential privacy. Note that calculating this average rating does not require any communication between the vendors, since every vendor knows all ratings for their own movies. We show the results in Fig. 2. The RMSE values for FMF are taken from [29, Figure 6(a), 6(b), and 7(a)].

In all experiment configurations in a vertical distribution setting for all ϵ, FMF is outperformed by recommending the average rating of a movie.

Horizontal distribution. In a horizontal distribution, the authors evaluate FMF by partitioning the MovieLens 10M dataset by user across 10 and 40 vendors. For *rating level* privacy, they report the MSE for $\epsilon \in \{0.037, 0.055, 0.086, 0.174, 0.355, 0.944\}$ and for *user level* privacy, they report the MSE for $\epsilon \in \{0.5, 1, 2, 3, 4, 5\}$.

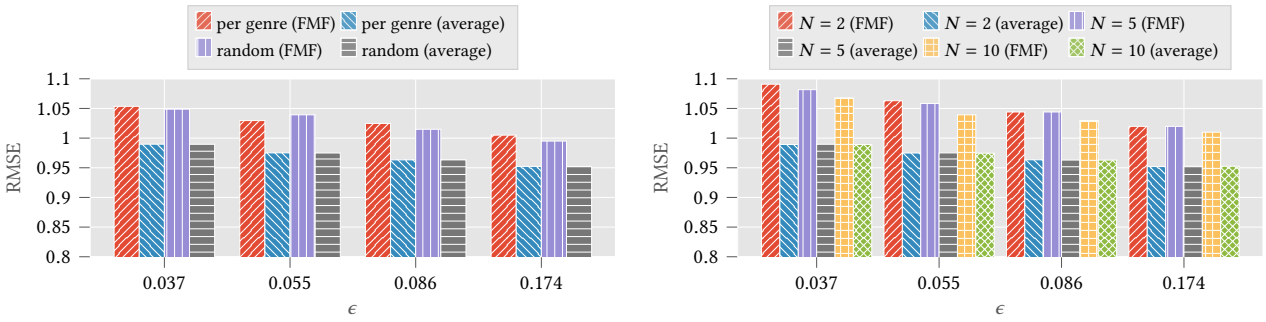
We replicate this setting and perform a baseline experiment where we learn a non-private MF model using only the local vendor data. Similar to FMF, we set the embedding dimension $d = 20$, to show that the reported results can be outperformed even with the same embedding dimension. We show the results for *rating level* privacy in Fig. 3a and for *user level* privacy in Fig. 3b. The RMSE values for FMF are taken from [29, Figure 9].

For both *rating level* and *user level* privacy, and for both 10 and 40 vendors, our baseline MF model trained on only local vendor data significantly outperforms FMF.

5 UPPER BOUNDS IN COMMON BASELINE

Splitting an existing dataset into equal parts and distributing these parts across multiple vendors appears a natural experiment configuration to evaluate the performance of a proposed solution for multi-vendor DPMF. This configuration is used in both works we evaluate in Section 4, and also in non-MF works, such as [5, 23, 40]. In this section, we evaluate this experiment configuration by showing the maximum possible utility gain the vendors can have by collaborating using an MF model. We do not consider privacy-preserving measures, since the maximum possible utility gain for a vendor occurs when the other vendors directly share data without taking the privacy requirement into account.

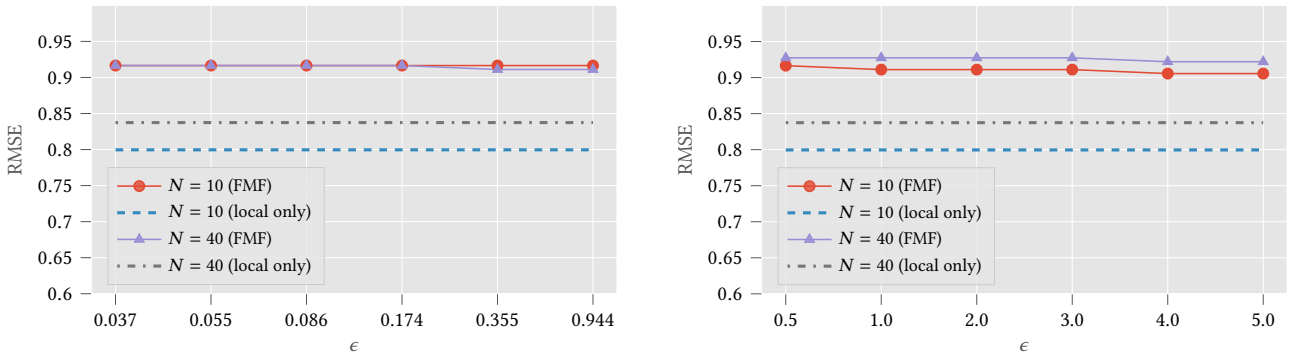
[2] $\text{RMSE} = \sqrt{\text{MSE}}$



(a) 18 vendors with per genre and random partitions

(b) N vendors with random partitions

Figure 2: FMF on MovieLens 10M in a vertical distribution scenario.



(a) N vendors with *rating level* privacy

(b) N vendors with *user level* privacy

Figure 3: FMF on MovieLens 10M in a horizontal distribution scenario.

5.1 Single Vendor vs. Multiple Vendors

In a horizontal distribution, single vendor and multi-vendor DPMF have a similar goal: ensure that the item latent vectors V are protected with differential privacy, while the user latent vectors U do not need to be. This similarity allows one to compare the model performance of single vendor and multi-vendor methods directly, as the privacy guarantee is equivalent. That is, the current state of the art method for single vendor DPMF, which is DPALS [10], reflects the current state of utility loss in DPMF in general. Multi-vendor DPMF is a strictly harder problem because of the added constraints of distributed data and protecting the intermediate training outputs, both of which likely increase the utility loss further.

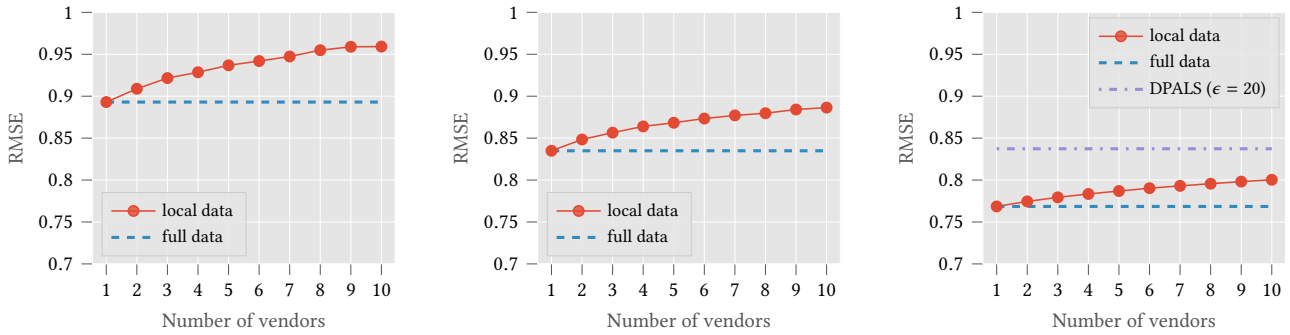
5.2 Experiment

We consider the three variants of the MovieLens dataset: 100k, 1M, and 10M, with the same experiment configuration, as follows. We randomly assign each user to 1 out of N vendors. For each vendor, we train a biased MF model on the dataset of that vendor, which we evaluate through a 10-fold cross-validation protocol. We repeat this process for N from 1 to 10. We use the result for $N = 1$ as the best possible RMSE for this model when all vendors collaborate, since in that case, the model is trained on the full dataset.

Figure 4 shows the results of our experiment. For the MovieLens 10M dataset, we include the best result for DPALS ($\epsilon = 20$). Our first observation is that the relative difference between the RMSE on the full data and the local vendor data is smaller for larger datasets. Intuitively, when vendors' datasets contain fewer users, they have the most to gain from collaborating. However, the utility loss of a model with differential privacy is typically inversely proportional to the dataset size. So, while the potential utility gain for vendors in MovieLens 100k is larger than for vendors in MovieLens 10M, the utility loss incurred by the privacy guarantee is almost certainly higher as well. The second observation we make is that the non-private model trained on only 1/10-th of MovieLens 10M still outperforms DPALS, which is trained on the full dataset. In other words, if DPALS could be applied to the multi-vendor setting as is, it would still be advantageous for the vendors to not collaborate.

6 CONCLUSIONS

In this section, we summarize issues with the methodology in existing work. We also discuss the challenges that must be overcome in any future method for multi-vendor DPMF.



(a) MovieLens 100k **(b) MovieLens 1M** **(c) MovieLens 10M**

Figure 4: Performance difference between full data and local data access for MovieLens datasets.

6.1 Methodological Issues in Existing Work

Empirical evaluation plays an important role in recommender systems research. Reporting metrics on well-known datasets allows researchers to compare new methods to established baselines to show improvements. However, empirical results in existing research in general recommender systems are often questionable, which has been demonstrated by Rendle et al. [38] and Dacrema et al. [11]. Questionable empirical results make it difficult to judge the performance of a newly proposed method relative to the existing baselines. Nonetheless, the method may still generate new insights and have value beyond the performance on specific datasets.

Our results in Section 4 indicate that problems with empirical results occur in existing work on multi-vendor DPMF as well. We argue that the consequences of unreliable empirical results in research into multi-vendor DPMF are even worse than those for general recommender systems research. First, for any method, the empirical results are used to justify the accuracy improvements vendors make when they collaborate. If the baseline results of vendors using their local data are inaccurate, any accuracy improvements are deceptive and cannot be used to determine whether collaboration is actually advantageous for vendors. Second, the impact of applying differential privacy to an existing model such as MF is not intuitive. It is practically impossible to estimate the effects of adding noise and limiting user sensitivity on the performance of the model. Therefore, without reliable empirical results, researchers cannot judge the proposed method in a meaningful way.

Dacrema et al. offer various guidelines and best practices for research in recommender systems to increase the reliability of the empirical evaluations [11, Section 5.4]. These suggestions apply to multi-vendor DPMF as well, so any future work should take them into account.

6.2 Challenges in Future Work

Collaborative filtering offers a unique challenge for differential privacy because users typically only interact with a small subset of all items, but *user level* differential privacy guarantees that even a user who rates every single item against the model prediction will have their privacy preserved. Since differential privacy considers the worst-case scenario of privacy leakage, these extreme users increase the required magnitude of the noise for all users. To prevent large magnitudes of noise, it is common in existing work to limit the number of ratings a single user can contribute to the model [10, 29, 30]. As a consequence, this limit also restricts the number of ratings a single vendor can contribute to the model when training a model collaboratively. That is, by collaborating with other vendors, a vendor can no longer utilize their local data to the fullest extent to train the model, which further limits the potential utility gain of collaboration, even without the addition of noise. It is unknown whether this problem is solvable, as *user level* differential privacy is still relatively poorly understood in machine learning [3], especially federated learning [22].

In practice, the number of observed ratings in a recommender system is not uniformly distributed over the items [31]. Instead, the distribution is skewed towards a small number of popular items (short head) and the number of items with a small number of ratings is much larger (long tail). It has been shown that the utility loss caused by differential privacy affects underrepresented classes in a dataset disproportionately [4]. Therefore, the recommendation accuracy for the items in the long tail will suffer more when differential privacy is applied to MF. Unfortunately, collaborating vendors in a horizontal distribution will have the most to gain from improving recommendations precisely in the long tail, as they are more likely to already have a sufficient number of ratings in the short head. Future work should consider this issue and potentially include experiments to show the impact of differential privacy on the long tail.

As we demonstrate in Section 5, even the current state of the art for single vendor DPMF incurs more utility loss than can be gained through collaboration. This result suggests that more effort is still needed to reduce the utility loss in single vendor DPMF before any solution for multi-vendor setting can be proposed. Nonetheless, we stress that the experiment in Section 5 is only a single experiment on three datasets. The rating prediction task is only one aspect of the performance of a recommender system. Future work may consider more specific scenarios with different metrics in which the

maximum possible utility gain is much higher than in our experiment. Still, it is essential that these specific scenarios are well motivated beforehand.

REFERENCES

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep Learning with Differential Privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS '16)*. Association for Computing Machinery, New York, NY, USA, 308–318. https://doi.org/10.1145/2976749.2978318

[2] Sungjin Ahn, Anoop Korattikara, Nathan Liu, Suju Rajan, and Max Welling. 2015. Large-Scale Distributed Bayesian Matrix Factorization Using Stochastic Gradient MCMC. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '15)*. Association for Computing Machinery, New York, NY, USA, 9–18. https://doi.org/10.1145/2783258.2783373

[3] Kareem Amin, Alex Kulesza, Andres Munoz, and Sergei Vassilvtiskii. 2019. Bounding User Contributions: A Bias-Variance Trade-off in Differential Privacy. In *Proceedings of the 36th International Conference on Machine Learning*. PMLR, 263–271.

[4] Eugene Bagdasaryan, Omid Poursaeed, and Vitaly Shmatikov. 2019. Differential Privacy Has Disparate Impact on Model Accuracy. In *Advances in Neural Information Processing Systems*, Vol. 32. Curran Associates, Inc.

[5] Alon Ben Horin and Tamir Tassa. 2021. Privacy Preserving Collaborative Filtering by Distributed Mediation. In *Fifteenth ACM Conference on Recommender Systems (RecSys '21)*. Association for Computing Machinery, New York, NY, USA, 332–341. https://doi.org/10.1145/3460231.3474251

[6] Arnaud Berlioz, Arik Friedman, Mohamed Ali Kaafar, Roksana Boreli, and Shlomo Berkovsky. 2015. Applying Differential Privacy to Matrix Factorization. In *Proceedings of the 9th ACM Conference on Recommender Systems (RecSys '15)*. Association for Computing Machinery, New York, NY, USA, 107–114. https://doi.org/10.1145/2792838.2800173

[7] Joseph A. Calandrino, Ann Kilzer, Arvind Narayanan, Edward W. Felten, and Vitaly Shmatikov. 2011. "You Might Also Like:" Privacy Risks of Collaborative Filtering. In *2011 IEEE Symposium on Security and Privacy*. 231–246. https://doi.org/10.1109/SP.2011.40

[8] Iván Cantador, Ignacio Fernández-Tobías, Shlomo Berkovsky, and Paolo Cremonesi. 2015. Cross-Domain Recommender Systems. In *Recommender Systems Handbook*, Francesco Ricci, Lior Rokach, and Bracha Shapira (Eds.). Springer US, Boston, MA, 919–959. https://doi.org/10.1007/978-1-4899-7637-6_27

[9] Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. 2021. Extracting Training Data from Large Language Models. In *30th USENIX Security Symposium (USENIX Security 21)*. 2633–2650.

[10] Steve Chien, Prateek Jain, Walid Krichene, Steffen Rendle, Shuang Song, Abhradeep Thakurta, and Li Zhang. 2021. Private Alternating Least Squares: Practical Private Matrix Completion with Tighter Rates. In *Proceedings of the 38th International Conference on Machine Learning*. PMLR, 1877–1887.

[11] Maurizio Ferrari Dacrema, Simone Boglio, Paolo Cremonesi, and Dietmar Jannach. 2021. A Troubling Analysis of Reproducibility and Progress in Recommender Systems Research. *ACM Transactions on Information Systems* 39, 2 (Jan. 2021), 20:1–20:49. https://doi.org/10.1145/3434185

[12] Irit Dinur and Kobbi Nissim. 2003. Revealing Information While Preserving Privacy. In *Proceedings of the Twenty-Second ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS '03)*. Association for Computing Machinery, New York, NY, USA, 202–210. https://doi.org/10.1145/773153.773173

[13] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. 2006. Our Data, Ourselves: Privacy Via Distributed Noise Generation. In *Advances in Cryptology - EUROCRYPT 2006 (Lecture Notes in Computer Science)*, Serge Vaudenay (Ed.). Springer, Berlin, Heidelberg, 486–503. https://doi.org/10.1007/11761679_29

[14] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating Noise to Sensitivity in Private Data Analysis. In *Theory of Cryptography (Lecture Notes in Computer Science)*, Shai Halevi and Tal Rabin (Eds.). Springer, Berlin, Heidelberg, 265–284. https://doi.org/10.1007/11681878_14

[15] Beyza Ermiş and A. Taylan Cemgil. 2020. Data Sharing via Differentially Private Coupled Matrix Factorization. *ACM Transactions on Knowledge Discovery from Data* 14, 3 (May 2020), 28:1–28:27. https://doi.org/10.1145/3372408

[16] Dashan Gao, Ben Tan, Ce Ju, Vincent W. Zheng, and Qiang Yang. 2020. Privacy Threats Against Federated Matrix Factorization. *arXiv:2007.01587 [cs, stat]* (July 2020). arXiv:2007.01587 [cs, stat]

[17] David Goldberg, David Nichols, Brian M. Oki, and Douglas Terry. 1992. Using Collaborative Filtering to Weave an Information Tapestry. *Commun. ACM* 35, 12 (Dec. 1992), 61–70. https://doi.org/10.1145/138859.138867

[18] Moritz Hardt, Raghu Meka, Prasad Raghavendra, and Benjamin Weitz. 2014. Computational Limits for Matrix Completion. In *Proceedings of The 27th Conference on Learning Theory*. PMLR, 703–725.

[19] F. Maxwell Harper and Joseph A. Konstan. 2015. The MovieLens Datasets: History and Context. *ACM Transactions on Interactive Intelligent Systems* 5, 4 (Dec. 2015), 19:1–19:19. https://doi.org/10.1145/2827872

[20] Jingyu Hua, Chang Xia, and Sheng Zhong. 2015. Differentially Private Matrix Factorization. In *Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI'15)*. AAAI Press, Buenos Aires, Argentina, 1763–1770.

[21] Prateek Jain, Om Dipakbhai Thakkar, and Abhradeep Thakurta. 2018. Differentially Private Matrix Completion Revisited. In *Proceedings of the 35th International Conference on Machine Learning*. PMLR, 2215–2224.

[22] Peter Kairouz, H. Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, Rafael G. L. D'Oliveira, Hubert Eichner, Salim El Rouayheb, David Evans, Josh Gardner, Zachary Garrett, Adrià Gascón, Badih Ghazi, Phillip B. Gibbons, Marco Gruteser, Zaid Harchaoui, Chaoyang He, Lie He, Zhouyuan Huo, Ben Hutchinson, Justin Hsu, Martin Jaggi, Tara Javidi, Gauri Joshi, Mikhail Khodak, Jakub Konečný, Aleksandra Korolova, Farinaz Koushanfar, Sanmi Koyejo, Tancrède Lepoint, Yang Liu, Prateek Mittal, Mehryar Mohri, Richard Nock, Ayfer Özgür, Rasmus Pagh, Mariana Raykova, Hang Qi, Daniel Ramage, Ramesh Raskar, Dawn Song, Weikang Song, Sebastian U. Stich, Ziteng Sun, Ananda Theertha Suresh, Florian Tramèr, Praneeth Vepakomma, Jianyu Wang, Li Xiong, Zheng Xu, Qiang Yang, Felix X. Yu, Han Yu, and Sen Zhao. 2021. Advances and Open Problems in Federated Learning. arXiv:1912.04977 [cs, stat]

[23] Harmanjeet Kaur, Neeraj Kumar, and Mohammad S. Obaidat. 2019. Multi-Party Secure Collaborative Filtering for Recommendation Generation. In *2019 IEEE Global Communications Conference (GLOBECOM)*. 1–6. https://doi.org/10.1109/GLOBECOM38437.2019.9013193

[24] Michael Kearns, Mallesh Pai, Aaron Roth, and Jonathan Ullman. 2014. Mechanism Design in Large Games: Incentives and Privacy. In *Proceedings of the 5th Conference on Innovations in Theoretical Computer Science (ITCS '14)*. Association for Computing Machinery, New York, NY, USA, 403–410. https://doi.org/10.1145/2554797.2554834

[25] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix Factorization Techniques for Recommender Systems. *Computer* 42, 8 (Aug. 2009), 30–37. https://doi.org/10.1109/MC.2009.263

[26] Bai Li, Changyou Chen, Hao Liu, and Lawrence Carin. 2019. On Connecting Stochastic Gradient MCMC and Differential Privacy. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*. PMLR, 557–566.

[27] Jianqiang Li, Ji-Jiang Yang, Yu Zhao, Bo Liu, Mengchu Zhou, Jing Bi, and Qing Wang. 2017. Enforcing Differential Privacy for Shared Collaborative Filtering. *IEEE Access* 5 (2017), 35–49. https://doi.org/10.1109/ACCESS.2016.2600258

[28] Ninghui Li, Min Lyu, Dong Su, and Weining Yang. 2016. *Differential Privacy: From Theory to Practice*. Morgan & Claypool Publishers.

[29] Zitao Li, Bolin Ding, Ce Zhang, Ninghui Li, and Jingren Zhou. 2021. Federated Matrix Factorization with Privacy Guarantee. *Proceedings of the VLDB Endowment* 15, 4 (Dec. 2021), 900–913. https://doi.org/10.14778/3503585.3503598

[30] Ziqi Liu, Yu-Xiang Wang, and Alexander Smola. 2015. Fast Differentially Private Matrix Factorization. In *Proceedings of the 9th ACM Conference on Recommender Systems (RecSys '15)*. Association for Computing Machinery, New York, NY, USA, 171–178. https://doi.org/10.1145/2792838.2800191

[31] Benjamin Marlin, Richard S. Zemel, Sam Roweis, and Malcolm Slaney. 2012. Collaborative Filtering and the Missing at Random Assumption. https://doi.org/10.48550/arXiv.1206.5267 arXiv:1206.5267 [cs, stat]

[32] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*. PMLR, 1273–1282.

[33] H. Brendan McMahan, Galen Andrew, Ulfar Erlingsson, Steve Chien, Ilya Mironov, Nicolas Papernot, and Peter Kairouz. 2019. A General Approach to Adding Differential Privacy to Iterative Training Procedures. https://doi.org/10.48550/arXiv.1812.06210 arXiv:1812.06210 [cs, stat]

[34] H. Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. 2018. Learning Differentially Private Recurrent Language Models. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

[35] Frank McSherry and Ilya Mironov. 2009. Differentially 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 (KDD '09)*. Association for Computing Machinery, New York, NY, USA, 627–636. https://doi.org/10.1145/1557019.1557090

[36] Xuying Meng, Suhang Wang, Kai Shu, Jundong Li, Bo Chen, Huan Liu, and Yujun Zhang. 2018. Personalized Privacy-Preserving Social Recommendation. *Proceedings of the AAAI Conference on Artificial Intelligence* 32, 1 (April 2018). https://doi.org/10.1609/aaai.v32i1.11714

[37] Steffen Rendle. 2012. Factorization Machines with libFM. *ACM Transactions on Intelligent Systems and Technology* 3, 3 (May 2012), 57:1–57:22. https://doi.org/10.1145/2168752.2168771
[38] Steffen Rendle, Li Zhang, and Yehuda Koren. 2019. On the Difficulty of Evaluating Baselines: A Study on Recommender Systems. *arXiv:1905.01395 [cs]* (May 2019). arXiv:1905.01395 [cs]
[39] Ruslan Salakhutdinov and Andriy Mnih. 2008. Bayesian Probabilistic Matrix Factorization Using Markov Chain Monte Carlo. In *Proceedings of the 25th International Conference on Machine Learning (ICML '08)*. Association for Computing Machinery, New York, NY, USA, 880–887. https://doi.org/10.1145/1390156.1390267
[40] Erez Shmueli and Tamir Tassa. 2020. Mediated Secure Multi-Party Protocols for Collaborative Filtering. *ACM Transactions on Intelligent Systems and Technology* 11, 2 (Feb. 2020), 15:1–15:25. https://doi.org/10.1145/3375402
[41] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership Inference Attacks Against Machine Learning Models. In *2017 IEEE Symposium on Security and Privacy (SP)*. 3–18. https://doi.org/10.1109/SP.2017.41
[42] Yu-Xiang Wang, Stephen Fienberg, and Alex Smola. 2015. Privacy for Free: Posterior Sampling and Stochastic Gradient Monte Carlo. In *Proceedings of the 32nd International Conference on Machine Learning*. PMLR, 2493–2502.
[43] Max Welling and Yee Whye Teh. 2011. Bayesian Learning via Stochastic Gradient Langevin Dynamics. In *Proceedings of the 28th International Conference on International Conference on Machine Learning (ICML'11)*. Omnipress, Madison, WI, USA, 681–688.
[44] Ligeng Zhu, Zhijian Liu, and Song Han. 2019. Deep Leakage from Gradients. In *Advances in Neural Information Processing Systems*, Vol. 32. Curran Associates, Inc.

A BASELINES

A.1 Averages

Algorithm 1 guarantees ϵ-differential privacy on calculating the average of data where every individual entry is bounded. It is taken from [28, Algorithm 2.4]. Laplace(b) in Algorithm 1 refers to a random sample of the Laplace distribution with mean 0 and scale b.

To calculate item averages (Algorithm 2), we first calculate the global average rating to fill in for any item for which no ratings exist in Ω_{train}. Through sequential composition, the calculation of the average rating of a single item is ϵ-differentially private. Since every rating is only involved in the calculation of a single average item rating, as long as the global average rating is calculated only once, we can use parallel composition to guarantee ϵ-differential privacy on all average item ratings.

To calculate the global average in Table 1, we use Algorithm 1 with all ratings in the dataset. We use the same algorithm in Table 1 and Fig. 2 for the average item ratings.

Algorithm 1 Differentially Private Average ([28, Algorithm 2.4])

procedure DPAvg(X, $x_{\min}$, $x_{\max}$, ϵ)
  $s \leftarrow \sum_{x \in X} x - |X| \cdot \frac{1}{2}(x_{\min}+x_{\max}) + \text{Laplace}(\frac{1}{\epsilon}(x_{\max}-x_{\min}))$
  $c \leftarrow |X| + \text{Laplace}(\frac{2}{\epsilon})$
  if $c \leq 1$ **then**
    return $\frac{1}{2}(x_{\min} + x_{\max})$
  else
    return $\frac{s}{c} + \frac{1}{2}(x_{\min} + x_{\max})$
  end if
end procedure

A.2 Matrix Factorization

We use an Bayesian model learned by a Gibbs sampling algorithm for all our matrix factorization experiments. This algorithm has the major advantage of having a minimal set of hyperparameters. We use libFM [37] as the implementation of this algorithm.

Algorithm 2 Differentially Private Average Item Rating

Require: $r_{\min}$ and $r_{\max}$ are the minimum and maximum rating respectively
procedure DPItemAvg(R, Ω_{train}, ϵ)
  $X \leftarrow \{R_{ij} \mid (i,j) \in \Omega_{\text{train}}\}$
  $\tilde{g} \leftarrow$ DPAvg($X, r_{\min}, r_{\max}, 0.01\epsilon$) ▷ global average
  for $k \in [m]$ **do**
    $X_k \leftarrow \{R_{ij} \mid (i,j) \in \Omega_{\text{train}} \wedge j = k\}$
    if $X_k = \emptyset$ **then** ▷ ensure that every item has at least one rating
      $X_k \leftarrow \{\tilde{g}\}$
    end if
    $\tilde{r}_k \leftarrow$ DPAvg($X_k, r_{\min}, r_{\max}, 0.99\epsilon$) ▷ item average
    Clamp $\tilde{r}_k$ to $[r_{\min}, r_{\max}]$
  end for
  return $\tilde{r}$
end procedure

Table 2 shows all the hyperparameters for the experiments in our work. σ is the variance of the random initialization of the latent vectors, d is the embedding dimension and T is the number of iterations.

Table 2: Hyperparameter choices

Experiment	σ	d	T
Fig. 1b	0.1	5	256
Fig. 3	0.1	20	256
Fig. 4a	0.1	16	100
Fig. 4b	0.1	24	100
Fig. 4c	0.1	32	100