

# Privacy-Preserving News Recommendation Model Learning

Tao Qi<sup>1</sup>, Fangzhao Wu<sup>2</sup>, Chuhan Wu<sup>1</sup>, Yongfeng Huang<sup>1</sup> and Xing Xie<sup>2</sup>

<sup>1</sup>Department of Electronic Engineering & BNRist, Tsinghua University, Beijing 100084, China

<sup>2</sup>Microsoft Research Asia, Beijing 100080, China

{taoqi.qt, wuchuhan15}@gmail.com yfhuang@tsinghua.edu.cn  
{fangzwu, xing.xie}@microsoft.com

## Abstract

News recommendation aims to display news articles to users based on their personal interest. Existing news recommendation methods rely on centralized storage of user behavior data for model training, which may lead to privacy concerns and risks due to the privacy-sensitive nature of user behaviors. In this paper, we propose a privacy-preserving method for news recommendation model training based on federated learning, where the user behavior data is locally stored on user devices. Our method can leverage the useful information in the behaviors of massive number users to train accurate news recommendation models and meanwhile remove the need of centralized storage of them. More specifically, on each user device we keep a local copy of the news recommendation model, and compute gradients of the local model based on the user behaviors in this device. The local gradients from a group of randomly selected users are uploaded to server, which are further aggregated to update the global model in the server. Since the model gradients may contain some implicit private information, we apply local differential privacy (LDP) to them before uploading for better privacy protection. The updated global model is then distributed to each user device for local model update. We repeat this process for multiple rounds. Extensive experiments on a real-world dataset show the effectiveness of our method in news recommendation model training with privacy protection.

## 1 Introduction

With the development of Internet and mobile Internet, online news websites and Apps such as Yahoo! News<sup>1</sup> and Toutiao<sup>2</sup> have become very popular for people to obtain news information (Okura et al.,

2017). Since massive news articles are posted online every day, users of online news services face heavy information overload (Zheng et al., 2018). Different users usually prefer different news information. Thus, personalized news recommendation, which aims to display news articles to users based on their personal interest, is a useful technique to improve user experience and has been widely used in many online news services (Wu et al., 2019b). The research of news recommendation has attracted many attentions from both academic and industrial fields (Okura et al., 2017; Wang et al., 2018; Lian et al., 2018; An et al., 2019; Wu et al., 2019a).

Many news recommendation methods have been proposed in recent years (Wang et al., 2018; Wu et al., 2019b; Zhu et al., 2019b). These methods usually recommend news based on the matching between the news representation learned from news content and the user interest representation learned from historical user behaviors on news. For example, Okura et al. (2017) proposed to learn news representations from the content of news articles via autoencoder, and learn user interest representations from the clicked news articles via Gated Recurrent Unit (GRU) network. They ranked the candidate news articles using the direct dot product of the news and user interest representations. These approaches all rely on the centralized storage of user behavior data such as news click histories for model training. However, users' behaviors on news websites and Apps are privacy-sensitive, the leakage of which may bring catastrophic consequences. Unfortunately, the centralized storage of user behavior data in server may lead to high privacy concerns from users and risks of large-scale private data leakage.

In this paper, we propose a privacy-preserving method for news recommendation model training. Instead of storing user behavior data on a central server, in our method it is locally stored on

<sup>1</sup><https://news.yahoo.com>

<sup>2</sup><https://www.toutiao.com/>

(and never leaves) users’ personal devices, which can effectively reduce the privacy concerns and risks (McMahan et al., 2017). Since the behavior data of a single user is insufficient for model training, we propose a federated learning based framework named *FedNewsRec* to coordinate massive user devices to collaboratively learn an accurate news recommendation model without the need to centralized storage of user behavior data. In our framework, on each user device we keep a local copy of the news recommendation model. Since the user behaviors on news websites or Apps stored in user device can provide important supervision information of how the current model performs, we compute the model gradients based on these local behaviors. The local gradients from a group of randomly selected users are uploaded to server, which are further aggregated to update the global news recommendation model maintained in the server. The updated global model is then distributed to each user device for local model update. We repeat this process for multiple rounds until the training converges. Since the local model gradients may also contain some implicit private information of users’ behaviors on their devices, we apply the local differential privacy (LDP) technique to these local model gradients before uploading them to server, which can better protect user privacy at the cost of slight performance sacrifice. We conduct extensive experiments on a real-world dataset. The results show that our method can achieve satisfactory performance in news recommendation by coordinating massive users for model training, and at the same time can well protect user privacy.

The major contributions of this work include:

(1) We propose a privacy-preserving method to train accurate news recommendation model by leveraging the behavior data of massive users and meanwhile remove the need to its centralized storage to protect user privacy.

(2) We propose to apply local differential privacy to protect the private information in local gradients communicated between user devices and server.

(3) We conduct extensive experiments on a real-world dataset to verify the proposed method in recommendation accuracy and privacy protection.

## 2 Related Work

### 2.1 News Recommendation

News recommendation can be formulated as a problem of matching between news articles and users.

There are three core tasks for news recommendation, i.e., how to model the content of news articles (news representation), how to model the personal interest of users in news (user representation), and how to measure the relevance between news content and user interest. For news representation, many feature based methods have been applied. For example, Lian et al. (2018) represented news using URLs, categories and entities. Recently, many deep learning based news recommendation methods represent news from the content using neural networks. For example, Okura et al. (2017) used denoising autoencoder to learn news representations from news content. Wu et al. (2019c) proposed to learn news representations from news titles via multi-head self-attention network. For user representation, existing news recommendation methods usually model user interest from their historical behaviors on news platforms. For example, Okura et al. (2017) learned user representations from the previously clicked news using GRU network. An et al. (2019) proposed a long- and short-term user representation model (LSTUR) for user interest modeling. It captures the long-term user interest via user ID embedding and the short-term user interest from the latest news click behaviors via GRU. For measuring the relevance between user interest and news content, dot product of user and news representation vectors is widely used (Okura et al., 2017; Wu et al., 2019a; An et al., 2019). Some methods also explore cosine similarity (Zhu et al., 2019b), feed-forward network (Wang et al., 2018), feature-interaction network (Lian et al., 2018).

These existing news recommendation methods all rely on centrally-stored user behavior data for model training. However, users’ behaviors on news platforms are privacy-sensitive. The centralized storage of user behavior data may lead to serious privacy concerns of users. In addition, the news platforms have high responsibility to prevent user data leakage, and have high pressure to meet the requirements of strict user privacy protection regulations like GDPR<sup>3</sup>. Different from existing news recommendation methods, in our method the user behavior data is locally stored on personal devices, and only the model gradients are communicated between user devices and server. Since the model gradients contain much less user information than the raw behavior data and they are further processed by the Local Differential Privacy (LDP) technique,

<sup>3</sup><https://eugdpr.org/>

our method can protect user privacy much better than existing news recommendation methods.

## 2.2 Federated Learning

Federated learning (McMahan et al., 2017) is a privacy-preserving machine learning technique which can leverage the rich data of massive users to train shared intelligent models without the need to centrally store the user data. In federated learning the user data is locally stored on user mobile devices and never uploaded to server. Instead, each user device computes a model update based on the local data, and the locally-computed model updates from many users are aggregated to update the shared model. Since model updates usually contain much less information than the raw user data, the risks of privacy leakage can be effectively reduced. Federated learning requires that the labeled data can be inferred from user interactions for supervised model learning, which can be perfectly satisfied in our news recommendation scenario, since the click and skip behaviors on news websites and Apps can provide rich supervision information.

Federated learning has been applied to training query suggestion model for smartphone keyboard and topic models (Jiang et al., 2019). There are also some explorations in applying federated learning to recommendation (Ammad et al., 2019; Chai et al., 2019). For example, Ammad et al. (2019) proposed a federated collaborative filtering (FCF) method. In FCF, the personal rating data is locally stored on user client and is used to compute the local gradients of user embeddings and item embeddings. The user embeddings are locally maintained in user client and are directly updated using the local gradient on each client. The item embeddings are maintained by a central server, and are updated using the aggregated gradients of many clients. Chai et al. (2019) proposed a federated matrix factorization method, which is very similar with FCF. However, these methods require all users to participate the process of federated learning to train their embeddings, which is not practical in real-world recommendation scenarios. Besides, these methods represent items using their IDs, and are difficult to handle new items since many news articles are posted every day which are all new items. Thus, these federated learning based recommendation methods have their inherent drawbacks, and are not suitable for news recommendation.

## 2.3 Local Differential Privacy

Local differential privacy (LDP) is an important technique to provide guarantees of privacy for sensitive information collection and analysis (Ren et al., 2018). It has attracted increasing attentions since user privacy protection has become a more and more important issue (Kairouz et al., 2014; Qin et al., 2016). A classical scenario of LDP is that there are a set of users, and each user  $u$  has a private value  $v$ , which is sent to a untrusted third-party aggregator so that the aggregator can learn some statistical information of the private value distribution among the users (Cormode et al., 2018). LDP can guarantee that the leakage of private information for each individual user is bounded by applying a randomized algorithm  $\mathcal{M}$  to private value  $v$  and sending the perturbed value  $\mathcal{M}(v)$  to the aggregator for statistical information inference. The randomized algorithm  $\mathcal{M}(\cdot)$  is called to satisfy  $\epsilon$ -local differential privacy if and only if for two arbitrary input private values  $v$  and  $v'$ , the following inequation holds:

$$\Pr[\mathcal{M}(v) = y] \leq e^\epsilon \Pr[\mathcal{M}(v') = y], \quad (1)$$

where  $y \in \text{range}(\mathcal{M})$ .  $\epsilon \geq 0$ , and it is usually called privacy budget. Smaller  $\epsilon$  means better private information protection. In many works (Sarathy and Muralidhar, 2010; Duchi et al., 2013),  $\mathcal{M}(\cdot)$  is implemented by adding Laplace noise to the private value. In this paper we apply LDP technique to the model gradients which are generated in user devices based on user behaviors and uploaded to server, to better protect user privacy and remove the need to a trusted server.

## 3 FedNewsRec for Privacy-Preserving News Recommendation

In this section we introduce our *FedNewsRec* method for privacy-preserving news recommendation model training. We first describe the news recommendation model. Then we describe the details of *FedNewsRec*.

### 3.1 Basic News Recommendation Model

Following previous works (Wu et al., 2019a; An et al., 2019), the news recommendation model in our method can be decomposed into two core sub-models, i.e., a *news model* to learn news representations and a *user model* to learn user representations.

**News Model** The *news model* aims to learn news representations to model news content. Its ar-

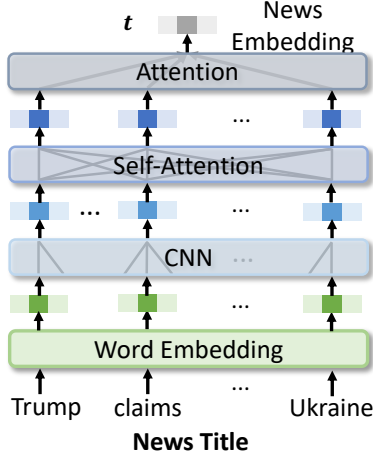


Figure 1: The architecture of *news model*.

architecture is shown in Fig. 1. Following (Wu et al., 2019b), we learn news representations from news titles. The *news model* contains four layers stacked from bottom to up. The first layer is word embedding, which converts the word sequence in a news title into a sequence of semantic word embedding vectors. The second layer is a CNN network, which is used to learn word representations by capturing local contexts. The third layer is a multi-head self-attention network (Vaswani et al., 2017), which can learn contextual word representations by modeling the long-range relatedness between different words. The fourth layer is an attention network, which is used to build a news representation vector  $\mathbf{t}$  from the output of multi-head self-attention network by selecting informative words.

**User Model** The *user model* is used to learn user representations to model their personal interest. Its architecture is shown in Fig. 2. Following (Okura et al., 2017), we learn user representation from their clicked news articles. Motivated by the LSTUR model proposed by An et al. (2019), we learn representations of users by capturing both long-term and short-term interests. The difference is that in LSTUR the embeddings of user IDs are used to model long-term interest, while in our *user model* it is learned from all the historical behaviors through a combination of a multi-head self-attention network and an attentive pooling network. This is because in the federated learning scenario, it is not practical that all users can participate the process of model training. Thus, the ID embeddings of many users in LSUTR cannot be learned. For short-term user interest modeling, our *user model* applies a GRU network to the recent behaviors of users, which is the same with LSUTR. The embeddings

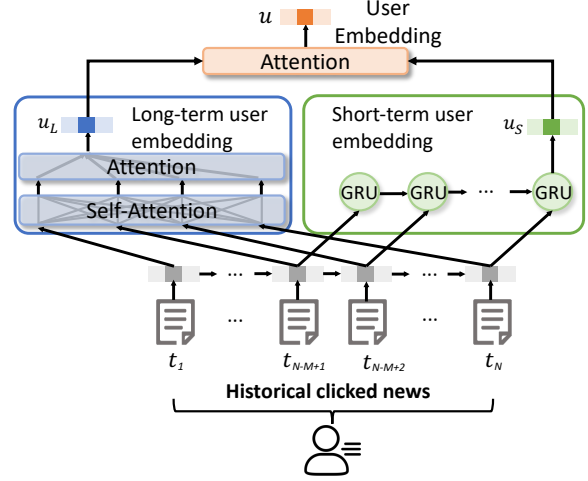


Figure 2: The architecture of *user model*.

of long-term interest and short-term interest are combined with an attention network into a unified user embedding vector  $\mathbf{u}$ .

**Model Training from User Behavior** Users' behaviors on news websites and Apps can provide useful supervision information to train the news recommendation models. For example, if a user  $u$  clicks a news article  $t$  which has low ranking score predicted by the model, then we can tune the model to give higher ranking score for this user-news pair. We propose to train the news recommendation model based on both click and non-click behaviors. More specifically, following (Wu et al., 2019b), for each news  $t_i^c$  clicked by user  $u$ , we randomly sample  $H$  news which are displayed in the same impression but not clicked. Assume this user has  $B_u$  click behaviors in total, then the loss function of the news recommendation model with parameter set  $\Theta$  is defined as:

$$\mathcal{L}_u(\Theta) = \sum_{i=1}^{B_u} L^i, \quad (2)$$

$$L^i = -\log\left(\frac{\exp(s(u, t_i^c))}{\exp(s(u, t_i^c)) + \sum_{j=1}^H \exp(s(u, t_{i,j}^{nc}))}\right), \quad (3)$$

where  $t_i^c$  and  $t_{i,j}^{nc}$  are clicked and non-clicked news articles shown in the same impression.  $s(u, t)$  is the ranking score of news  $t$  for user  $u$ , which is defined as the dot product of their embedding vectors, i.e.,  $s(u, t) = \mathbf{u}^T \mathbf{t}$ .

### 3.2 The Framework of FedNewsRec

Next, we introduce our *FedNewsRec* framework for privacy-preserving news recommendation model

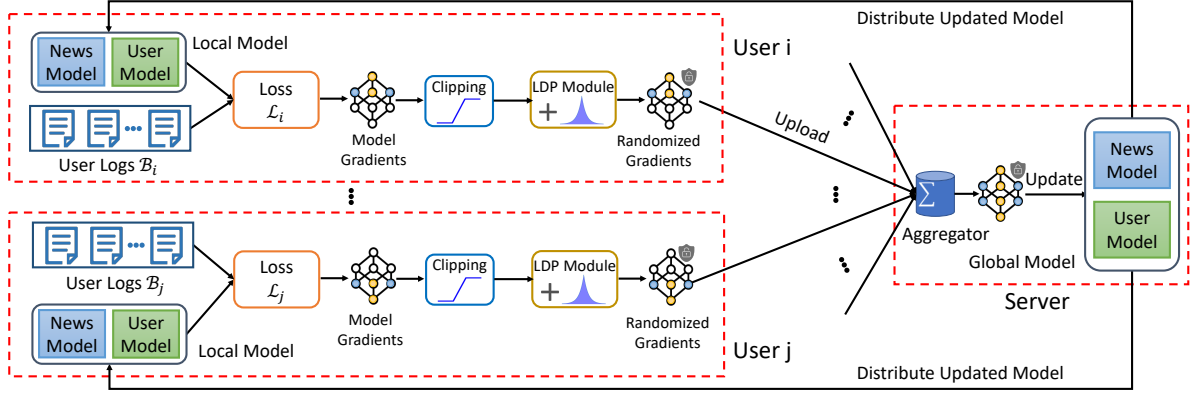


Figure 3: The framework of our privacy-preserving news recommendation approach.

training, which is shown in Fig. 3. In our *FedNewsRec* framework, the user behaviors on news platforms (websites or Apps) are locally stored on the user devices and never uploaded to server. In addition, the servers which provide news services do not record nor collect the user behaviors, which can reduce the privacy concerns of users and the risks of data leakage. Since an accurate news recommendation model can effectively improve users' news reading experiences and the behavior data from a single user is far from sufficient for training an accurate and unbiased model, in our *FedNewsRec* framework we propose to coordinate a large number of user devices to collectively train intelligent news recommendation models.

Following (McMahan et al., 2017), each user device which participates the model training is called a client. Each client has a copy of the current news recommendation model  $\Theta$  which is maintained by the server. Assume user  $u$ 's client has accumulated a set of behaviors on news platforms which is denoted as  $\mathcal{B}_u$ , then we compute a local gradient of model  $\Theta$  according to the behaviors  $\mathcal{B}_u$  and the loss function defined in Eq. (3), which is denoted as  $g_u = \frac{\partial \mathcal{L}_u}{\partial \Theta}$ . Although the local model gradient  $g_u$  is computed from a set of behaviors rather than a single behavior, it may still contain some private information of user behaviors (Zhu et al., 2019a). Thus, for better privacy protection, we apply local differential privacy (LDP) technique to the local model gradients. Denote the randomized algorithm applied to  $g_u$  as  $\mathcal{M}$ , which is defined as:

$$\mathcal{M}(g_u) = \text{clip}(g_u, \delta) + n, \quad (4)$$

$$n \sim \text{La}(0, \lambda), \quad (5)$$

where  $n$  is Laplace noise with 0 mean. The param-

eter  $\lambda$  can control the strength of Laplace noise, and larger  $\lambda$  can bring better privacy protection. The function  $\text{clip}(x, y)$  is used to limit the value of  $x$  with the scale of  $y$ . It is motivated by some studies which show that applying gradient clipping can help avoid potential gradient explosion and is beneficial for model training (Zhang et al., 2019). Denote the randomized gradient as  $\tilde{g}_u = \mathcal{M}(g_u)$ . After clip and randomization operation, it is more difficult to infer the raw user behaviors from the gradients. The user client uploads the randomized local model gradient  $\tilde{g}_u$  to the server.

In our *FedNewsRec* framework, a server is used to maintain the news recommendation model and update it via the model gradients from a large number of users. In each round, the server randomly selects a random fraction  $r$  (e.g., 10%) of the user clients, and sends the current news recommendation model  $\Theta$  to them. Then it collects and aggregates the local model gradients from the selected user clients as follows:

$$\bar{g} = \frac{1}{\sum_{u \in \mathcal{U}} |\mathcal{B}_u|} \sum_{u \in \mathcal{U}} |\mathcal{B}_u| \cdot \tilde{g}_u, \quad (6)$$

where  $\mathcal{U}$  is the set of users selected for the learning process in this round, and  $\mathcal{B}_u$  is the set of behaviors of user  $u$  for local model gradient computation.

Then the aggregated gradient  $\bar{g}$  is used to update the global news recommendation model  $\Theta$  maintained in the server:

$$\Theta = \Theta - \eta \cdot \bar{g}, \quad (7)$$

where  $\eta$  is the learning rate. The updated global model is then distributed to user devices to update their local models. This process is repeated until the model training converges.

### 3.3 Discussions on Privacy Protection

Next, we discuss why our *FedNewsRec* framework can protect user privacy in news recommendation model training. First, in our method the private user behavior data is stored on user own devices, and is never uploaded to server. Only the model gradients inferred from the local user behaviors are communicated with the server. According to the data processing inequality (McMahan et al., 2017), these gradients never contain more private information than the raw user behaviors, and usually contain much less information (McMahan et al., 2017). Thus, the user privacy can be better protected compared with the centralized storage of user behavior data as did in existing news recommendation methods. Second, the local model gradients are computed from a group of user behaviors instead of a single behavior. Thus, it is not very easy to infer a specific behavior from the local model gradients uploaded to server. Third, we apply the local differential privacy technique to the local model gradients before uploading by adding Laplace noise to them. It can strengthen the privacy protection of the private information in local model gradients. According to (Choi et al., 2018), Laplace noise in LDP can achieve  $\epsilon$ -local differential privacy, and  $\epsilon = \frac{\max_{v,v'} |\mathcal{M}(v) - \mathcal{M}(v')|}{\lambda}$ , where  $v$  and  $v'$  are arbitrary values in local model gradient. Since the upper bound of  $\max_{v,v'} |\mathcal{M}(v) - \mathcal{M}(v')|$  in our *FedNewsRec* framework is  $2\delta$ , the upper bound of the privacy budget  $\epsilon$  is  $\frac{2\delta}{\lambda}$ . We can see that by increasing  $\lambda$  (i.e., the strength of the noise), we can achieve a smaller privacy budget  $\epsilon$  which means better privacy protection. However, strong noise will hurt the accuracy of aggregated gradients. Thus,  $\lambda$  should be selected based on the trade-off between privacy protection and model performance.

## 4 Experiment

### 4.1 Dataset and Experimental Settings

Our experiments were conducted on a public news recommendation dataset (named *Adressa*) collected from a Norwegian news website (Gulla et al., 2017) and another real-world dataset collected from Microsoft News<sup>4</sup> (named *MSN-News*).<sup>5</sup> For the *Adressa* dataset, following Hu et al. (2020), we used user logs in the first five days to construct

<sup>4</sup><https://www.msn.com/en-us>

<sup>5</sup>Our dataset and codes will be publicly available in <https://github.com/JulySinceAndrew/FedNewsRec-EMNLP-Findings-2020>.

	<i>MSN-News</i>	<i>Adressa</i>
# users	100,000	528,514
# news	118,325	16,004
# impressions	1,341,853	-
# positive behaviors	2,006,289	2,411,187
# negative behaviors	48,051,601	-
avg. # title length	11.52	6.60

Table 1: The statistical information of the dataset.

users’ click history, used logs in the 6-th day for model training, and used logs in the 7-th day for model evaluation. Since the *Adressa* dataset does not contain non-clicked data, we randomly sampled 20 news as negative samples for each click to construct the test set. For the *MSN-News* dataset, we randomly sampled 100,000 users and their behavior logs in 5 weeks (from October 19 to November 22, 2019). We assume that the behavior logs of different users are stored in a decentralized way to simulate the real application of privacy-preserving news recommendation model training. We used the behaviors in the last week for test and the remaining behaviors for training. In addition, since in practical applications not all users can participate the model training, we randomly selected half of the users for training and tested the model on all users. The detailed statistics of the two datasets are listed in Table 1. Following (Wu et al., 2019b), we used the average scores of AUC, MRR, nDCG@5, nDCG@10 of all impressions in the test set to evaluate the performance. We repeated each experiment five times and reported average results and stand errors.

In experiments we used the 300-dimensional pre-trained Glove embedding to initialize word embeddings. The number of the self-attention head is 20 and the output dimension of each head is 20. The dimension of GRU hidden state is 400.  $H$  in Eq. (3) is 4. The fraction  $r$  of users participating in model training in each round is 2%. The learning rate  $\eta$  in Eq. (7) is 0.5.  $\delta$  in Eq. (4) is 0.005 and  $\lambda$  in Eq. (5) is 0.015. These hyper-parameters are all selected according to cross-validation on the training set.

### 4.2 Effectiveness Evaluation

First, we verify the effectiveness of the proposed *FedNewsRec* method. We compared with many methods, including: (1) *FM* (Rendle, 2012), factorization machine, a classic method for recommendation; (2) *DFM* (Lian et al., 2018), deep fusion model for news recommendation; (3) *EBNR* (Okura

Method	MSN-News				Adressa			
	AUC	MRR	nDCG@5	nDCG@10	AUC	MRR	nDCG@5	nDCG@10
FM	58.41±0.04	27.19±0.05	28.98±0.04	34.57±0.06	61.94±0.80	26.59±0.33	22.69±0.54	32.17±0.46
DFM	61.25±0.26	28.68±0.10	30.62±0.21	36.38±0.23	65.14±0.69	34.74±0.89	33.17±1.46	39.79±1.08
EBNR	63.64±0.15	29.50±0.14	31.57±0.13	37.38±0.20	65.70±0.72	30.23±0.49	29.37±0.53	36.38±0.44
DKN	62.38±0.19	29.40±0.15	31.59±0.11	37.27±0.21	67.53±1.90	32.33±2.79	31.84±2.78	39.96±2.52
DAN	62.54±0.23	29.44±0.18	31.67±0.14	37.31±0.25	64.03±3.10	33.37±2.63	31.61±3.03	38.60±3.02
NAML	64.52±0.24	30.93±0.17	33.39±0.16	39.07±0.19	69.20±2.07	35.18±1.49	34.78±1.85	42.34±1.97
NPA	64.29±0.20	30.63±0.15	33.11±0.17	38.89±0.23	66.70±2.42	34.68±1.77	33.72±2.09	41.18±1.99
NRMS	65.72±0.16	31.85±0.20	34.59±0.18	40.25±0.17	67.97±2.23	33.16±2.54	32.37±3.59	40.41±2.82
FCF	51.03±0.27	22.24±0.14	22.97±0.21	28.44±0.23	53.33±1.28	23.04±2.68	20.24±2.77	27.09±2.61
FedNewsRec	64.65±0.15	30.60±0.09	33.03±0.11	38.77±0.10	69.91±2.53	35.55±1.85	33.74±2.45	41.47±2.78
CenNewsRec	66.45±0.17	31.91±0.22	34.62±0.18	40.33±0.24	71.02±2.09	36.31±2.52	35.73±3.71	43.98±2.52

Table 2: The news recommendation results of different methods.

et al., 2017), using GRU for user modeling (Cho et al., 2014); (4) *DKN* (Wang et al., 2018), using knowledge-aware CNN network for news representation in news recommendation; (5) *DAN* (Zhu et al., 2019b), using CNN to learn news representations from both news title and entities and using LSTM to learn user representations; (6) *NAML* (Wu et al., 2019a), learning news representations via attentive multi-view learning; (7) *NPA* (Wu et al., 2019b), using personalized attention network to learn news and user representations; (8) *NRMS* (Wu et al., 2019d), learning representations of news and users via multi-head self-attention network; (9) *FCF* (Ammad et al., 2019), a federated collaborative filtering method for recommendation; (10) *CenNewsRec*, which has the same news recommendation model with *FedNewsRec* but is trained on centralized user behavior data.

The results are listed in Table 2. First, by comparing *FedNewsRec* with SOTA news recommendation methods such as *NRMS*, *NPA* and *EBNR*, our method can achieve comparable and even better performance on news recommendation. It validates the effectiveness of our approach in learning accurate models for personalized news recommendation. Moreover, different from these existing news recommendation methods which are all trained on centralized storage of user behavior data, in our *FedNewsRec* the user behavior data is stored on local user devices and is never uploaded. Thus, our method can train accurate news recommendation model and meanwhile better protect user privacy.

Second, our method can perform better than existing federated learning based recommendation methods like *FCF* (Ammad et al., 2019). The performance of *FCF* is not good in news recommendation. This is because *FCF* requires each user and each item to participate the training process to learn

their embeddings. However, in practical application not all the users can participate the training due to different reasons. In addition, news articles on online news platforms expire very quickly, and new news articles continuously emerge. Thus, many items for recommendation are news items, and unseen in the training data, which cannot be handled by *FCF*. In our method we learn news representations from news content and learn user representations from their behaviors using neural models. Therefore, our method can handle the problem of new users and new items, and is more suitable for news recommendation scenario.

Third, *FedNewsRec* performs worse than *CenNewsRec* which has the same news recommendation model with *FedNewsRec* but is trained on the centralized user behavior data. This is intuitive since centralized data is more beneficial for model training than decentralized data. In addition, in *FedNewsRec* we apply local differential privacy technique with Laplace noise to protect the private information in model gradients, which leads to the aggregated gradient for model update less accurate. Luckily, the gap between the performance of *FedNewsRec* and *CenNewsRec* is not very big. Thus, our *FedNewsRec* method can achieve much better privacy protection at the cost of acceptable performance decline. These results validate the effectiveness of our method.

### 4.3 Influence of User Number

In this section, we explore whether our *FedNewsRec* method can exploit the useful behavior information of massive users in a federated way to train accurate news recommendation models. In the following sections, we only show the experimental results on the *MSN-News* dataset. We randomly select different numbers of users for model training,

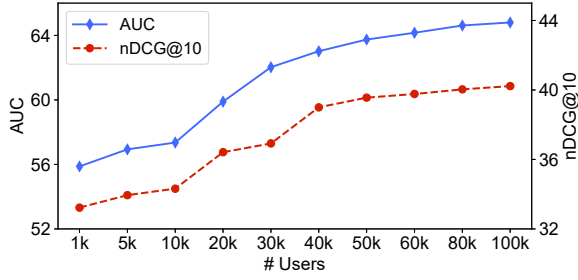


Figure 4: Performance with different numbers of users.

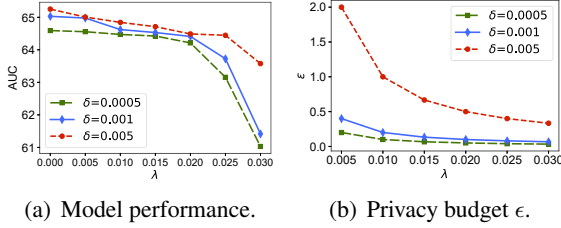


Figure 5: Influence of the hyper-parameters  $\lambda$  and  $\delta$ .

and use all users for evaluation. The experimental results are shown in Fig. 4.

From Fig. 4 we have several observations. First, when the number of users is small (e.g., 1000), the performance of news recommendation model trained on the behavior data of these users is not satisfactory. This is because the behaviors of a single user are usually very limited, and behavior data of a small number of users is insufficient to train accurate news recommendation model. This result validates the motivation of our *FedNewsRec* method to coordinate a large number of users in a federated way for model training. Second, when the number of users participating in training increases, the performance of *FedNewsRec* improves. It indicates that *FedNewsRec* can effectively exploit the useful behavior information from different users to collectively train an accurate news recommendation model, which validates the effectiveness of our framework. Third, when the number of users is big enough, further incorporating more users can only bring marginal performance improvement. This result shows that a reasonable number of users are sufficient for news recommendation model training, and it is unnecessary to involve too many or all users which is costly and impractical.

#### 4.4 Hyper-parameter Analysis

In this section, we explore the influence of hyper-parameters on our method. We show the results of two important hyper-parameters, i.e.,  $\delta$  in Eq. (4)

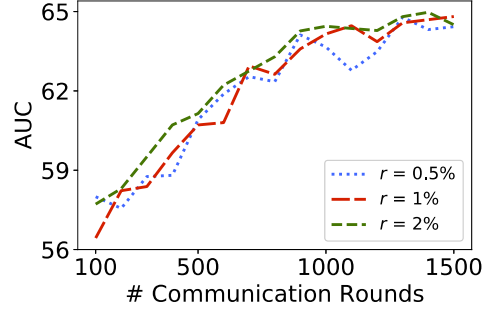


Figure 6: Convergence of model training.

and  $\lambda$  in Eq. (5) which serve in the local differential privacy module of our *FedNewsRec* framework. The results are shown in Fig. 5. In Fig. 5(a) we show the performance of our method with different  $\lambda$  and  $\delta$  values. We find that a large  $\lambda$  value can lead to the performance decline. This is because larger  $\lambda$  means stronger Laplace noise added to the gradients in LDP module, making the aggregated gradient for model update less accurate. In addition, our method tends to have better performance when  $\delta$  is larger. This is because fewer gradients will be affected in the clip operation when  $\delta$  is larger. In Fig. 5(b) we show the upper bound of the privacy budget, i.e.,  $\epsilon$  in Section 3.3, with different  $\lambda$  and  $\delta$  values. We can find that with larger  $\lambda$  value and smaller  $\delta$  value, the privacy budget  $\epsilon$  becomes lower, which means better privacy protection. This is intuitive, since larger  $\lambda$  value and smaller  $\delta$  value indicate that stronger noise is added and more gradient values are clipped, making it more difficult to recover the original model gradients. Combining Fig. 5(a) and Fig. 5(b) we can see that the better privacy protection is achieved by some sacrifice of the performance, and we need to select  $\lambda$  and  $\delta$  values based on the trade-off between privacy protection and news recommendation performance.

#### 4.5 Convergence Analysis

Next we explore the convergence of the model training in *FedNewsRec*, and the results are shown in Fig. 6. We can see that the training process can converge in about 1,500 rounds under different settings of  $r$  (i.e., ratio of selected users for model training in each round). It indicates that *FedNewsRec* can train news recommendation model efficiently.

#### 4.6 Effectiveness of User Model

In this section, we conduct ablation studies to evaluate the effectiveness of the short- and long- term user interest modeling in our user model. The ex-



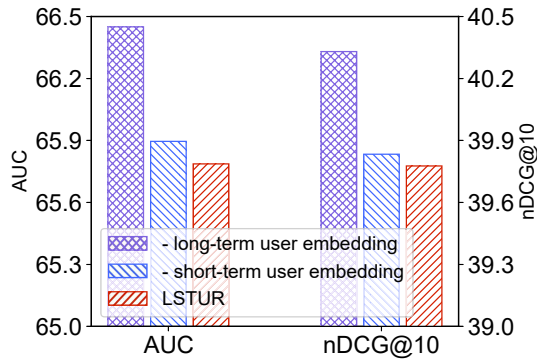


Figure 7: Effectiveness of User Model

perimental results are shown in Fig. 7, from which we have several observations. First, after removing the short-term user embedding, the performance of our method declines. This is because users sometimes tend to read news related to the topics they recently cared about (An et al., 2019). Our user model learns the short-term user embedding from the sequence of users’ recent clicked news via a GRU network, which can effectively capture users’ short-term interest. Thus, removing the short-term user embedding makes the unified user embedding loss some information of users’ recent reading preference and causes performance decline. Second, after removing the long-term user embedding, the performance of our method also declines. This is because users may read some news according to their long-term interests, which may not be reflected by their recent reading history (An et al., 2019). To address this issue, our user model learns long-term user embedding by capturing the relatedness among users’ clicked news, which can effectively capture users’ long-term interest. After removing it, the unified user embedding losses the information of the long-term interest, which hurts the recommendation accuracy.

## 5 Conclusion

In this paper, we propose a privacy-preserving method for news recommendation model training. Different from existing methods which rely on centralized storage of user behavior data, in our method the user behaviors are locally stored on user devices. We propose a *FedNewsRec* framework to coordinate a large number of users to collectively train accurate news recommendation models from the behavior data of these users without the need to upload it. In our method each user client computes local model gradients based on the user behaviors

on device, and sends them to server. The server coordinates the training process and maintains a global news recommendation model. It aggregates the local model gradients from massive users and updates the global model using the aggregated gradient. Then the server sends the updated model to user clients and this process is repeated for multiple rounds. In order to further protect the private information in the local model gradients, we apply local differential privacy to them by adding Laplace noise. The experiments on real-world dataset show that our method can achieve comparable performance with SOTA news recommendation methods, and meanwhile can better protect user privacy.

## Acknowledgments

This work was supported by the National Key Research and Development Program of China under Grant number 2018YFC1604002, and the National Natural Science Foundation of China under Grant numbers U1836204, U1705261, and 61862002.

## References

- Muhammad Ammad, 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*.
- 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*, pages 336–345.
- Di Chai, Leye Wang, Kai Chen, and Qiang Yang. 2019. Secure federated matrix factorization. *arXiv preprint arXiv:1906.05108*.
- Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111.
- Woo-Seok Choi, Matthew Tomei, Jose Rodrigo Sanchez Vicarte, Pavan Kumar Hanumolu, and Rakesh Kumar. 2018. Guaranteeing local differential privacy on ultra-low-power systems. In *ISCA*, pages 561–574.
- Graham Cormode, Somesh Jha, Tejas Kulkarni, Ninghui Li, Divesh Srivastava, and Tianhao Wang. 2018. Privacy at scale: Local differential privacy in practice. In *SIGMOD*, pages 1655–1658.

- John C Duchi, Michael I Jordan, and Martin J Wainwright. 2013. Local privacy and statistical minimax rates. In *FOCS*, pages 429–438.
- Jon Atle Gulla, Lemei Zhang, Peng Liu, Özlem Özgöbek, and Xiaomeng Su. 2017. The adressa dataset for news recommendation. In *Proceedings of the international conference on web intelligence*, pages 1042–1048.
- Linmei Hu, Siyong Xu, Chen Li, Cheng Yang, Chuan Shi, Nan Duan, Xing Xie, and Ming Zhou. 2020. Graph neural news recommendation with unsupervised preference disentanglement. In *ACL*, pages 4255–4264.
- Di Jiang, Yuanfeng Song, Yongxin Tong, Xueyang Wu, Weiwei Zhao, Qian Xu, and Qiang Yang. 2019. Federated topic modeling. In *CIKM*, page 1071–1080.
- Peter Kairouz, Sewoong Oh, and Pramod Viswanath. 2014. Extremal mechanisms for local differential privacy. In *NIPS*, pages 2879–2887.
- Jianxun Lian, Fuzheng Zhang, Xing Xie, and Guangzhong Sun. 2018. Towards better representation learning for personalized news recommendation: a multi-channel deep fusion approach. In *IJCAI*, pages 3805–3811.
- Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *AISTATS*, pages 1273–1282.
- Shumpei Okura, Yukihiro Tagami, Shingo Ono, and Akira Tajima. 2017. Embedding-based news recommendation for millions of users. In *KDD*, pages 1933–1942.
- Zhan Qin, Yin Yang, Ting Yu, Issa Khalil, Xiaokui Xiao, and Kui Ren. 2016. Heavy hitter estimation over set-valued data with local differential privacy. In *CCS*, pages 192–203. ACM.
- Xuebin Ren, Chia-Mu Yu, Weiren Yu, Shusen Yang, Xinyu Yang, Julie A McCann, and S Yu Philip. 2018. High-dimensional crowdsourced data publication with local differential privacy. *TIFS*, pages 2151–2166.
- Steffen Rendle. 2012. Factorization machines with libfm. *TIST*, page 57.
- Rathindra Sarathy and Krish Muralidhar. 2010. Some additional insights on applying differential privacy for numeric data. In *International Conference on Privacy in Statistical Databases*, pages 210–219.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*, pages 6000–6010.
- Hongwei Wang, Fuzheng Zhang, Xing Xie, and Minyi Guo. 2018. Dkn: Deep knowledge-aware network for news recommendation. In *WWW*, pages 1835–1844.
- Chuhan Wu, Fangzhao Wu, Mingxiao An, Jianqiang Huang, Yongfeng Huang, and Xing Xie. 2019a. Neural news recommendation with attentive multi-view learning. *IJCAI*, pages 3863–3869.
- Chuhan Wu, Fangzhao Wu, Mingxiao An, Jianqiang Huang, Yongfeng Huang, and Xing Xie. 2019b. Npa: Neural news recommendation with personalized attention. In *KDD*, pages 2576–2584.
- Chuhan Wu, Fangzhao Wu, Mingxiao An, Tao Qi, Jianqiang Huang, Yongfeng Huang, and Xing Xie. 2019c. Neural news recommendation with heterogeneous user behavior. In *EMNLP*, pages 4876–4885.
- Chuhan Wu, Fangzhao Wu, Suyu Ge, Tao Qi, Yongfeng Huang, and Xing Xie. 2019d. Neural news recommendation with multi-head self-attention. In *EMNLP*, pages 6390–6395.
- Jingzhao Zhang, Tianxing He, Suvrit Sra, and Ali Jadbabaie. 2019. Why gradient clipping accelerates training: A theoretical justification for adaptivity. In *ICLR*.
- Guanjie Zheng, Fuzheng Zhang, Zihan Zheng, Yang Xiang, Nicholas Jing Yuan, Xing Xie, and Zhenhui Li. 2018. Drn: A deep reinforcement learning framework for news recommendation. In *WWW*, pages 167–176.
- Ligeng Zhu, Zhijian Liu, and Song Han. 2019a. Deep leakage from gradients. *arXiv preprint arXiv:1906.08935*.
- Qiannan Zhu, Xiaofei Zhou, Zeliang Song, Jianlong Tan, and Guo Li. 2019b. Dan: Deep attention neural network for news recommendation. In *AAAI*, pages 5973–5980.