Privacy-Preserving Federated Depression Detection from Multi-Source Mobile Health Data

Xiaohang Xu, *Student Member, IEEE*, Hao Peng, *Member, IEEE*, Md Zakirul Alam Bhuiyan, *Senior Member, IEEE*, Zhifeng Hao, Lianzhong Liu, Lichao Sun, Lifang He, *Member, IEEE*

***Abstract*—Depression is one of the most common mental illnesses, and the symptoms shown by patients are different, making it difficult to diagnose in the process of clinical practice and pathological research. Although researchers hope that artificial intelligence can contribute to the diagnosis and treatment of depression, the traditional centralized machine learning methods need to aggregate patient data, and the data privacy of patients with mental illness needs to be strictly confidential, which hinders machine learning algorithms' clinical application. To solve the problem of medical data privacy with depression, we implement a study of federated learning to analyze and diagnose depression. First, we propose a general multi-view federated learning framework using multi-source data, which can extend any traditional machine learning model to support federated learning across different institutions or parties. Secondly, we employ later fusion methods to solve the problem of inconsistent time series of multi-view data. Finally, we compare the federated framework with other cooperative learning frameworks in performance and discuss the related results. The experimental results show that in the case of participating in federated learning with enough participants, the prediction accuracy of depression score can reach 85.13%, which is about 15% higher than local training. When the number of participants is small and the amount of data is sufficient, the prediction accuracy of depression score can also reach 84.32%, and the improvement rate is about 9%.**

***Index Terms*—Federated learning, Depression, Data privacy, Mobile device.**

I. Introduction

DEPRESSION is a very common disease in real life. More than 300 million people worldwide suffer from depression [1]. At present, the diagnosis of depression depends almost entirely on the subjective judgment of the doctor through communication with the patient and the relevant questionnaires filled out. Hamilton Depression Rating Scale (HDRS) [2] and Young Mania Rating Scale (YMRS) [3] are commonly used evaluation criteria for doctors when diagnosing depression. In order to better help doctors diagnose depression, researchers analyze patient data by introducing machine learning technologies [4], [5]. But when using machine learning technology, there is a contradiction between the performance of the model and the protection of data privacy [6].

First, traditionally, the trainer implements centralized training by collecting a large amount of data [7], [8]. Although Wang et al. [9] proposed a scheme to avoid data privacy leakage in centralized learning, hospitals need to protect the privacy of patients' diagnosis data, so different medical institutions cannot gather and share data [10], which greatly affects the accuracy of the model [11]. For example, in the work of electrocardiogram [12], because a single medical institution cannot collect enough high-quality data, the predictive ability of the model cannot achieve the role of clinical assistance. Second, although there are many privacy protection machine learning algorithms [13], [14], which are difficult to achieve good training results. The privacy protection machine learning method [15] needs to increase the noise according to the sensitivity of the algorithm's intermediate product, so under the limited privacy budget, the prediction performance of the privacy algorithm is often poor. Third, due to the huge gap between various medical institutions, the patient data they have varies greatly. In order to deal with various situations, algorithms and software are required to have a high generalization ability, and it is difficult for the model to obtain sufficient accuracy and specificity without data exchange.

To address the above data privacy leakage, privacy algorithms, and data quality limitations, in 2016, Google proposed a method called federated learning [16] to break the problem of data silos due to data privacy. Each participant does not need to centralize data to train a machine learning model, instead, it aggregates the trained model in one place and uses federated averaging technology to continuously optimize the model so that the data can be available to all participating facilities. However, Google has not implemented federated learning in the medical application, most of the research [17], [18] using the federated learning framework in the medical field is based on the existing data of hospitals. It mainly includes the diagnosis of the characteristics of patients with specific diseases, reducing the cost of diagnosis and treatment, medical image processing and other issues [11]. As mobile devices become more and more popular, smart phones, wearable

Manuscript received February 12 2021; revised July 20 2021; accepted August 31 2021. This work was supported by the National Key R&D Program of China (No.2018YFC0830804), the National Natural Science Foundation of China under Grant 62002007, S&T Program of Hebei under Grant 20310101D, Foundation of State Key Laboratory of Public Big Data (No.PBD2021-18), and Fundamental Research Funds for the Central Universities. *(Corresponding author: Hao Peng.)*

X. Xu is with the State Key Laboratory of Public Big Data, Guizhou University, Guizhou Guiyang, 550025, and with the School of Cyber Science and Technology, Beihang University, Beijing 100191, China. (e-mail: misslyysy@buaa.edu.cn).

H. Peng and L. Liu are with the School of Cyber Science and Technology, Beihang University, Beijing 100191, China. (e-mail: {penghao, lz_liu}@buaa.edu.cn).

M. Z. A. Bhuiyan is with the Department of Computer and Information Sciences Fordham University JMH 334, E Fordham Road, Bronx, NY 10458 USA. (e-mail: mbhuiyan3@fordham.edu).

Z. Hao is with the Department of Mathematics, College of Science, Shantou University, Guangdong 515063, China. (e-mail: zfhao@gdut.edu.cn).

L. Sun and L. He are with the Department of Computer Science and Engineering, Lehigh University, Bethlehem, PA 18015 USA. (e-mail: {lis221, lih319}@lehigh.edu).

devices and other devices are also recording users' information all the time, which are a risk of privacy leakage [19], [20]. According to existing researches [21], as one of the most important tools for information transmission in patients' lives, mobile phones can also be an important data source for disease prediction. We believe that keyboard keystrokes, such as the interval between two keystrokes, can be used as a form of biometric identification to predict depression by analyzing the keystroke habits of patients with depression. The typing speed of depression patients is usually different from that of normal people, which may be caused by emotional instability during the onset of the disease [22].

Our work uses a virtual keyboard customized for mobile phones to collect metadata (including key letters, special characters, and phone accelerometer values). Using more than 1.3 million key-presses from 20 users, each of whom additionally completed at least 1 patient health questionnaire. We regard the user's key-presses at least five seconds apart as the beginning and no operation after five seconds of the last keypress as the end of a session which is usually kept within 1 minute. We use federated learning architecture at the session level to model DeepMood [23], a deep learning architecture based on late data fusion. However, in real application, the amount of data held by different medical institutions is different, the number of medical institutions in different regions is also different. To this end, we divide our work into two parts. Firstly, we distribute the data to different parties for training according to the IID (Independent and Identically Distributed) method, but the amount of data that each party has during each training is not the same, and the number of parties participating in the training is also different each time. This setting simulates the real situation in different regions and different medical institutions. At the same time, in order to verify the influence of federated learning on model training, we simulated a data island environment and set up local training for each party that did not participate in federated learning. Furthermore, we assign the data to each party according to non-IID, and discuss the influence of non-IID on the prediction results. The experimental results show that the model prediction accuracy reaches 85.13% in the case of IID and 76.95% in the case of non-IID. Our code is open-sourced at https://github.com/RingBDStack/Fed_mood.

The contributions of this work can be included as follows:

- The first multi-source mobile health data application guided by federated learning is proposed, which makes full use of multi-view data to achieve privacy-preserving federated depression detection.
- In the IID setting, increasing the number of parties and the amount of data owned by parties will improve the accuracy of the depression diagnosed, but it can be affected by duplicate data. In the non-IID setting, the model based on the late data fusion has stronger robustness.
- Extensive experiments and analysis in medical depression detection prove that federated learning has the advantage of accuracy in the tasks of IID and non-IID data settings, which can improve accuracy 12% averaged.

The rest of this article is organized as follows. The section II section introduces the background of multi-view learning, federated learning and privacy protection. At the same time, we analyzed the principle of the late fusion model. The task definition and the federated learning framework are described in Section III. The data sources, experimental settings and results are outlined in section IV. Finally, we summarize the paper in section V.

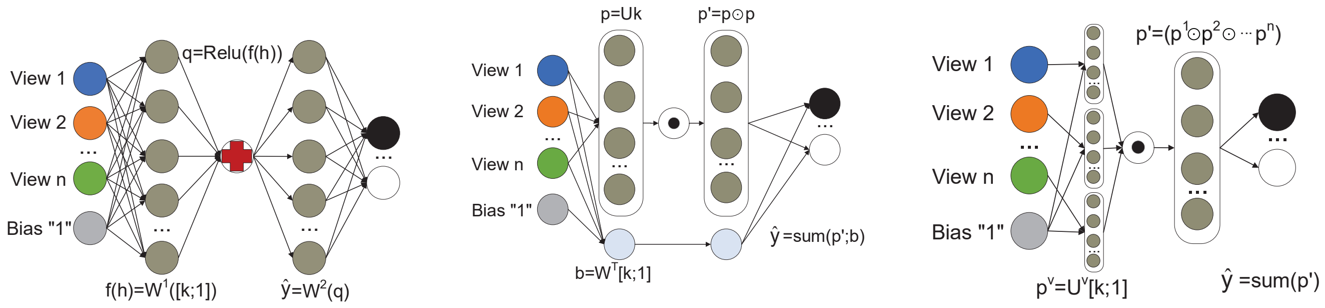
II. Background

A. Related Work

In this section, we introduce the related research results of federated learning and multi-view learning, and discuss the recent proposed federated multi-view learning.

Multi-View learning. Xu et al. [24] pointed out that multi-view learning requires the use of one function to model one perspective and uses other perspectives to jointly optimize all functions. Cao et al. [25] used tensor product to process multi-view data. Yao et al. [26] integrated CNN, LSTM, and graph embedding to tackle the complex nonlinear spatial and temporal dependency in a multi-view way. Shen et al. [27] improved the task of accurate left ventricular segmentation from heterogeneous data with cross-vendor, cross-center, and multi-view in the ultrasonic telemedicine application combining IoT (Internet of Things) and ultrasound. Nan et al. [28] divided the dataset into multi-view data according to the sensor source and achieved good scalability in medical applications through the consistency and complementarity of different view data. Rokni et al. [29] used a wearable device combined with a multi-view autonomous learning method to monitor the user's physical activity or medical complications in a highly dynamic environment without a large amount of labeled training data. Serra et al. [30] performed multi-view clustering to subtype patients and explained how to combine clustering and classification in a multi-view scenario to automatically diagnose neurodegenerative diseases. In addition, some work integrated multi-view into the process of deep learning [31] and transfer learning [32], so as to help expand samples from data.

Federated learning. Here we mainly refer to the medical application of federated learning and some common federated multi-view deep learning framework. Kim Y et al. [33] used joint analysis of data from multiple hospitals to discover the phenotype of a specific patient population under the condition that no data left the local hospital. In the case of federated learning, the algorithm can find the "sickle cell/chronic pain" characterization that cannot be found within a single hospital, avoiding deviations in the results due to population differences and small samples. Lee et al. [34] proposed a privacy protection platform in the federated environment, which could find similar patients from different hospitals without sharing patient-level information. Huang L et al. [35] improved the performance of federated learning for predicting mortality and length of stay by using feature autoencoders and patient clustering. There are also some studies that combine multi-perspective learning with federated learning. Adrian Flanagan et al. [36] proposed the federated



(a) Fully connected layer. (b) Factorization Machine layer. (c) Multi-view Machine layer.

Fig. 1. A comparison of different strategies for fusing multi-view data from the perspective of computational graph.

multi-view matrix factorization method and address cold-start problem. Huang et al. [37] proposed FL-MV-DSSM, which is the first general content-based joint multi-view framework, which successfully extended traditional federated learning to federated multi-view learning. Kang et al. [38] proposed the FedMVT algorithm for semi-supervised learning, which can improve the performance of vertical federated learning with limited overlapping samples. Most current federated multi-view learning researches focus on the solution of the cold start of the recommendation system, however, our framework uses data collected from mobile devices to solve medical mood prediction problems.

Federated privacy protection. There are three main ways [39] to protect data privacy in the federated learning framework: Secure Multi-Party Computing (SMC) and Differential Privacy Mechanism (DP) and Homomorphic Encryption (HE). Secure multi-party computing mainly uses secure communication and encryption algorithms to protect the model aggregation security of different participants in the federated learning [40]. Since the federated framework does not need to aggregate data but transmits gradients or model parameters, SMC only needs to encrypt related parameters, which saves a lot of encryption calculation cost. However, the improved strategy based on SMC still adds extra time cost compared with the traditional federated framework. How to balance the time cost and the loss of data value over time has become a problem to be solved. Differential privacy protects data privacy by adding noise to the data source, while ensuring that the loss of data quality is controllable [41], [42]. By adding noise to the models or gradients uploaded by participants, the contribution of personal data in the dataset is masked to prevent reverse data leakage. Because of the problem that the data after adding noise is still close to the original data, Sun et al. [43], [44] used local differential privacy and noise-free differential privacy mechanisms to decrease the risk of information exposure. However, the introduction of differential privacy may reduce the accuracy of the global model, and it will be difficult for the central server to measure the contribution of each party to encourage different parties to participate in the federation. Homomorphic encryption can calculate the ciphertext data without decryption [45]. In the federated framework, each party can homomorphically encrypt the parameters they want to upload, and the central server can complete the aggregation process of the federated model without decryption. Since data and models are not transmitted with plain text, there is no leakage of the original data level. However, local encryption and decryption operations increase computing power consumption, and the transmission of ciphertext also increases additional communication cost.

B. Later Fushion Model

Since the dataset we use has the problem that the time series under three views have different frequencies and cannot be aligned, in this section, we introduce the later fusion strategy adopted by the model to make the time series of the data consistent [46], [25]. We set the output vector at the end of the $p-th$ view sequence as $k^{(p)}$, and let $\{k^{(p)} \in R^{d_k}\}_{p=1}^{n}$ be the multi-view data where m is the number of views.

Fully connected layer. We first consider the simplest way to connect multi-views directly, ie, $k = [k^{(1)}; k^{(2)}; ...; k^{(n)}] \in R^d$, where d is the total number of multi-view features, and typically $d = (2)nd_k$ for one directional (bidirectional) GRU. The connected hidden state k is inserted into the fully connected neural network through a nonlinear function $\sigma(\cdot)$. The feature interaction mode of the input unit is as follows:

$$\begin{aligned} p &= relu(W^{(1)}[k;1]), \\ \hat{y} &= W^{(2)}p, \end{aligned} \tag{1}$$

where $W^{(1)} \in R^{k\times(d+1)}$, $W^{(2)} \in R^{c\times h}$, h is the number of hidden units, c is the number of classes, and the constant signal "1" is to model the global bias. To simplify the illustration, we only set a hidden layer as shown in Fig 1(a).

Factorization Machine layer. As shown in Fig 1(b), instead of transforming the input with a nonlinear function, we directly model the features of each input part as follows:

$$\begin{aligned} p_c &= U_c k, \\ b_c &= W_c^T[k;1], \\ \hat{y_c} &= sum([p_c \odot p_c; b_c]), \end{aligned} \tag{2}$$

where $U_c \in R^{f\times d}$, $W_c \in R^{d+1}$, f is the number of factor units, c denotes the c-th class, and $\odot$ is the element-wise multiplication.

Multi-view Machine layer. Only considering the second-order feature interaction of the input data may not be comprehensive enough. We nest interaction to the m-th order between m

views to generate the final output $\hat{y_c}$ for the c-th class in the following way:

$$\hat{y_c} = \beta_0 + \sum_{v=1}^{m}\sum_{i_v=1}^{d_v} \beta_{i_v}^{(v)} k_{i_v}^{(v)} + \cdots + \sum_{i_1=1}^{d_1}\cdots\sum_{i_m=1}^{d_m} \beta_i(\prod_{p=1}^{m} k_{i_v}^{(v)}), \quad (3)$$

where β is the global offset, the second part is the first-order fusion, and the last part is the m-th order fusion. Next, the output vector $k_{i_v}^{(v)}$ is combined with the constant 1 as an additional feature. The Eq. 3 can be rewritten as follows:

$$\hat{y_c} = \sum_{i_1=1}^{d_1+1}\cdots\sum_{i_m=1}^{d_m+1} \omega_{i_1,\cdots,i_m}(\prod_{v=1}^{m}[k_{i_v}^{(v)} : 1]), \quad (4)$$

where $\omega_{d_1+1,\ldots,d_m+1} = \beta_0$ and $\omega_{i_1,\ldots,i_m} = \beta_{i_1,\ldots,i_m}$, $\forall i_v \leq d_v$. Next, we decompose the m-th order weight tensor $\omega_{i_1,\ldots,i_m}$ into k factors: $C \times U^{(1)} \times \cdots \times U^{(m)}$. $U^{(m)} \in R^{k\times(d_h+1)}$ is the factor matrix of the m-th view and $C \in R^{k\times\cdots\times k}$ is the identity tensor. Finally, we transform Eq. 4 as follows:

$$\hat{y_c} = \sum_{i_1=1}^{d_k+1}\cdots\sum_{i_m=1}^{d_k+1}(\sum_{f=1}^{h}\prod_{v=1}^{m}[k_{i_v}^{(v)} : 1](i_v)). \quad (5)$$

As shown in the figure 1(c), we can simplify Eq. 5 as follows:

$$\begin{aligned} p_c^{(v)} &= U_c^{(v)}[k^{(v)}; 1], \\ \hat{y_c} &= sum([p_c^{(1)} \odot \ldots \odot p_c^{(m)}]). \end{aligned} \quad (6)$$

where $U_c^v \in R^{h\times d_k+1}$ is the factor matrix of the v-th view for the c-th class. $\hat{y_c}$ is the final output for the c-th class.

Finally, we apply the dropout operation before merging the output of each model to improve the performance of the deep neural network by preventing the joint action of feature detectors, thereby preventing overfitting. In the factorization machine layer and the multi-view machine layer, we can still directly calculate the gradient of the model parameters like the operation of the fully connected layer. Thus, the loss function of the final prediction result can be back-propagated back to each initial input view after fusion.

III. IoT-Data Silo Island Problem and Methodology

In this section, we introduce how to use IoT-data to train local and federated learning models. We first discuss the reasons for task definition, and then introduce the federated learning framework proposed by Google.

A. Problem Description

In the absence of federated learning frameworks, medical institutions can only use local datasets without interactive processes when using machine learning algorithms to build models for disease diagnosis, medical imaging research, and so on. We retain the local learning model as a comparison to measure the improvement effect of the federated learning algorithm on the multi-views heterogeneous data training model. We have conceived the following three situations.

At first there were several hospitals in a city $\{H_1, \ldots, H_m\}$. Assuming that patients with bipolar I disorder, bipolar II disorder, and normal people who are suspected of being sick will go to different hospitals for testing depression scores, the hospital will also record the patient's mobile terminal data. From a certain moment, we stop the collection of data by hospitals. At this time, each hospital has a fixed amount of data D_x. Each medical institution will first use its own local data to train the model and test its effect. The results obtained at this time are generally difficult to use as a reference for the diagnosis of depression. Each participant will cooperate with other medical institutions for federated training, and in this process, new medical institutions will continue to participate. Without reducing the total number of communication rounds, we increase the degree of parallelism to test the changes in the prediction effect.

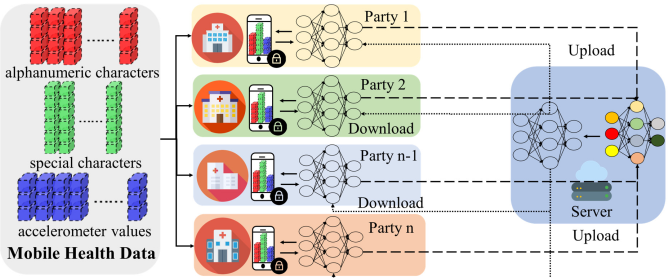


Fig. 2. The architecture of federated learning. Firstly, the participating parties have data on normal people, bipolar I and bipolar II users, and do not interact with different parties. At the beginning of each communication round, the server will assign the global model to all participating parties in this round. Next, all activated parties will train the local model through its own mobile health data and upload it to the server. Finally, the server updates the global model according to the uploaded local model.

At a certain moment, the number of hospitals in a certain city is constant with n $\{H_1, \ldots, H_n\}$, and no new hospitals will be established in this city for a certain period of time. Patients will go to different hospitals on average as described above, and all medical institutions will predict depression mood through local training and federated learning. Initially, each hospital has a small amount of data D_a. As patients continue to come to the hospital for treatment and review, the hospital will continue to increase the amount of data D_x. When the data added by each hospital reaches a mark value D_f, the participants will restart the training in hopes of improving the prediction effect.

In the actual medical environment, the data owned by each hospital must be non-IID. We assume that patients with bipolar I disorder, bipolar II disorder and normal people who are suspected of being sick will only go to a specific hospital H_x for treatment. Each medical institution has a different amount of patient data, and the serious condition of patients is inconsistent, resulting in extreme data distribution. On this basis, we detect the impact of this extreme distribution data for the accuracy of the model compared to IID data. The specific data division is introduced in section IV-C1.

B. Federated learning

Due to the privacy issues of patient health data stored in hospitals, we cannot use these data for centralized machine learning. So federated learning is a good solution to tackle the problem of data privacy, during the training of the federated learning model, the data owned by each hospital participating in the model training collaboration can be saved locally without uploading. Each hospital uses its own data to download the model from the server for training, and upload the trained model or gradient to the server for aggregation, and then the server sends the aggregated model or gradient information to each hospital. Considering the communication cost, connection reliability, and other issues, we adopt the model average method for training. Assuming that there are K hospitals participating in federated learning when the global model parameters are updated in the t round, the K-th participant calculates the local average gradient of the current model according to Eq. 7, and the server aggregates these gradients and uses them to update the global model according to Eq. 8.

$$g_k = \nabla F_k(\omega_t), \tag{7}$$

$$\omega_{t+1} \leftarrow \omega_t - \eta \sum_{k=1}^{K} \frac{n_k}{n} g_k, \tag{8}$$

where g_k is the average gradient of the current local model ω_t, and η is the learning rate.

According to Eq. 9, each hospital uses local data to perform one (or more) steps to calculate the gradient descent, the existing model parameters locally and sends the locally updated model parameters to the server. The server then calculates the weighted average of all the transmitted models according to Eq. 10 and sends the aggregated model parameters to each hospital.

The literature [16] shows that compared with the purely distributed SGD, the improved scheme can reduce the amount of communication used by 10-100 times, and can choose the update optimizer of the gradient other than SGD.

$$\forall k, \omega_{t+1}^{(k)} \leftarrow \overline{\omega}_t - \eta g_k, \tag{9}$$

$$\overline{\omega}_{t+1} \leftarrow \sum_{k=1}^{K} \omega_{t+1}^{(k)}, \tag{10}$$

where $\overline{\omega_t}$ is the existing model parameter of the local client.

In this work, we use the federated learning framework to implement the emotion prediction through multi-views heterogeneous data collected by the mobile terminal and focus on studying the influence of the number of participants and the amount of data in IID data and the model accuracy declined of non-IID data. We first fix the number of parties participating and change the amount of data owned by each party in the experiment. Next, we keep the amount of data owned by each party and increase different parties with the same amount of data to participate in the federated learning. Finally, we divide the dataset unequally to form a non-IID scenario to study the change of model accuracy.

Algorithm 1: Federated Averaging. The K is the total number of party, M is the local minibatch size, E is the number of local epochs, and η is the learning rate.

```
Server executes:
  Initialize $\omega_0$
  for each round $t = 1, 2, \dots$ do
    $t \leftarrow$ random choose$(1, K)$
    $C_t \leftarrow$(random set of $t$ parties)
    for each party $k \in C_t$ in parallel do
      $\omega_{t+1}^{(k)} \leftarrow localtraining(k, \overline{\omega}_t)$
    $\overline{\omega}_{t+1} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n} \omega_{t+1}^{(k)}$

Localtraining$(k, \overline{\omega}_t)$ : $//k$ parties training in parallel
  for each local epoch $i$ from 1 to $S$ do
    $D_m \leftarrow$(split $D_k$ into batches of size $M$ randomly)
    for each batches $b$ from 1 to $B = \frac{n_k}{M}$ do
      $\omega_{b+1}^{(k)} \leftarrow \omega_{b,i}^{(k)} - \eta \nabla F_k(\omega_t)$
  return $\omega_{t+1}^{(k)} = \omega_{B,S}^{(k)}$ to server
```

IV. Experiments

In this section, we introduce how to use data generated by personal mobile devices to train deep learning models. We assume that the hospital allocates a special mobile device to each user to collect alphanumeric characters, special characters, and accelerometer values used in the session, and the hospital performs a weekly HDRS test for patients. Due to the particularity of the diagnosis of depression, patients may go to multiple hospitals to try and seek treatment, and some hospitals may have the same patient data.

A. Dataset

The data used in the experiment comes from a real observation study based on a free mobile app named BiAffect. It should be noted that the data source of BiAffect is from users in the United States. Due to the large differences in input methods with different languages, it may not apply to countries that do not use English as their mother tongue. In the data collection stage, the researchers provide the users with a special Android smartphone. The special smartphone uses a customized virtual keyboard to replace the default keyboard, so as to collect the metadata input by the user without affecting the operation in the background. The metadata collected by the keyboard includes the user key input time, the number of keystrokes, and the phone accelerometer value. The three types of metadata collected are as follows:

Alphanumeric characters. In order to protect user privacy, BiAffect did not collect specific alphanumeric characters. It only collected the duration of the keypress, the duration before the last keypress was pressed, and the distance from the last key to the coordinate axis on the horizontal and vertical axes.

Special characters. Due to special characters having far fewer keystrokes than alphanumeric characters, BiAffect performed one-hot encoding for operations including space, backspace, and keyboard switching.

Accelerometer value. The accelerometer records every 60ms during every activated session. Because different users have different typing speeds, the accelerometer values are more densely recorded than alphanumeric characters.
We define a session as a duration that begins with lasting five or more seconds since the last keypress and lasting until five or more seconds passed between keypresses. Due to the user typing habits, the duration of the session is generally less than one minute.

All depression patients receive the Hamilton Depression Rating Scale (HDRS) [2] and the Young Man Mania Scale (YMRS) [3] which are the effective assessment questionnaire for bipolar disorder diagnosis once a week. After collecting all the tested patients, we divide the data with bipolar patients and the control group data with normal participants. There are 6 participants suffering from bipolar I disorder, including severe episodes ranging from bipolar disorder to depression, 6 participants suffering from bipolar II disorder, including clinical manifestations of mildly elevated mood between mild manic episodes and severe episodes, and 8 participants are diagnosed as normal users. Since the evaluation process only relies on the communication between the patient and the doctor, and the indicators given by the evaluation scale, the results of the diagnosis are not necessarily reliable. Therefore, we try to predict the occurrence of depression from an objective perspective by recording real-time data of patients.

B. Experimental Setup

Our model is implemented using Keras with Tensorflow as the backend. All experiments are conducted on a 64 core Intel Xeon CPU E5-2680 v4@2.40GHz with 512GB RAM and 1× NVIDIA Tesla P100-PICE GPU. We use RMSProp [47] as the training optimizer. We retain sessions with keypresses between 10 and 100, and finally generate 14960 samples. Each user contributes the first 80% sessions for training and the rest for validation.

TABLE I
Parameter configuration.

Parameter	Value
DNN communication rounds	400
DNN local epochs	15
DFM communication rounds	300
DFM local epochs	20
DMVM communication rounds	400
DMVM local epochs	15
Batch size	256
Learning rate	0.001
Dropout fraction	0.1
Maximum sequence length	100
Minimum sequence length	10

We set the parameters based on experience and some experimental comparisons, including the number of communication rounds, the number of local epochs, batch size, learning rate, and dropout rate. We consider sessions with the HDRS score between 0 and 7 (inclusive) as negative samples (normal) and those with HDRS greater than or equal to 8 as positive samples (from mild to severe depression).

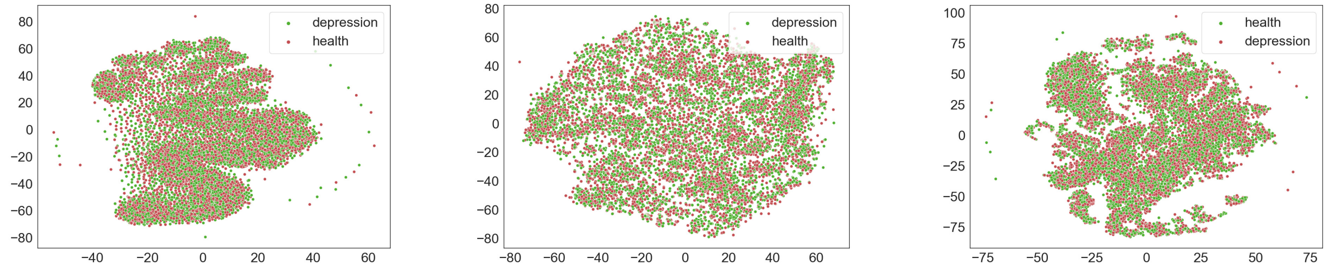
TABLE II
The accuracy of the compared models under different local epochs and communication rounds. We show the best results with boldface.

Communication Rounds		100	200	300	400	500
Model	local epochs					
DNN	5	79.19	82.95	84.35	**85.01**	84.42
	10	80.05	83.58	84.98	**85.25**	84.68
	15	83.35	86.18	86.35	**86.38**	85.21
	20	84.72	84.72	**85.21**	83.88	83.72
DFM	5	81.88	84.35	**84.91**	84.81	84.62
	10	81.82	84.32	84.48	84.48	**84.65**
	15	83.38	**85.18**	84.68	84.48	83.88
	20	84.02	85.25	**85.31**	85.01	84.15
DMVM	5	78.92	84.25	85.31	**86.85**	85.18
	10	81.39	84.68	**86.01**	85.78	84.72
	15	82.61	84.48	85.68	**86.95**	83.95
	20	81.01	81.98	**82.88**	82.45	82.55

In order to study the influence of local epochs parameters, we evenly distribute the training dataset to 8 participants for testing. The results are shown in the Table II, we can find: (1) As the number of communication rounds increases, the accuracy shows a trend of first rising and then a slight decrease. (2) Our work is different from the results of Zhao et al. [48]. A large number of local epochs can significantly improve the effect of federated learning. However, when epochs $\equiv 20$, the accuracy of DMVM in any communication round shows a downward trend. These results show that increasing the local epoch can make the training more stable and speed up the convergence speed, but it may not make the global model converge to a higher accuracy level. In other words, over-optimizing the local datasets may cause performance loss. (3) In the first 300 epochs, the fusion efficiency of DFM is higher than that of DNN and DMVM, which shows the improvement effect of the fusion layer, and DFM achieves better local minima of loss functions in some results. Compared with centralized learning, due to the sharp reduction of the amount of local data, the effect of DMVM fusion of multi-view and multi-level features will be affected to a certain extent. Because the three models get different results when the local epochs are 15 and 20, we perform the parameters separately, as shown in Table I.

C. IID Experiments

1) Compared Methods: We compare FedAVG with the following methods, each of which represents a different strategy for data interaction.
Local Training: Local training means that each party only uses its data for training, without any interaction with other parties.
CDS [49]: Collaborative data sharing is a traditional centralized machine learning strategy, which requires that each party uploads its patient data to the center server for training.
IIL [49], [50]: Institutional incremental learning is a serial training method. Each party transfers its model to the next participant after training finishes, until all have trained once.
CIIL [49], [50]: Cyclic institutional incremental learning repeats the IIL training process. It keeps consistent with the



(a) Alphanumeric characters.

(b) Special characters.

(c) Accelerometer value.

Fig. 3. Visualization of labeling with TSNE for three views

number of federated learning local training epochs and looping repeatedly through the parties.

In each experiment, the models we compared are summarized as follows:

DMVM: The proposed DeepMood architecture with a multi-view machine layer for data fusion.

DFM: The proposed DeepMood architecture with a factorization machine layer for data fusion.

DNN: The proposed DeepMood architecture with a conventional fully connected layer for data fusion.

In this work, for the IID setting, we randomly assign each client a uniform distribution of three data types: normal users, bipolar I disorder patients, and bipolar II disorder patients. The specific methods as follows: 1. The number of participants owned data remains unchanged, and the number of parallel participants is increasing. The amount of data owned by each party is fixed at 1500, and the number of hospitals participating in the training gradually increases from 4. We test the training effect of up to 24 parallel participants. 2. The number of parallel participants remains unchanged, and the amount of data owned by each participant is increasing. We set the number of concurrent participants to 8 to be consistent with the experiment of setting hyperparameters. The amount of data owned by each party gradually increases from 100, and we use about 25% (3000) of the total data as the maximum value of the experiment. To make the results of the experiment more stable, we conduct each group of experiments five times and average the results.

2) Evaluation criteria: In order to evaluate the influence of federated learning and local training on the prediction results, we adopt the following measures: Accuracy is one of the most frequently used criteria, which represents the ratio of the number of correctly predicted samples to the total number of predicted samples. In the federated learning experiment, the central server can test the final global model with its own test dataset. In the local training experiment, we regard the local data as a whole and compare the number of samples correctly predicted by each participant with the test dataset.

3) Experiment Result: Table III shows the mood prediction effect of increasing parallel parties. Since local training has no interactive process and the amount of data owned by each participant is constant, the final result has always been between 73% and 75%. In most cases, CDS can achieve the best prediction effect, but the best effect of the DMVM model using CIIL can reach 85.29%, which is about 18% higher than local training without updating model weight. Table IV shows the accuracy performance of increasing the amount of data for each participant. The accuracy of the local training without weight update and FedAvg training is increasing at the same time. When the amount of data for each party is small (data<1000), the improvement effect of FedAvg compared with local training can reach up to 16.7%. When the amount of data for each participant is large enough (data=3000), the FedAvg enhancement effect is up to 10.5%, which is a small difference from the result of CIIL.

TABLE III
ACCURACY PERFORMANCE OF THE IID EXPERIMENTS I. WE SHOW THE BEST RESULTS WITH BOLDFACE.

Number of party	Metrics	DNN	DFM	DMVM
4	Local Training	74.31±1.96	75.14±0.84	73.43±0.33
	CDS	**82.21±2.15**	81.44±1.03	79.53±0.63
	FedAVG	81.14±0.91	**82.55±1.22**	80.95±0.80
	IIL	79.09±0.93	78.51±0.96	77.81±0.55
	CIIL	81.12±1.25	81.07±1.13	**80.98±0.94**
8	Local Training	73.44±0.61	73.37±0.91	72.67±0.72
	CDS	**82.94±0.28**	**83.66±1.25**	**82.95±0.20**
	FedAVG	82.83±1.23	82.66±0.65	81.56±1.31
	IIL	78.99±0.90	77.74±1.08	77.51±1.85
	CIIL	82.66±0.68	82.88±1.00	81.84±2.73
12	Local Training	74.06±0.52	74.44±0.44	72.38±0.55
	CDS	83.52±0.97	**85.26±0.90**	83.09±0.38
	FedAVG	**84.62±0.56**	83.04±0.84	82.87±1.80
	IIL	79.71±0.60	78.27±1.70	77.03±1.34
	CIIL	83.24±2.42	83.60±1.44	**84.09±1.49**
16	Local Training	73.56±0.22	73.79±0.87	72.48±0.75
	CDS	**84.63±1.44**	**85.07±0.52**	84.78±1.04
	FedAVG	83.83±1.23	83.88±0.65	82.81±1.31
	IIL	78.84±0.48	80.39±0.99	78.27±1.12
	CIIL	83.52±1.80	83.58±2.56	**85.19±1.98**
24	Local Training	73.88±0.55	74.55±0.52	72.40±0.52
	CDS	**86.42±0.54**	**85.97±0.50**	84.79±1.28
	FedAVG	85.13±0.53	84.29±0.98	83.74±1.10
	IIL	79.45±2.92	81.44±0.72	79.40±1.70
	CIIL	84.93±0.47	84.87±0.94	**85.29±0.83**

4) Discussion: As shown in Table III, when the amount of data held by each party is constant, we can find that CDS can always maintain the best effect on DNN and DFM models in most cases, but in the DMVM model, CIIL has the best result. We can see from Table IV that when the amount of data is 1000, the data of each party is not repeated, and the federated learning framework has achieved the best results under the three models. When the amount of data is 1500, the model is affected by repeated data, the prediction performance of FedAVG declined slightly, but the prediction effect of CIIL is still rising. Since the prediction accuracy of CIIL mostly depends on the effect of the last trained party model, we

TABLE IV
ACCURACY PERFORMANCE OF THE IID EXPERIMENTS II. WE SHOW THE BEST RESULTS WITH BOLDFACE.

Number of data	Metrics	DNN	DFM	DMVM
100	Local Training	64.20±0.57	62.30±1.79	61.80±3.40
	CDS	**74.39±0.48**	**73.66±1.17**	**73.19±0.66**
	FedAVG	71.76±1.76	71.72±1.93	70.26±2.01
	IIL	62.95±1.70	67.22±2.41	61.26±3.71
	CIIL	69.58±2.44	73.04±1.43	70.43±1.77
500	Local Training	68.78±1.20	68.94±0.66	68.28±0.37
	CDS	**79.82±1.65**	78.36±0.42	**78.11±0.85**
	FedAVG	78.65±0.77	80.44±0.32	78.10±1.21
	IIL	73.89±2.92	72.72±2.57	72.44±0.73
	CIIL	76.12±0.72	**80.58±1.73**	77.71±0.42
1000	Local Training	71.92±1.39	72.83±1.42	71.29±0.50
	CDS	82.04±1.78	80.98±1.04	81.51±0.85
	FedAVG	**83.09±1.54**	**82.59±0.45**	**82.67±1.02**
	IIL	76.47±2.35	77.07±0.59	76.42±1.46
	CIIL	82.66±0.51	81.83±1.45	81.65±0.61
1500	Local Training	72.67±0.61	73.37±0.91	73.44±0.72
	CDS	**82.94±0.28**	**83.66±1.25**	**82.95±0.20**
	FedAVG	82.83±1.23	82.66±0.65	81.56±1.31
	IIL	78.99±0.90	77.74±1.08	77.51±1.85
	CIIL	82.79±0.68	82.88±1.00	81.84±2.73
2000	Local Training	75.16±1.03	75.58±0.61	73.95±0.70
	CDS	83.31±1.70	**84.90±1.24**	83.49±0.48
	FedAVG	**84.00±0.85**	83.59±1.52	83.03±1.08
	IIL	80.02±1.95	79.20±1.26	79.68±0.70
	CIIL	83.43±0.37	84.70±1.61	**83.74±1.29**
3000	Local Training	77.26±0.98	78.72±0.55	75.91±0.58
	CDS	83.77±0.46	**86.12±1.32**	**85.03±0.35**
	FedAVG	84.32±1.12	84.30±0.49	83.90±1.38
	IIL	81.45±2.35	81.30±1.19	80.18±0.21
	CIIL	**84.73±2.62**	84.43±1.12	85.01±1.11

guess that repeated input data will seriously affect the fusion interaction mode of the multi-view machine layer. Compared with CDS, the last participation of CIIL has less repeated data, and the federated framework will be affected by repeated data due to the last epoch of global model updates. Therefore, CIIL has a stronger anti-interference ability, and the best accuracy result can eventually reach 85.29%. As shown in Table IV, when the number of data is 1500, since the amount of data owned by each party exceeds the total amount of data, each user has duplicate data, which leads to a slight decrease of accuracy. When the amount of data owned by each party is 1000, the data owned by each participant is unique at this time, and the improvement effect of federated training is about 15%, which is the best performance in all experiments. Table II shows if each participant can average divide into the total dataset, DMVM can achieve the best result of 86.95%. At the same time, it can be seen from Table III that when the data owned by the participants is duplicated, the prediction effect of DMVM is the worst, and DFM can still maintain a better prediction performance, but it is slightly lower than DNN. We consider that repeated input data will seriously affect the fusion interaction mode of the multi-view machine layer.

D. Non-IID Experiments

In the real medical environment, the data owned by the hospital should be non-IID. In this subsection we introduce the methods and results of non-IID experiments.

1) Compared Methods: In the real medical environment, the data owned by the hospital should be non-IID. In this subsection, we introduce the methods and results of non-IID experiments. The models we compared are shown in Sec. IV-C1. For non-IID settings, we totally have 8 normal users' personal data, 6 bipolar I disorder patients data, 6 bipolar II disorder patients data. There are 4 hospitals participating in the training experiment, and each hospital has two normal users data, one bipolar I disorder patients data, and one bipolar II disorder patients data. Due to the amount of data generated by patients is different, so the amount of data owned by hospitals is also inconsistent.

TABLE V
ACCURACY PERFORMANCE OF NON-IID EXPERIMENT AND IID. WE SHOW THE BEST RESULTS WITH BOLDFACE.

Types of data	Metrics	DNN	DFM	DMVM
non-IID	CDS	**83.93±1.01**	**82.28±1.04**	**83.18±0.30**
	FedAVG	76.95±1.66	71.59±2.97	76.84±1.74
	IIL	68.81±3.04	68.81±0.98	70.32±0.23
	CIIL	73.16±2.21	74.16±1.53	76.51±1.45
IID	CDS	**82.21±2.15**	81.44±1.03	79.53±0.63
	FedAVG	81.14±0.91	**82.55±1.22**	80.95±0.80
	IIL	79.09±0.93	78.51±0.96	77.81±0.55
	CIIL	81.12±1.25	81.07±1.13	**80.98±0.94**

2) Evaluation criteria: Our evaluation criteria are consistent with Sec. IV-C2, and accuracy is still used as the criterion for evaluating mood prediction.

3) Experiment Result: As shown in Table V, the prediction accuracy of CDS under the non-IID setting is far higher than the distributed cooperative learning method, and the federated learning prediction accuracy of the three models decreased by 5.2% (DNN), 13.3% (DFM), and 5.1% (DMVM). We analyze that the nature of the extreme distribution of non-IID data is the reason for the decline in prediction effect. We also find that the prediction effects of the two models that do not use nonlinear functions for feature interaction are significantly different under the non-IID setting. Due to the large difference in the number of patient data owned by each party and the completely different patient data types, the second-order feature interaction fails to integrate all features well, and the log also shows that its prediction accuracy fluctuates more than DMVM.

4) Discussion: For non-IID experiments, CDS can still maintain about 83% prediction accuracy, while federated learning and CIIL both show different degrees of accuracy drop. However, under the non-IID setting, the federated learning framework surpasses CIIL on the DMVM model, which proves the superiority of the multi-view machine layer under the federated framework. At the same time, for the non-IID experiment, we also find that the accuracy of the validation set is distributed between 50% and 75% during the training process in the first hospital, while the training logs of other hospitals show that the accuracy of the validation set is almost above 90% after each epoch of local training. Furthermore, in order to test the influence of different views on the model prediction accuracy, as shown in Fig 3, we visualize the data of each view. We find that the distribution of Spec. is too scattered, and it is difficult to distinguish normal people from patients in special operations such as backspace, space, and keyboard switching. Alph. and Accel. have better categorizable results from a single view. These also illustrate from the other hand that there are obvious differences in typing patterns between normal people and depressed patients, including the duration of keystrokes. And judging from the distribution of accelerometer values, the way depression patients use mobile phones is also different. In summary, it is necessary to merge data from different views as input.

V. Conclusion

Due to the fact that the patient's medical data must be kept strictly confidential, the limitation of sharing data has led to the problem of data islands, and federated learning plays a key role in solving the problem of data islands. In this work, we use the data records generated by the user when typing on the mobile phone and the user's HDRS score to predict depression through the DeepMood architecture. For IID data, with different amounts of data, the accuracy of federated learning is about 10%-15% higher than that of local training without weight update. For non-IID data, accuracy is only reduced by 13% at most. In order to protect the privacy of patients, the slight decline of model accuracy is completely acceptable. However, we have not yet dealt with the weights of participants with poor performance in the training. Our next work is to consider constructing an appropriate incentive mechanism to weaken the influence of participants with poor contribution on the overall prediction effect, so as to fully reduce the influence of non-IID data on the model.

References

[1] WHO, *The global burden of disease: 2004 update*. World Health Organization, 2008.

[2] M. Hamilton, "The hamilton rating scale for depression," in *Assessment of depression*. Springer, 1986, pp. 143–152.

[3] R. C. Young, J. T. Biggs, V. E. Ziegler, and D. A. Meyer, "A rating scale for mania: reliability, validity and sensitivity," *The British journal of psychiatry*, vol. 133, no. 5, pp. 429–435, 1978.

[4] A. Grünerbl, A. Muaremi, V. Osmani, G. Bahle, S. Oehler, G. Tröster, O. Mayora, C. Haring, and P. Lukowicz, "Smartphone-based recognition of states and state changes in bipolar disorder patients," *JBHI*, vol. 19, no. 1, pp. 140–148, 2014.

[5] R. S. McGinnis, E. W. McGinnis, J. Hruschak, N. L. Lopez-Duran, K. Fitzgerald, K. L. Rosenblum, and M. Muzik, "Wearable sensors and machine learning diagnose anxiety and depression in young children," in *2018 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*. IEEE, 2018, pp. 410–413.

[6] A. M. Darcy, A. K. Louie, and L. W. Roberts, "Machine learning and the profession of medicine," *Jama*, vol. 315, no. 6, pp. 551–552, 2016.

[7] Y. Cao, H. Peng, and S. Y. Philip, "Multi-information source hin for medical concept embedding," *Proceedings of the PAKDD*, vol. 12085, p. 396, 2020.

[8] Z. Liu, X. Li, H. Peng, L. He, and S. Y. Philip, "Heterogeneous similarity graph neural network on electronic health records," in *Proceedings of the IEEE Big Data*, 2020, pp. 1196–1205.

[9] T. Wang, Z. Cao, S. Wang, J. Wang, L. Qi, A. Liu, M. Xie, and X. Li, "Privacy-enhanced data collection based on deep learning for internet of vehicles," *IEEE TII*, 2019.

[10] M. Hao, H. Li, X. Luo, G. Xu, H. Yang, and S. Liu, "Efficient and privacy-enhanced federated learning for industrial artificial intelligence," *IEEE TII*, vol. 16, no. 10, pp. 6532–6542, 2019.

[11] W. Li, F. Milletarì, D. Xu, N. Rieke, J. Hancox, W. Zhu, M. Baust, Y. Cheng, S. Ourselin, M. J. Cardoso *et al.*, "Privacy-preserving federated brain tumour segmentation," in *Proceedings of the MLMI*. Springer, 2019, pp. 133–141.

[12] M. A. Serhani, H. T El Kassabi, H. Ismail, and A. Nujum Navaz, "Ecg monitoring systems: Review, architecture, processes, and key challenges," *Sensors*, vol. 20, no. 6, p. 1796, 2020.

[13] J. Que, X. Jiang, and L. Ohno-Machado, "A collaborative framework for distributed privacy-preserving support vector machine learning," in *AMIA Annual Symposium Proceedings*, vol. 2012. American Medical Informatics Association, 2012, p. 1350.

[14] S. Che, H. Peng, L. Sun, Y. Chen, and L. He, "Federated multi-view learning for private medical data integration and analysis," *arXiv preprint arXiv:2105.01603*, 2021.

[15] Z. Ji, Z. C. Lipton, and C. Elkan, "Differential privacy and machine learning: a survey and review," *arXiv preprint arXiv:1412.7584*, 2014.

[16] H. B. McMahan, E. Moore, D. Ramage, and B. A. y Arcas, "Federated learning of deep networks using model averaging. corr abs/1602.05629 (2016)," *arXiv:1602.05629*, 2016.

[17] T. S. Brisimi, R. Chen, T. Mela, A. Olshevsky, I. C. Paschalidis, and W. Shi, "Federated learning of predictive models from federated electronic health records," *International journal of medical informatics*, vol. 112, pp. 59–67, 2018.

[18] S. Silva, B. A. Gutman, E. Romero, P. M. Thompson, A. Altmann, and M. Lorenzi, "Federated learning in distributed medical databases: Meta-analysis of large-scale subcortical brain data," in *2019 IEEE 16th international symposium on biomedical imaging (ISBI 2019)*. IEEE, 2019, pp. 270–274.

[19] H. Song, G. A. Fink, and S. Jeschke, *Security and privacy in cyber-physical systems: Foundations, principles, and applications*. John Wiley & Sons, 2021.

[20] D. Rawat, C. Brecher, H. Song, and S. Jeschke, "Industrial internet of things: Cybermanufacturing systems," *Cham, Switzerland: Springer*, 2017.

[21] E. Agu, P. Pedersen, D. Strong, B. Tulu, Q. He, L. Wang, and Y. Li, "The smartphone as a medical device: Assessing enablers, benefits and challenges," in *proceeding of the IEEE IoT-NC*, 2013, pp. 48–52.

[22] F. Hussain, J. P. Stange, S. A. Langenecker, M. G. McInnis, J. Zulueta, A. Piscitello, B. Cao, H. Huang, S. Y. Philip, P. Nelson *et al.*, "Passive sensing of affective and cognitive functioning in mood disorders by analyzing keystroke kinematics and speech dynamics," in *Digital Phenotyping and Mobile Sensing*. Springer, 2019, pp. 161–183.

[23] B. Cao, L. Zheng, C. Zhang, P. S. Yu, A. Piscitello, J. Zulueta, O. Ajilore, K. Ryan, and A. D. Leow, "Deepmood: modeling mobile phone typing dynamics for mood detection," in *Proceedings of the ACM KDD*, 2017, pp. 747–755.

[24] C. Xu, D. Tao, and C. Xu, "A survey on multi-view learning," *arXiv:1304.5634*, 2013.

[25] B. Cao, H. Zhou, G. Li, and P. S. Yu, "Multi-view machines," in *Proceedings of ACM WSDM*, 2016, pp. 427–436.

[26] H. Yao, F. Wu, J. Ke, X. Tang, Y. Jia, S. Lu, P. Gong, J. Ye, and Z. Li, "Deep multi-view spatial-temporal network for taxi demand prediction," *arXiv:1802.08714*, 2018.

[27] Y. Shen, H. Zhang, Y. Fan, A. P. W. Lee, and L. Xu, "Smart health of ultrasound telemedicine based on deeply-represented semantic segmentation," *IEEE IoT-J*, 2020.

[28] Y. Nan, W. Li, F. Lu, C. Luo, J. Li, and A. Zomaya, "Developing practical multi-view learning for clinical analytics in p4 medicine," *IEEE Transactions on Emerging Topics in Computing*, 2021.

[29] S. A. Rokni and H. Ghasemzadeh, "Plug-n-learn: automatic learning of computational algorithms in human-centered internet-of-things applications," in *Proceeding of the IEEE DAC*. IEEE, 2016, pp. 1–6.

[30] A. Serra, P. Galdi, and R. Tagliaferri, "Multiview learning in biomedical applications," in *Artificial Intelligence in the Age of Neural Networks and Brain Computing*. Elsevier, 2019, pp. 265–280.

[31] J. Zhang, B. Cao, S. Xie, C.-T. Lu, P. S. Yu, and A. B. Ragin, "Identifying connectivity patterns for brain diseases via multi-side-view guided deep architectures," in *Proceedings of the SDM*. SIAM, 2016, pp. 36–44.

[32] Q. Wu, H. Wu, X. Zhou, M. Tan, Y. Xu, Y. Yan, and T. Hao, "Online transfer learning with multiple homogeneous or heterogeneous sources," *IEEE TKDE*, vol. 29, no. 7, pp. 1494–1507, 2017.

[33] Y. Kim, J. Sun, H. Yu, and X. Jiang, "Federated tensor factorization for computational phenotyping," in *Proceedings of the ACM KDD*, 2017, pp. 887–895.

[34] J. Lee, J. Sun, F. Wang, S. Wang, C.-H. Jun, and X. Jiang, "Privacy-preserving patient similarity learning in a federated environment: development and analysis," *JMIR*, vol. 6, no. 2, p. e20, 2018.

[35] L. Huang, A. L. Shea, H. Qian, A. Masurkar, H. Deng, and D. Liu, "Patient clustering improves efficiency of federated machine learning to predict mortality and hospital stay time using distributed electronic medical records," *JBI*, vol. 99, p. 103291, 2019.

[36] A. Flanagan, W. Oyomno, A. Grigorievskiy, K. E. Tan, S. A. Khan, and M. Ammad-Ud-Din, "Federated multi-view matrix factorization for personalized recommendations," *arXiv:2004.04256*, 2020.

[37] M. Huang, H. Li, B. Bai, C. Wang, K. Bai, and F. Wang, "A federated multi-view deep learning framework for privacy-preserving recommendations," *arXiv:2008.10808*, 2020.

[38] Y. Kang, Y. Liu, and T. Chen, "Fedmvt: Semi-supervised vertical federated learning with multiview training," *arXiv:2008.10838*, 2020.

[39] L. Lyu, H. Yu, X. Ma, L. Sun, J. Zhao, Q. Yang, and P. S. Yu, "Privacy and robustness in federated learning: Attacks and defenses," *arXiv preprint arXiv:2012.06337*, 2020.

[40] R. Canetti, U. Feige, O. Goldreich, and M. Naor, "Adaptively secure multi-party computation," in *Proceedings of the ACM STOC*, 1996, pp. 639–648.

[41] B. Jiang, J. Li, G. Yue, and H. Song, "Differential privacy for industrial internet of things: opportunities, applications and challenges," *IEEE IoT-J*, 2021.

[42] H. Peng, H. Li, Y. Song, V. Zheng, and J. Li, "Federated knowledge graphs embedding," *arXiv preprint arXiv:2105.07615*, 2021.

[43] L. Sun and L. Lyu, "Federated model distillation with noise-free differential privacy," *arXiv preprint arXiv:2009.05537*, 2020.

[44] L. Sun, J. Qian, X. Chen, and P. S. Yu, "Ldp-fl: Practical private aggregation in federated learning with local differential privacy," *arXiv preprint arXiv:2007.15789*, 2020.

[45] A. Acar, H. Aksu, A. S. Uluagac, and M. Conti, "A survey on homomorphic encryption schemes: Theory and implementation," *ACM Computing Surveys (CSUR)*, vol. 51, no. 4, pp. 1–35, 2018.

[46] S. Rendle, "Factorization machines with libfm," *ACM TIST*, vol. 3, no. 3, pp. 1–22, 2012.

[47] T. Tieleman and G. Hinton, "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude," *COURSERA: Neural networks for machine learning*, vol. 4, no. 2, pp. 26–31, 2012.

[48] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-iid data," *arXiv:1806.00582*, 2018.

[49] M. J. Sheller, B. Edwards, G. A. Reina, J. Martin, S. Pati, A. Kotrotsou, M. Milchenko, W. Xu, D. Marcus, R. R. Colen *et al.*, "Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data," *Scientific reports*, vol. 10, no. 1, pp. 1–12, 2020.

[50] K. Chang, N. Balachandar, C. Lam, D. Yi, J. Brown, A. Beers, B. Rosen, D. L. Rubin, and J. Kalpathy-Cramer, "Distributed deep learning networks among institutions for medical imaging," *JAMIA*, vol. 25, no. 8, pp. 945–954, 2018.

Xiaohang Xu is currently a master student at the School of Cyber Science and Technology in Beihang University. He received his Bachelor degree in School of Control and Computer Engineering from North China Electric Power University, Beijing, China, in 2019. His research interests include data privacy, federated learning and reinforcement learning.

Hao Peng is currently an Assistant Professor at the School of Cyber Science and Technology, and Beijing Advanced Innovation Center for Big Data and Brain Computing in Beihang University. His research interests include deep Learning, data mining, reinforcement learning, and federated learning.

Md Zakirul Alam Bhuiyan is currently an assistant professor with Department of Computer and Information Sciences, Fordham University, US. His research interests include IoT/CPS applications, machine learning. He mainly focuses on big data, cloud, networked and distributed sensing systems. He has published more than 130 research articles in top conferences and journals like COMMAG, TDSC, TC, TPDS, TII.

Zhifeng Hao received his B.S. degree in Mathematics from the Sun Yat-Sen University in 1990, and his Ph.D. degree in Mathematics from Nanjing University in 1995. He is currently a Professor in the Department of Mathematics, College of Science, Shantou University. His research interests involve various aspects of algebra, machine learning, data mining, evolutionary algorithms.

Lianzhong Liu is currently a professor with the School of Cyber Science and Technology in Beihang University. His research directions include social network analysis, media mining and information system modeling. He has published more than 90 research articles in top conferences and journals.

Lichao Sun is currently an Assistant Professor in the Department of Computer Science and Engineering at Lehigh University. His research interests include deep learning and data mining. He mainly focuses on security and privacy, social network and natural language processing applications. He has published more than 15 research articles in top conferences and journals like KDD, WSDM, TII, TMC, AAAI.

Lifang He is currently an Assistant Professor in the Department of Computer Science and Engineering at Lehigh University. Before her current position, Dr. He worked as a postdoctoral researcher in the Department of Biostatistics and Epidemiology at the University of Pennsylvania. Her current research interests include machine learning, data mining, tensor analysis, with major applications in biomedical data and neuroscience.