PrivateRec: Differentially Private Model Training and Online Serving for Federated News Recommendation

Ruixuan Liu[1], Yanlin Wang2, Yang Cao[3], Lingjuan Lyu[4], Weike Pan[5], Yun Chen[6], Hong Chen[1]
[1]Renmin University of China, [2]Microsoft Research Asia, [3]Kyoto University, [4]Sony, [5]Shenzhen University, [6]Shanghai University of Finance and Economics
{ruixuan.liu,chong}@ruc.edu.cn,yanlwang@microsoft.com
yang@i.kyoto-u.ac.jp,lingjuan.lv@sony.com,panweike@szu.edu.cn,yunchen@sufe.edu.cn

arXiv:2204.08146v2 [cs.IR] 14 Dec 2022

ABSTRACT

Collecting and training over sensitive personal data raise severe privacy concerns in personalized recommendation systems, and federated learning can potentially alleviate the problem by training models over decentralized user data. However, a theoretically private solution in both the training and serving stages of federated recommendation is essential but still lacking. Furthermore, naively applying differential privacy (DP) to the two stages in federated recommendation would fail to achieve a satisfactory trade-off between privacy and utility due to the high-dimensional characteristics of model gradients and hidden representations. In this work, we propose a federated news recommendation method for achieving a better utility in model training and online serving under a DP guarantee. We first clarify the DP definition over behavior data for each round in the life-circle of federated recommendation systems. Next, we propose a privacy-preserving online serving mechanism under this definition based on the idea of decomposing user embeddings with public basic vectors and perturbing the lower-dimensional combination coefficients. We apply a random behavior padding mechanism to reduce the required noise intensity for better utility. Besides, we design a federated recommendation model training method, which can generate effective and public basic vectors for serving while providing DP for training participants. We avoid the dimension-dependent noise for large models via label permutation and differentially private attention modules. Experiments on real-world news recommendation datasets validate that our method achieves superior utility under a DP guarantee in both training and serving of federated news recommendations.

KEYWORDS

Federated Learning, Privacy-preserving, Recommender System

ACM Reference Format:
Ruixuan Liu[1], Yanlin Wang2, Yang Cao[3], Lingjuan Lyu[4], Weike Pan[5], Yun Chen[6], Hong Chen[1]. 2022. PrivateRec: Differentially Private Model Training and Online Serving for Federated News Recommendation. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
Conference'17, July 2017, Washington, DC, USA
© 2022 Copyright held by the owner/author(s).
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Nowadays, news recommendation systems are indispensable for users to filter a large amount of news articles and relieve the information overload in the web applications. To model personalized interests for accurate recommendations, most news recommendation methods [5, 11, 18, 35, 42, 49] depend on training over user behaviors such as historical clicks. During recommendation services, personal behavior data are also necessary for the model to rank candidate news articles that fit user interests. However, the trust and privacy is also an essential concern on the wide web. The behavior data contain abundant personal identifiable information and are risky to be exchanged directly. Thus, an ever-increasing privacy concern arises in the whole society for the current recommendation system. Many law regulations (e.g., GDPR[1], CCPA[2]) are adopted to limit the transportation and exploitation of personal data. As a result, collecting behavior data for training and serving in recommendation systems may be forbidden in the near future, making an effective recommendation service challenging.

Federated learning (FL) [25] is proposed as a new paradigm to train models on scattered data, which mitigates the privacy concern because personal data are only kept on users' devices. Federated recommendation systems (FRS) [9, 23, 28, 32, 36, 38] enable multiple users and a server collaboratively train recommendation models by exchanging model gradients instead of personal data. However, two critical challenges hinder the implementation of federated recommendations.

First, an effective solution with theoretical privacy protection throughout the life-circle of FRS is essential but still lacking. Existing privacy attacks indicate the possibility of inferring private information in federated training by observing local model updates or even stealing more private data by manipulating global model parameters [29, 50]. Also, private data can be leaked in the prediction stage [48]. Similarly, potential adversaries [8, 14] in FRS can infer users' clicking histories and track their personal intents. However, existing FRS methods do not consider the theoretical privacy guarantee [3, 22, 23, 28, 32, 33] or only consider the privacy in training stage [21, 36, 38]. The connection between privacy-preserving training and serving has hitherto received scant attention.

Second, a utility hurdle arises in the attempt of providing a theoretical privacy guarantee for federated news recommendations. As a golden standard of privacy criterion, local differential privacy (LDP) [13] can be applied in federated training by perturbing local model updates before sending to the server[32, 33]. Since the noise magnitude is dimension-dependent and news recommendation models

[1]https://gdpr-info.eu
[2]https://oag.ca.gov/privacy/ccpa

are typically large deep models, the utility drop is significant. In addition, for the FRS serving with millions of candidate news, it is impractical to conduct local serving [10, 17] where each user is required to store all candidate items for local ranking due to enormous communication and memory costs. Thus, we follow a more practical way of online serving [33], where user embeddings encoded with local historical clicks are sent to the server for recommendations. Typically, a larger-dimensional user embedding is more potent to describe user interests, but the utility can be ruined with the dimension-dependent noise for a DP guarantee [13]. Therefore, it is challenging to provide a satisfactory utility under a reasonable privacy for both training and serving.

To solve above challenges, we propose a federated news recommendation framework *PrivateRec* with a theoretical privacy guarantee and a decent recommendation performance. First of all, we rethink the essential question of how differential privacy [13] is defined in training and serving for federated news recommendation. Then, we preserve the privacy of historical clicks by injecting DP noise when encoding user embeddings during federated training and serving. Additionally, we permute the true labels in the training data for preserving users' true responses to candidate items, which results in the overall privacy guarantee of *PrivateRec*. To avoid the severe utility drop caused by dimension-dependent noise in training[32, 33], we decompose a user embedding into low-dimensional attentions before perturbation in both training and serving. We further reduce the noise magnitude with the privacy amplification effect of the random padding in the user encoder. Experiments validate a decent utility-privacy trade-off of our method and emphasize the necessity of considering the privacy-preserving serving utility when design federated recommendation solutions.

To summarize, the main contributions of this paper are three folds: (1) We propose a novel and unified framework *PrivateRec* for better utility in training and serving of a theoretically private federated news recommendation system. (2) We design privacy-preserving mechanisms for federated training and serving with a decent recommendation performance. *PrivateRec* is based on the idea of decomposing the user embedding into a lower-dimensional attention vector to avoid the utility drop by the dimension curse for both training and inference. Then we design a random padding mechanism to further reduce injected noise and improve the performance. (3) Extensive experiments and analysis on real-world datasets validate the significant utility improvement of training and serving in *PrivateRec*. The utility lower bound under an extremely strong privacy is improved by 5-11% on AUC.

2 RELATED WORK

Personalized news recommendation is important for intelligent online news service. Many deep learning-based recommendation models have been proposed for this task [31, 41, 42, 44]. Generally, their frameworks include three core components, i.e., news model, user model, and click prediction module. The news model can learn news representations from news content with CNN [41] or pre-trained language models [45]. The user model aims to model user interest from their clicked news with components as GRU network [31], personalized attention [42] and multi-head self-attention [44]. Last, the clicking prediction module estimates the matching score between a news embedding and a user embedding, which can be implemented with the dot product [4], the outer product [16] and the dense network [39]. However, these works rely on centralized user data for model training and serving and violates privacy [32].

Federated recommendation systems (FRS) [3, 17, 23, 32, 43] are adapted from FedAvg [25] to avoid centralized data collection. Since exchanging gradients still poses a threat to user privacy [8, 50], privacy-preserving FRS methods are necessary. Homomorphic encryption can be used to encrypt local model update [7, 8], but it largely increases the communication or computation cost for deep models. Anonymity [23] or perturbation [32] can mitigate the privacy risks, but do not provide theoretical privacy guarantee as DP. A bandit framework [20] for FRS apply DP to output a private sum reward, but cannot be generalized to recommendation model training. A comprehensive study [21] discussed vertical, horizontal and local FRS with per-rating and per-user privacy DP, but only limits to matrix factorization model. It should be noted that existing works for FRS are designed under different DP definitions, such as bounding the influence of a local App [17], the meta feature for an item [10], or the indistinguishment for any two users [30, 36]. There also multiple DP definitions applied in general FL, such as user-level [26], sample-level [1], local-user-level [30] and parameter-level [37]. As far as we know, there is no investigation for privacy definitions in deep recommendation models.

Moreover, only a few works [10, 17, 33] discuss the model serving in federated learning. A unified framework [33] is proposed for training and serving with both recall and ranking models, while our work is built for ranking models. Even the noise is injected on gradients for training and on user interests for serving, there is no theoretical privacy definition. Different from ours, they perturb the gradient vectors in training which result in a model-size-dependent noise and cannot reach a descent utility-privacy trade-off if we formulate their noise as to a DP level. In addition, each user in Uni-FedRec [33] sends multiple interest vectors, which enlarges the privacy risks by multiple times given a same perturbation level for each vector. In other works [10, 17], the server sends all items to a user to perform local serving with the local data on multiple platforms, which does not violate privacy but is impractical for the high communication and memory cost. In this work, we fill the gap by revisiting various DP definitions and formulating a privacy notion for every communication throughout the life-circle of federated recommendation.

3 METHODOLOGY

3.1 Threat Model and Privacy Definition

There are two parties in *PrivateRec*: (1) $\mathcal{S}$, the server who organizes the model training and returns the recommended results in serving. Typically, $\mathcal{S}$ is an honest-but-curious server that follows the standard workflow but is curious about user's behavior data for more commercial interests. (2) $\mathcal{U} = \mathcal{U}_t \cup \mathcal{U}_s$, the set of users $\mathcal{U}_t$ who participant in federated training and the set of users $\mathcal{U}_s$ who query recommendation services. Specifically, federated news recommendation requires local historical clicking log $N_h = \{n_1, \cdots, n_H\}$ and clicked candidate clicking log $N_c = \{n_1, \cdots, n_C\}$ for training, and need N_h to encode user interests for serving. The information that any potential third-party adversary can access is no more than $\mathcal{S}$,

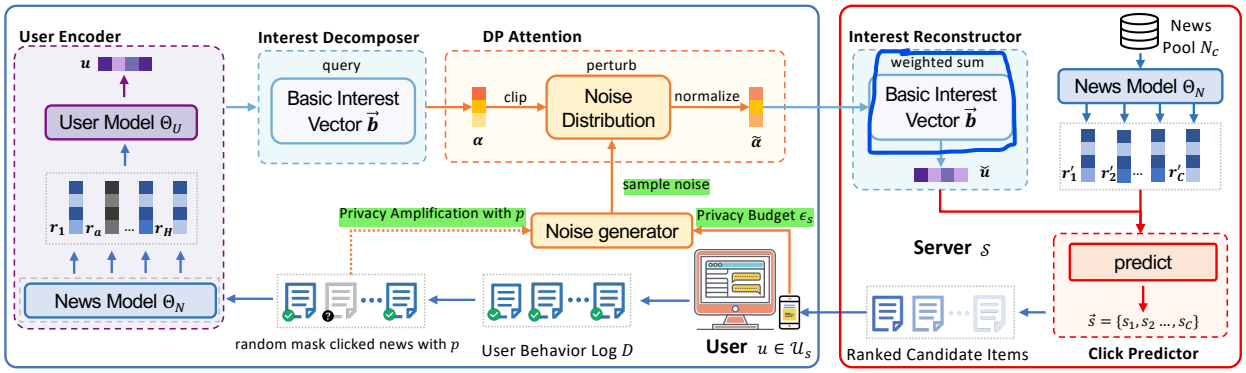

Figure 1: Privacy-preserving online serving in *PrivateRec*.

so *PrivateRec* aims to protect local N_h and N_c for each user against an untrusted server $\mathcal{S}$.

We revisit standard DP definitions in FL as follows: (1) User-level [26]: the adversary cannot identify the existence of a user by observing the distributed model parameters, which requires the server to be trusted. Complementing it with SecureAgg [7] can defend untrusted servers but requires a non-negligible cost for training and cannot be applied in the serving stage. (2) Sample-level [1]: when DPSGD [1] is conducted in local optimization, the adversary cannot identify the existence of a training sample in the local dataset by observing the uploaded updates. This DP definition requires the assumption that each sample is independently sampled from a distribution. However, the training data in news recommendation models typically include samples with partially overlapping historical clicks and do not satisfy this assumption. Also, DPSGD cannot be applied in model serving. (3) Local-user-level [30]: any party cannot distinguish the local updates from any two different users. Since effective news recommendation models in the industry are typically deep models [45], applying local DP on local updates suffers from the notorious utility drop issue. (4) Parameter-level [37]: any party that observes a value of local update cannot identify the existence of a local training sample, which is a relaxed definition of sample-level DP and faces the same problems for training and serving.

In *PrivateRec*, we define the differential privacy as the plausibility of whether an item is clicked by a user against $\mathcal{S}$, which is derived from the central model for DP in a local view as Definition 1. This fine-grained DP definition would bring advantages of: (1) unified privacy standard across training and serving, which enable users to quantify the overall privacy cost in a federated recommendation service with a clear notion. (2) support for personalized privacy settings to define which news categories are non-sensitive (e.g., entertainment) and spend privacy budget only for sensitive ones (e.g., politics, health).

3.2 Privacy-Preserving Online Serving

In this section, we introduce detailed modules in *PrivateRec* serving as shown in Fig. 1, which lay the foundation for designing *PrivateRec* training. When the training finishes, any user with a deployed model can request for a recommendation service via online serving by sending the user embedding $\mathbf{u}$ to $\mathcal{S}$. With mechanism $\mathcal{M}$ on the user-end and a news pool N_c on the server-end, we are supposed to ensure $\mathcal{M}$ satisfies (ϵ_s, δ)-DP in Definition 1 when a user sends $\mathcal{M}(D)$ to the server for recommendation results over N_c.

Definition 1. For two adjacent user behavior logs $D \in \mathbb{D}$ and $D' \in \mathbb{D}$ with only one clicked behavior different and any output $z \in \text{Range}(\mathcal{M})$, a mechanism $\mathcal{M}: D \to \text{Range}(\mathcal{M})$ is (ϵ, δ)-differentially private if and only if:

$$\Pr[\mathcal{M}(D) = z] \le e^{\epsilon} \cdot \Pr[\mathcal{M}(D') = z] + \delta.$$

Algorithm 1 Online Serving in *PrivateRec*

User: $u \in \mathcal{U}_s$: $N_h, \theta, \epsilon_s, \delta_s, p, \Theta = \vec{\mathbf{b}} \circ \Theta_U \circ \Theta_N$
1: $\tilde{\boldsymbol{\alpha}} = \text{GetPrivAttn}(N_h, \theta, \epsilon_s, \delta_s, p, \vec{\mathbf{b}}, \Theta_U, \Theta_N)$
2: send $\tilde{\boldsymbol{\alpha}}$ to $\mathcal{S}$
Server $\mathcal{S}$: $\tilde{\boldsymbol{\alpha}}, \vec{\mathbf{b}}, N_s$
3: reconstruct user embedding $\check{\mathbf{u}} = \sum_i^B \tilde{\alpha}_i \mathbf{b}_i$
4: get news embeddings for $n_i \in N_c$ with $\mathbf{r}'_{i \in [C]} = \Theta_N(n_i)$
5: get matching scores over C news $\vec{s} = \{s_i = \check{\mathbf{u}}^\top \cdot \mathbf{r}'_i, i \in [C]\}$
6: return the ranked news list

3.2.1 Vanilla DP User Embedding (VDP). A conventional paradigm [12] for this goal is locally perturbing the user embedding. The user embedding is first clipped to θ with $\bar{\mathbf{u}} = \frac{\mathbf{u}}{\max(1, \frac{\|\mathbf{u}\|_2}{\theta})}$. Then a noise vector $\tilde{n}$ is drawn from the Gaussian distribution $\mathcal{N}(0, \sigma^2 \mathbf{I}_{d \times d})$, where $\sigma = \frac{S}{\epsilon_s}\sqrt{2\log\frac{1.25(1-p)}{\delta}}$. The sensitivity $S = \max_{D \simeq D'} ||\mathcal{M}(D) - \mathcal{M}(D')||_2 = 2\theta$ bounds the maximum change that one clicked item causes to the user embedding. Finally, the user sends $\tilde{\mathbf{u}} = \bar{\mathbf{u}} + \tilde{\boldsymbol{n}}$ to the server. If the Laplace mechanism is applied when $\delta = 0$, $\tilde{\mathbf{n}}$ is drawn from $\text{Lap}(S/\epsilon_s)$. Usually, the user embedding with a larger dimension encodes more information. However, it is obvious that the intensity of noise $\mathbb{E}[||\tilde{\boldsymbol{n}}||^2]$ scales with the dimension d of the user embedding. In other words, the information in a higher-dimensional user embedding would be submerged by the noise. Hence, VDP cannot provide a decent trade-off between utility and privacy, as we validate in Section 4.

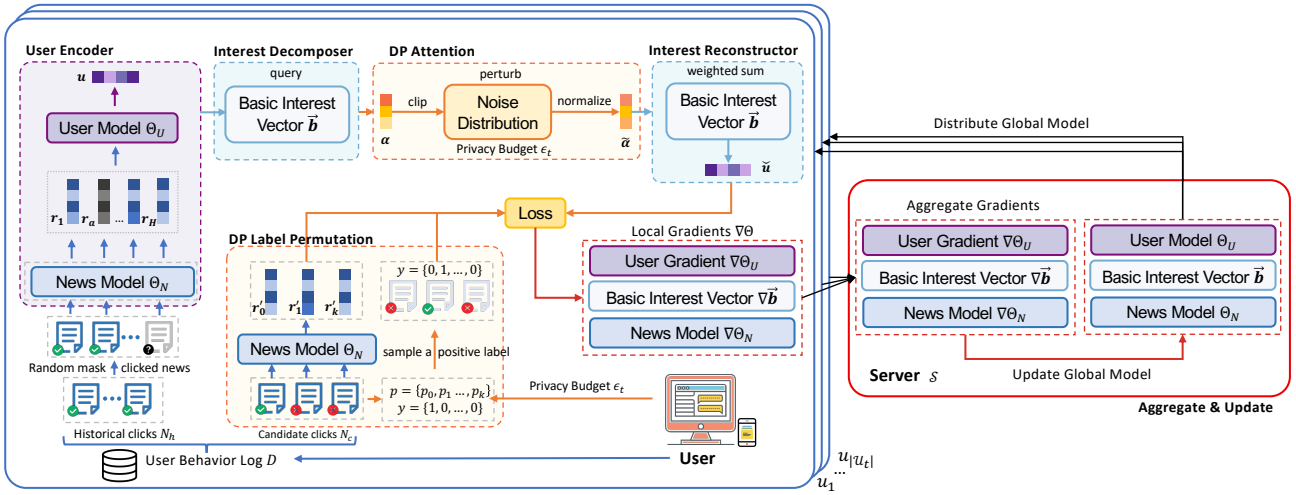

Figure 2: Privacy-preserving model training in *PrivateRec*.

Algorithm 2 GetPrivAttn(·)

Input: $N_h, \theta, \epsilon, p, \vec{\mathbf{b}}, \Theta_U, \Theta_N = \Theta_e \circ \Theta_f \circ \mathbf{e}_0$
Output: $\tilde{\boldsymbol{\alpha}}$

1: $\mathbf{r}_0 = \Theta_f(\vec{\mathbf{e}_0})$, sensitivity $S = \theta$
2: **for** each clicked item $n_i \in N_h$ **do**
3: $\quad \mathbf{r}_i = \Theta_N(n)$
4: $\quad \tilde{\mathbf{r}}_i = \begin{cases} \mathbf{r}_i & \text{w.p. } 1-p, \\ \mathbf{r}_0 & \text{w.p. } p. \end{cases}$
5: **end for**
6: user embedding $\mathbf{u} = \Theta_U(\vec{\tilde{\mathbf{r}}})$
7: attention score $\boldsymbol{\alpha} = \text{Softmax}(\frac{QK^\top}{\sqrt{d}})$, where $Q = \mathbf{u}, K = \vec{\mathbf{b}}$
8: clipping $\bar{\boldsymbol{\alpha}} = \frac{\boldsymbol{\alpha}}{\max(1, \frac{\|\boldsymbol{\alpha}\|_2}{\theta})}$
9: sample noise vector $\tilde{n} \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_{B\times B}), \sigma = \frac{S}{\log\frac{e^\epsilon - p}{1-p}} \sqrt{2\log\frac{1.25(1-p)}{\delta}}$
10: return private attention vector $\tilde{\boldsymbol{\alpha}} = \{\tilde{\alpha}_i = \frac{\text{SoftPlus}(\bar{\alpha}_i + \tilde{n}_i)}{\sum_j \text{SoftPlus}(\bar{\alpha}_j + \tilde{n}_j)}, i \in [B]\}$

3.2.2 DP User Embedding with Interest Decomposition. Thus, we are motivated to reduce the intensity of noise by decomposing $\mathbf{u}$ into a lower-dimensional vector before the perturbation. As shown in Algorithm 1, each user in *PrivateRec* sends the perturbed lower dimensional vector, which is motivated by the interest decomposition idea of Uni-FedRec[33]. Different from Uni-FedRec[33], we decompose a single user embedding instead of a set of interest embeddings for multiple clusters. The server can reconstruct a user embedding to the original dimension d with $\vec{\mathbf{b}}$, which is a set of basic vectors to represent B abstract user interests. $\vec{\mathbf{b}}$ is public for $\mathcal{U}_s$ and privacy-preserving for $\mathcal{U}_t$ because training it has spent the privacy budget (ϵ_t, δ_t) of $\mathcal{U}_t$.

The key step of the interest decomposition is Line 7 in Algorithm 2. We decompose the user embedding $\mathbf{u}$ into a low-dimensional vector $\boldsymbol{\alpha}$ by querying $\mathbf{u}$ to $\vec{\mathbf{b}}$ with a scaled dot product. Since variant user interests can be generalized into several basic vectors [46], we have $B \ll d$. Thus, the intensity of the DP noise only scales with B, which avoids the dimension curse on $\mathbf{u}$.

Then, we provide the DP guarantee by perturbing the vector $\boldsymbol{\alpha}$ with the Differentially Private Attention (DPA) module, as shown from Line 9 to Line 10 of Algorithm 2. It should be noted that the sensitivity is $S = \theta$ because the attention scores are always positive for any two adjacent datasets. We use SoftPlus [15] for positive attention weights and apply normalization to keep the summation to 1. After receiving a B dimensional attention vector $\boldsymbol{\alpha}$, $\mathcal{S}$ can reconstruct the user embedding with a weighted summation over $\vec{\mathbf{b}}$ in Line 3 of Algorithm 1. So the communication cost that a user spends for a recommendation query is reduced from $O(d)$ to $O(B)$.

3.2.3 Privacy Amplification by Behavior Padding. Based on the interest decomposition, we further reduce the noise magnitude with the privacy amplification effect from a random padding, as shown from Line 1 to Line 5 in Algorithm 2. In other words, by randomly masking some items with public information, we can apply a smaller σ for a given privacy. Previous works [24, 40] that utilize this effect usually mask values into null, which inevitably incur the information loss. Instead, we pad the masked news to

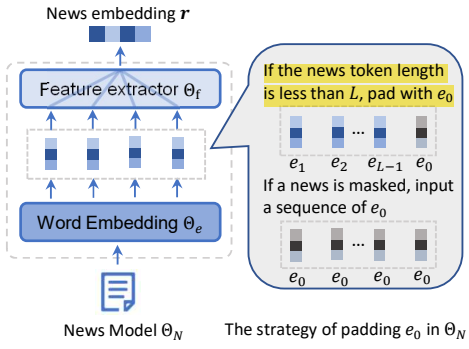

Figure 3: Details of the news model and the padding vector.

an anonymous news embedding $\boldsymbol{r}_0$ with the probability of p. It is generated from the feature extractor Θ_f by inputting a sequence of padding token embedding $\mathbf{e}_0$, as shown in Fig. 3. Since the padding token embedding $\mathbf{e}_0$ is used to pad empty token in training, the generated $\boldsymbol{r}_0$ is more informative than null because it can encode some general interest information.

Same as $\overrightarrow{\mathbf{b}}$, $\mathbf{e}_0$ is trained by spending $\mathcal{U}_t$'s privacy budget, thus padding with the generated r_0 does not incur any extra privacy concerns than masking items to null values for $\mathcal{U}_s$. It can improve the model serving utility with less noise under the same level of privacy while reducing the information loss caused by masking actual items to null. We will elaborate on how to train the padding token embedding $\mathbf{e}_0$ and basic vectors $\overrightarrow{\mathbf{b}}$ in the next section. Ultimately, we can derive the privacy guarantee as follows.

Theorem 2. *With GetPrivAttn(·) as the local mechanism $\mathcal{M}$, sending $\tilde{\boldsymbol{\alpha}} = \mathcal{M}(D)$ to $\mathcal{S}$ in Algorithm 1 is (ϵ_s, δ_s)-DP for any $D \in \mathbb{D}$.*

3.3 Privacy-Preserving Model Training

As shown in Fig. 2, the global model in *PrivateRec* is $\Theta = \Theta_N \circ \Theta_U \circ \overrightarrow{\mathbf{b}}$. Compared to the conventional federated news recommendation framework [32], we introduce a set of basic vectors $\overrightarrow{\mathbf{b}}$ for getting the basics to decompose the user embedding. For each training round as shown in Algorithm 3, the server $\mathcal{S}$ first samples r percent users from $\mathcal{U}_t$ and distributes Θ to them. Then, the sampled user u_i updates the local copy of Θ with the mechanism $\mathcal{M}$ and sends $\nabla\Theta_i$ to the server. Finally, the server aggregates local gradients and updates Θ with FedAdam [34]. We aim to guarantee the $\mathcal{M}$ that outputs local gradients satisfies (ϵ_t, δ_t)-DP over $D = N_h \circ N_c$.

The DPA module from Line 14 provides (ϵ_t, δ_t)-DP for N_h. Recall that we require an anonymous news embedding $\mathbf{r}_0$ which can encode general context information for all items. We illustrate two kinds of padding in Fig. 3. For a news of which the token length is less than L, we pad it with $\boldsymbol{e}_0$. For a news that is sampled to be masked, we replace all word embeddings with $\boldsymbol{e}_0$ for getting a anonymous news embedding $\boldsymbol{r}_0$ Hence, as the news model Θ_N learning to capture the context information, the padding vector $\mathbf{e}_0$ is trained to encode general information across all items, which makes it more informative than a direct nullification [40]. With the post-processing property [13], the privacy guarantee holds for N_h for the rest local processing. For a more stable model convergence, we replace the SoftPlus function [15] in Algorithm 2 to Relu [2].

The Differentially Private Label Permutation (DPL) module is designed for protecting clicking behavior in N_c. For each item in the candidate set N_c, we privately permute the label by sampling a positive one based on the exponential mechanism [27], as shown in Line 17 of Algorithm 3. With the parallel composition property of differential privacy [19], we have:

Theorem 3. *With LocalUpdate(·) as the local mechanism $\mathcal{M}$, sending $\nabla\Theta = \mathcal{M}(D)$ to $\mathcal{S}$ in Algorithm 3 is (ϵ_t, δ_t)-DP for any $D \in \mathbb{D}$.*

4 EXPERIMENTS

We conduct experiments on two real-world datasets: MIND[1] [47] and NewsFeeds with three news recommendation models: NRMS [44], NAML [41], and PLM-NR [45]. Baselines in our experiments include: 1) *Centralized* recommendation, where the server trains the model over all collected personal data. 2) *DP-FedRec*, where a global recommendation model is trained over local data. Local gradient and user embedding is perturbed for training and serving in *DP-RedRec* for a comparison under the same privacy level. Under the privacy definition 1, the noise is applied to each local gradient in training with gradient clipping norm $\theta = 0.005$ and to the user embedding in serving with clipping threshold $\theta = 0.001$. It should be noted that $\epsilon = \infty$ indicates the non-private *FedRec* without any noise. We show results with the off-the-shelf Laplacian mechanism[13] and $\delta = 0$. Gaussian mechanism[6, 13] can achieve a similar trend. Details of the baselines and hyperparameters are listed in the Appendix.

[1]MIND-small from https://msnews.github.io/

Algorithm 3 Model Training in *PrivateRec*

Input: $\Theta = \Theta_N \circ \Theta_U \circ \overrightarrow{\mathbf{b}}, \epsilon_t, \delta_t, D = N_h \circ N_c$, r
Output: Θ^T

1: ▷ Run by the server $\mathcal{S}$
2: initialize Θ
3: **for** round $t \in [T]$ **do**
4: sample m users from $\mathcal{U}_t$, $m = \lfloor r \cdot |\mathcal{U}_t| \rfloor$
5: **for** each sampled user u_i **do**
6: pull Θ^t from the $\mathcal{S}$
7: $\nabla\Theta_i^t = \text{LocalUpdate}(\Theta^t)$
8: **end for**
9: $\Theta^{t+1} = FedAdam(\Theta^t, \nabla\Theta^t_{i\in[m]})$
10: **end for**
11: deploy the final global model Θ^T to local devices
12:
13: ▷ LocalUpdate(·)
14: $\tilde{\boldsymbol{\alpha}} = \text{GetPrivAttn}(N_h, \theta, \epsilon_t, \delta_t, p, \overrightarrow{\mathbf{b}}, \Theta_U, \Theta_N)$
15: reconstruct user embedding $\tilde{\mathbf{u}} = \sum_i^B \tilde{\alpha}_i \mathbf{b}_i$
16: randomly sample k non-clicked items from N_c
17: sampling probability $p_{i\in[k+1]} = \frac{e^{y_i\beta}}{k+e^\beta}$ where $\beta = \max\{0, \log\frac{k}{C-1}e^{\epsilon_t}\}$
18: sample one item with the probabilities of $p_{i\in[k+1]}$ and set $y_i = 1$
19: get candidate news embeddings for $n_i \in N_c$ with $\mathbf{r}'_{i\in[k+1]} = \Theta_N(n_i)$
20: calculate matching score for each item $s_{i\in[k+1]} = \mathbf{u}^\top \mathbf{r}'_i$
21: get the gradient $\nabla\Theta = \frac{\partial\mathcal{L}}{\partial\Theta}$, where $\mathcal{L} = -\sum_{i=1}^{k+1} y_i \times \log\frac{e^{s_i}}{\sum_{j=1}^{k+1} e^{s_j}}$

4.1 Private Training Performance

First, we focus on evaluating the utility of privacy-preserving federated training by setting $\epsilon_s = \infty$ and compare *PrivateRec* with *DP-FedRec*. From Table, 1, we can see that models in *Centralized* recommendation can achieve the best performance at the cost of user privacy. *FedRec* with $\epsilon_t = \infty$ can mitigate the direct privacy leakage without data collection and achieve the suboptimal performance, but no theoretical privacy is ensured during training. For privacy-preserving baselines, we observe that *PrivateRec* outperforms *DP-FedRec* over both datasets with higher metric scores. Also, we observe a larger utility drop from *FedRec* with $\epsilon_t = \infty$ to *DP-FedRec* on the PLM-NR model, because the noise amount on gradients scales with the model size.

Then, we visualize the privacy-utility trade-off in federated training with various privacy budget $\epsilon_t = \{1, 5, 10, 20, \infty\}$ in Fig. 6. First, the model performance of *PrivateRec* is better than *DP-FedRec* for all budgets ϵ_t. Additionally. the performance of *PrivateRec* for a small privacy budget $\epsilon_t = 5$ is still acceptable with AUC 61.35. Second,

Table 1: Performance of centralized and federated training methods. (ϵ_t and ϵ_s indicate the privacy budget for training and serving under the Definition 1. ∞ indicates no privacy protection is applied. - indicates direct personal data collection.)

Model	ϵ_t	Baseline	NewsFeeds				MIND			
			AUC	MRR	nDCG5	nDCG10	AUC	MRR	nDCG5	nDCG10
NAML	-	Centralized	64.01±0.14	31.08±0.13	33.31±0.14	41.15±0.14	65.35±0.16	31.36±0.07	33.82±0.09	39.55±0.08
	∞	DP-FedRec	63.22±0.08	30.29±0.08	32.34±0.09	40.27±0.08	63.04±0.10	29.93±0.08	32.08±0.10	37.75±0.10
	10	DP-FedRec	54.84±0.55	25.08±0.34	25.67±0.43	33.47±0.44	55.31±0.31	24.88±0.16	25.91±0.16	31.52±0.17
	10	PrivateRec	61.05±0.37	28.46±0.32	30.03±0.38	38.12±0.37	57.51±0.42	26.27±0.26	27.76±0.31	33.34±0.26
NRMS	-	Centralized	64.48±0.13	31.57±0.12	33.94±0.14	41.72±0.14	65.82±0.11	31.68±0.08	34.29±0.06	39.99±0.07
	∞	DP-FedRec	63.32±0.19	30.46±0.18	32.56±0.21	40.44±0.21	62.80±0.47	29.59±0.34	31.66±0.37	37.32±0.34
	10	DP-FedRec	55.48±0.74	25.22±0.40	25.91±0.54	33.82±0.58	54.59±0.54	24.75±0.27	25.64±0.32	31.28±0.30
	10	PrivateRec	61.69±0.16	29.04±0.15	30.70±0.23	38.77±0.20	58.32±0.20	26.56±0.28	27.93±0.69	33.67±0.49
PLM-NR	-	Centralized	64.69±0.21	31.42±0.19	33.71±0.24	41.67±0.21	67.08±0.15	32.49±0.10	35.21±0.14	40.94±0.13
	∞	DP-FedRec	63.48±0.35	30.35±0.26	32.46±0.33	40.46±0.33	66.05±0.25	31.72±0.21	34.28±0.30	40.05±0.28
	10	DP-FedRec	57.74±0.11	26.05±1.17	27.01±1.48	35.19±1.41	54.87±0.21	24.45±0.25	25.51±0.33	31.18±0.25
	10	PrivateRec	62.52±0.17	29.60±0.18	31.50±0.21	39.52±0.22	58.75±0.14	26.70±0.28	28.22±0.32	33.90±0.25

Table 2: Performance of privacy-preserving model serving.

Model	ϵ_t	ϵ_s	Baseline	NewsFeeds				MIND			
				AUC	MRR	nDCG5	nDCG10	AUC	MRR	nDCG5	nDCG10
NAML	∞	10	DP-FedRec	50.59±0.08	22.44±0.06	22.28±0.04	30.19±0.03	50.32±0.03	22.75±0.05	23.41±0.04	28.94±0.03
	10	10	DP-FedRec	50.25±0.05	22.21±0.06	21.96±0.07	29.92±0.03	50.11±0.09	22.69±0.07	23.31±0.09	28.83±0.08
	10	10	PrivateRec	**61.63±0.19**	**29.10±0.21**	**30.81±0.26**	**38.81±0.22**	**57.52±0.13**	**26.39±0.22**	**27.88±0.26**	**33.45±0.22**
NRMS	∞	10	DP-FedRec	50.32±0.14	22.37±0.11	22.18±0.13	30.07±0.13	50.23±0.06	22.79±0.05	23.44±0.06	28.98±0.05
	10	10	DP-FedRec	50.18±0.11	22.11±0.06	21.86±0.10	29.74±0.06	50.19±0.10	22.71±0.06	23.34±0.07	28.87±0.07
	10	10	PrivateRec	**61.37±0.22**	**28.77±0.16**	**30.42±0.21**	**38.47±0.20**	**57.00±0.04**	**26.15±0.14**	**27.49±0.18**	**32.74±0.11**
PLM-NR	∞	10	DP-FedRec	50.44±0.03	21.75±0.06	21.40±0.06	29.45±0.03	50.59±0.01	22.61±0.06	23.18±0.05	28.78±0.06
	10	10	DP-FedRec	50.16±0.09	21.85±0.07	21.57±0.09	29.48±0.08	50.03±0.04	22.17±0.06	22.70±0.11	28.26±0.10
	10	10	PrivateRec	**62.62±0.14**	**29.74±0.10**	**31.57±0.11**	**39.66±0.11**	**58.69±0.29**	**26.37±0.11**	**27.75±0.18**	**33.57±0.08**

the decomposition of user interest has a negligible effect on the federated training utility with $B = 5$. When $\epsilon_t = \infty$, the performance of *PrivateRec* is comparable to the non-private baseline with $\epsilon_t = \infty$. This observation validates our claim that thousands of user interests can be summarized into several basic interests.

4.2 Privacy-Preserving Serving Performance

Then, we evaluate the privacy-preserving model serving for *PrivateRec* and *DP-FedRec* in Table 2 under the same serving privacy budget ϵ_s. First, we can observe that for two versions of baseline, i.e., *DP-FedRec* with or without private training, *PrivateRec* can achieve a significantly better performance across all models and datasets.

Then we investigate the privacy-utility trade-off in Fig. 4. First, we can observe that the model serving performance of *PrivateRec* is consistently better than *DP-FedRec* under various serving privacy budgets ϵ_s. Second, we find the a proper amount of noise injected in *PrivateFec* training can help the privacy-preserving model serving. From Fig. 4, the performance of *PrivateRec* with $\epsilon = \infty$ in the dashed line cannot outperform privacy-preserving training *PrivateRec* for ϵ_s from 0.001 to 10. This fact emphasizes our claim that the design of privacy-preserving federated training should co-design with the privacy-preserving model serving as an end-to-end solution for the federated recommendation system. If ϵ_t is too small (e.g, 1), the performance upper bound under ($\epsilon_s = \infty$) is limited. If ϵ_t is too large (e.g., 20), the performance upper bound is higher. And under a smaller ϵ_s, it is not as robust as the performance with a moderate $\epsilon_t = 10$. Thus, we conclude that the ϵ_t can be the knob to tune the privacy-utility trade-off in privacy-preserving model serving. More importantly, we find that the utility gain of *PrivateRec* in federated training can maintain for privacy-preserving model serving, whereas *DP-FedRec* cannot. As shown in Fig. 4, the utility of *FedRec* with $\epsilon_s = \infty$ is better than *PrivateRec*. However, the model serving utility is catastrophically ruined by the DP noise.

Next, we evaluate the effect of privacy amplification by behavior padding in Fig. 5. The motivation for padding the sequence of users' historical behavior representations is to reduce the amount of noise with the privacy amplification effect by nullification. Thus, in the first sub-figure, we show the required noise perturbation scales for the same serving privacy budget ϵ_s w.r.t different padding ratios. We can see that the required noise is reduced with a larger padding ratio. Additionally, this effect is more significant when the privacy requirement for serving is stricter (i.e, with a smaller ϵ_s). Generally, p controls the trade-off between the personalized interest and the generalized interest as well as the strength of noise reduction. From

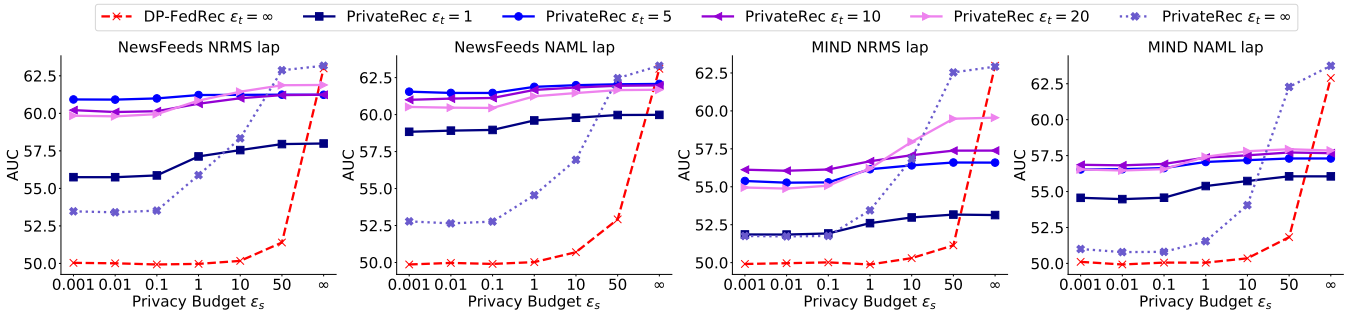

Figure 4: Performance of privacy-preserving model serving. ($\epsilon_t = \infty$ for *DP-FedRec* and $\epsilon_t = 10, B = 5$ for *PrivateRec*)

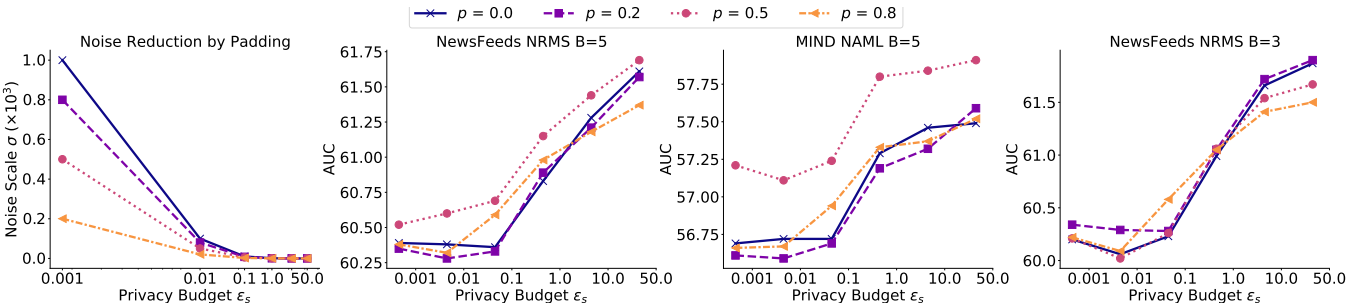

Figure 5: Effect of privacy amplification by behavior padding.

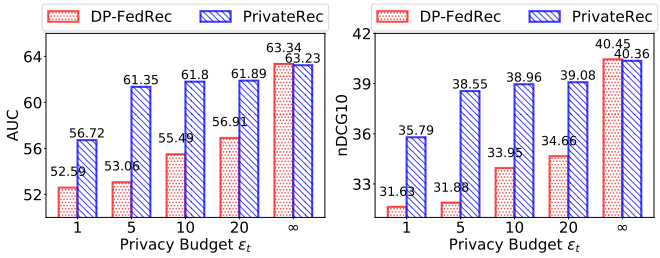

Figure 6: Performance of *DP-FedRec* and *PrivateRec* with different ϵ_t in federated training on NewsFeeds.

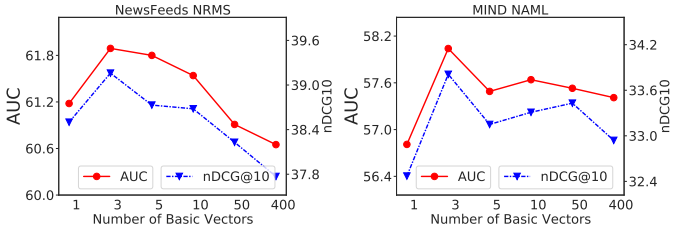

Figure 7: Influence of the number of basic vectors on the training performance of *PrivateRec* with $\epsilon_t = 10, \epsilon_s = \infty$.

the right three sub-figures, we can observe that if p is too small, the noise reduction by the privacy amplification is negligible. If p is too large, the personalized information is erased, which reduces the utility of model serving. Concretely, the best p for *PrivateRec* is 0.5 when $B = 5$ and 0.2 when $B = 3$. This is reasonable because a smaller B indicates more coarse-grained interest summarization, and the information for each basic vector is more abstract. Thus, for *PrivateRec* with a smaller B, the personalized interest information is more important, thus a smaller p is preferred.

Last, we evaluate the influence of the number of basic vectors in Fig. 7 for *PrivateRec* federated training. We observe that the best number of basic vectors for *NewsFeeds* and *MIND* datasets are $B = 3$. If B is too small, the basic vectors might be too coarse-grained and cannot express the personal user interest. If B is too large, the dimension of an attention vector α is large, which results in an increased amount of noise.

5 CONCLUSIONS

In this paper, we are the first to rethink and formulate the privacy definitions in federated news recommendation. Then, we propose a differentially private federated news recommendation framework *PrivateRec*, which can achieve a better utility for training and serving under a formal privacy definition. To avoid the dimension-dependent noise and improve utility in privacy-preserving federated training and serving, we decompose the high-dimensional and privacy-sensitive user embedding into a combination of public basic vectors and add noise to the combination coefficients. We further protect privacy in training with a label perturbation module. In addition, to further reduce noise, we utilize the amplification effect by randomly padding user historical behavior representations. Experiments on two real-world news recommendation datasets validate our method's effectiveness and utility improvement on the model training and serving stage. However, *PrivateRec* are designed

for recommendation models with individual user encoder and item encoder. Thus, we plan to extend the formulated privacy definition and privacy mechanisms for general recommendation models.

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 *CCS*. 308–318.

[2] Abien Fred Agarap. 2018. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375* (2018).

[3] Muhammad Ammad-Ud-Din, Elena Ivannikova, Suleiman A Khan, Were Oyomno, Qiang Fu, Kuan Eeik Tan, and Adrian Flanagan. 2019. Federated collaborative filtering for privacy-preserving personalized recommendation system. *arXiv preprint arXiv:1901.09888* (2019).

[4] Mingxiao An, Fangzhao Wu, Heyuan Wang, Tao Di, Jianqiang Huang, and Xing Xie. 2019. Neural ctr prediction for native ad. In *CCL*. Springer, 600–612.

[5] Mingxiao An, Fangzhao Wu, Chuhan Wu, Kun Zhang, Zheng Liu, and Xing Xie. 2019. Neural news recommendation with long-and short-term user representations. In *ACL*. 336–345.

[6] Borja Balle and Yu-Xiang Wang. 2018. Improving the gaussian mechanism for differential privacy: Analytical calibration and optimal denoising. In *International Conference on Machine Learning*. PMLR, 394–403.

[7] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. 2017. Practical secure aggregation for privacy-preserving machine learning. In *CCS*. 1175–1191.

[8] Di Chai, Leye Wang, Kai Chen, and Qiang Yang. 2020. Secure federated matrix factorization. *IEEE Intelligent Systems* (2020).

[9] Chen Chen, Jingfeng Zhang, Anthony KH Tung, Mohan Kankanhalli, and Gang Chen. 2020. Robust federated recommendation system. *arXiv preprint arXiv:2006.08259* (2020).

[10] Chaochao Chen, Jun Zhou, Bingzhe Wu, Wenjing Fang, Li Wang, Yuan Qi, and Xiaolin Zheng. 2020. Practical privacy preserving POI recommendation. *TIST* 11, 5 (2020), 1–20.

[11] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *RecSys*. 191–198.

[12] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In *TCC*. Springer, 265–284.

[13] Cynthia Dwork, Aaron Roth, et al. 2014. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.* 9, 3-4 (2014), 211–407.

[14] Dashan Gao, Ben Tan, Ce Ju, Vincent W Zheng, and Qiang Yang. 2020. Privacy threats against federated matrix factorization. *arXiv preprint arXiv:2007.01587* (2020).

[15] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep sparse rectifier neural networks. JMLR Workshop and Conference Proceedings, 315–323.

[16] Xiangnan He, Xiaoyu Du, Xiang Wang, Feng Tian, Jinhui Tang, and Tat-Seng Chua. 2018. Outer product-based neural collaborative filtering. *arXiv preprint arXiv:1808.03912* (2018).

[17] Mingkai Huang, Hao Li, Bing Bai, Chang Wang, Kun Bai, and Fei Wang. 2020. A Federated Multi-View Deep Learning Framework for Privacy-Preserving Recommendations. *arXiv preprint arXiv:2008.10808* (2020).

[18] Chao Li, Zhiyuan Liu, Mengmeng Wu, Yuchi Xu, Huan Zhao, Pipei Huang, Guoliang Kang, Qiwei Chen, Wei Li, and Dik Lun Lee. 2019. Multi-interest network with dynamic routing for recommendation at Tmall. In *CIKM*. 2615–2623.

[19] Ninghui Li, Min Lyu, Dong Su, and Weining Yang. 2016. Differential privacy: From theory to practice. *Synthesis Lectures on Information Security, Privacy, & Trust* 8, 4 (2016), 1–138.

[20] Tan Li, Linqi Song, and Christina Fragouli. 2020. Federated recommendation system via differential privacy. In *ISIT*. IEEE, 2592–2597.

[21] 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 (2021), 900–913.

[22] Feng Liang, Weike Pan, and Zhong Ming. 2021. Fedrec++: Lossless federated recommendation with explicit feedback. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 4224–4231.

[23] Guanyu Lin, Feng Liang, Weike Pan, and Zhong Ming. 2020. Fedrec: Federated recommendation with explicit feedback. *IEEE Intelligent Systems* (2020).

[24] Lingjuan Lyu, Xuanli He, and Yitong Li. 2020. Differentially private representation for nlp: Formal guarantee and an empirical study on privacy and fairness. *arXiv preprint arXiv:2010.01285* (2020).

[25] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*. PMLR, 1273–1282.

[26] H Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. 2017. Learning differentially private recurrent language models. *arXiv preprint arXiv:1710.06963* (2017).

[27] Frank McSherry and Kunal Talwar. 2007. Mechanism design via differential privacy. In *FOCS*. IEEE, 94–103.

[28] Khalil Muhammad, Qinqin Wang, Diarmuid O'Reilly-Morgan, Elias Tragos, Barry Smyth, Neil Hurley, James Geraci, and Aonghus Lawlor. 2020. Fedfast: Going beyond average for faster training of federated recommender systems. In *SIGKDD*. 1234–1242.

[29] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2019. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *SP*. IEEE, 739–753.

[30] Thông T Nguyên, Xiaokui Xiao, Yin Yang, Siu Cheung Hui, Hyejin Shin, and Junbum Shin. 2016. Collecting and analyzing data from smart device users with local differential privacy. *arXiv preprint arXiv:1606.05053* (2016).

[31] Shumpei Okura, Yukihiro Tagami, Shingo Ono, and Akira Tajima. 2017. Embedding-based news recommendation for millions of users. In *SIGKDD*. 1933–1942.

[32] Tao Qi, Fangzhao Wu, Chuhan Wu, Yongfeng Huang, and Xing Xie. 2020. Privacy-Preserving News Recommendation Model Training via Federated Learning. (2020).

[33] Tao Qi, Fangzhao Wu, Chuhan Wu, Yongfeng Huang, and Xing Xie. 2021. UniFedRec: A Unified Privacy-Preserving News Recommendation Framework for Model Training and Online Serving. *arXiv preprint arXiv:2109.05236* (2021).

[34] Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečnỳ, Sanjiv Kumar, and H Brendan McMahan. 2020. Adaptive federated optimization. *arXiv preprint arXiv:2003.00295* (2020).

[35] Steffen Rendle. 2010. Factorization machines. In *ICDM*. IEEE, 995–1000.

[36] Hyejin Shin, Sungwook Kim, Junbum Shin, and Xiaokui Xiao. 2018. Privacy enhanced matrix factorization for recommendation with local differential privacy. *TKDE* 30, 9 (2018), 1770–1782.

[37] Reza Shokri and Vitaly Shmatikov. 2015. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*. 1310–1321.

[38] Stacey Truex, Ling Liu, Ka-Ho Chow, Mehmet Emre Gursoy, and Wenqi Wei. 2020. LDP-Fed: Federated learning with local differential privacy. In *Proceedings of the Third ACM International Workshop on Edge Systems, Analytics and Networking*. 61–66.

[39] Hongwei Wang, Fuzheng Zhang, Xing Xie, and Minyi Guo. 2018. DKN: Deep knowledge-aware network for news recommendation. In *WWW*. 1835–1844.

[40] Ji Wang, Jianguo Zhang, Weidong Bao, Xiaomin Zhu, Bokai Cao, and Philip S Yu. 2018. Not just privacy: Improving performance of private deep learning in mobile cloud. In *SIGKDD*. 2407–2416.

[41] Chuhan Wu, Fangzhao Wu, Mingxiao An, Jianqiang Huang, Yongfeng Huang, and Xing Xie. 2019. Neural news recommendation with attentive multi-view learning. *arXiv preprint arXiv:1907.05576* (2019).

[42] Chuhan Wu, Fangzhao Wu, Mingxiao An, Jianqiang Huang, Yongfeng Huang, and Xing Xie. 2019. Npa: Neural news recommendation with personalized attention. In *SIGKDD*. 2576–2584.

[43] Chuhan Wu, Fangzhao Wu, Yang Cao, Yongfeng Huang, and Xing Xie. 2021. Fedgnn: Federated graph neural network for privacy-preserving recommendation. *arXiv preprint arXiv:2102.04925* (2021).

[44] Chuhan Wu, Fangzhao Wu, Suyu Ge, Tao Qi, Yongfeng Huang, and Xing Xie. 2019. Neural news recommendation with multi-head self-attention. In *EMNLP-IJCNLP*. 6389–6394.

[45] Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. 2021. Empowering News Recommendation with Pre-trained Language Models. *arXiv preprint arXiv:2104.07413* (2021).

[46] Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. 2021. Two Birds with One Stone: Unified Model Learning for Both Recall and Ranking in News Recommendation. *arXiv preprint arXiv:2104.07404* (2021).

[47] Fangzhao Wu, Ying Qiao, Jiun-Hung Chen, Chuhan Wu, Tao Qi, Jianxun Lian, Danyang Liu, Xing Xie, Jianfeng Gao, Winnie Wu, et al. 2020. Mind: A large-scale dataset for news recommendation. In *ACL*. 3597–3606.

[48] Xiaoyu Zhang, Chao Chen, Yi Xie, Xiaofeng Chen, Jun Zhang, and Yang Xiang. 2021. Privacy Inference Attacks and Defenses in Cloud-based Deep Neural Network: A Survey. *arXiv preprint arXiv:2105.06300* (2021).

[49] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In *SIGKDD*. 1059–1068.

[50] Ligeng Zhu, Zhijian Liu, and Song Han. 2019. Deep Leakage from Gradients. *NeurIPS* 32 (2019), 14774–14784.