

(RESEARCH ARTICLE)



Graph-Aware Hierarchical Federated Intelligence with Adaptive Differential Privacy for Robust Non-IID Spectrum Sensing in Next-Generation Wireless Networks

Nivedan Suresh ¹, Supraja Ayyamgari ², Charan Thumma ³ and Abhignan Srivatsava Srihashyam ^{4,*}

¹ Independent Researcher Staff Software Engineer, Visa Inc., Austin, TX, USA.

² Independent Researcher Scholar, VA, USA

³ Independent Researcher, Stone ridge, VA, USA

⁴ Senior Software Engineer, Target Corporation, Lakeville, MN, US

* Corresponding Author

ORCID Details

Nivedan Suresh: <https://orcid.org/0009-0001-8186-8690>

Supraja Ayyamgari: <https://orcid.org/0009-0005-0447-4363>

Charan Thumma: <https://orcid.org/0009-0004-6375-4982>

Abhignan Srivatsava Srihashyam: <https://orcid.org/0009-0006-1197-2982>

GSC Advanced Research and Reviews, 2024, 20(03), 340–359

Publication History: Received on 04 August 2024; revised on 21 September 2024; accepted on 24 September 2024

Article DOI: <https://doi.org/10.30574/gscarr.2024.20.3.0333>

Copyright © 2024 Author(s) retain the copyright of this article. This article is published under the terms of the Creative Commons Attribution License 4.0

Abstract

The rapid evolution of AI-native wireless networks has created an increasing demand for scalable, privacy-preserving, and intelligent spectrum sensing frameworks capable of operating effectively under highly heterogeneous and Non-Independent and Identically Distributed (Non-IID) data environments. Conventional federated learning-based spectrum sensing approaches often experience model divergence, excessive communication overhead, privacy-utility trade-offs, and reduced sensing accuracy in dynamic wireless scenarios. To address these challenges, this paper proposes GHFI-Net (Graph-Aware Hierarchical Federated Intelligence Network), a novel collaborative spectrum sensing framework that integrates Graph Neural Networks (GNNs), hierarchical federated learning, adaptive differential privacy, and multi-agent intelligence to enhance distributed spectrum analytics. The proposed framework introduces a graph-aware client clustering mechanism that dynamically captures spatial, temporal, and spectral relationships among distributed wireless nodes, enabling Graph Attention Networks (GATs) and GraphSAGE models to learn robust representations of spectrum occupancy. An adaptive privacy management module allocates personalized differential privacy budgets according to client reliability, data quality, and heterogeneity, thereby improving the privacy-utility balance while minimizing unnecessary perturbations. In addition, hierarchical aggregation, knowledge distillation, and multi-agent collaborative decision-making significantly reduce communication costs, accelerate model convergence, and improve scalability across edge, fog, and cloud computing layers. Experimental evaluation demonstrates that GHFI-Net achieves superior spectrum sensing accuracy, faster convergence, enhanced communication efficiency, and stronger privacy preservation compared with existing federated learning methods. The proposed framework provides a scalable, explainable, and privacy-aware foundation for autonomous spectrum management, cognitive radio intelligence, and next-generation AI-native 6G wireless communication systems.

Keywords: AI-native wireless networks, collaborative spectrum sensing, federated learning, Graph Neural Networks (GNNs), Graph Attention Networks (GATs), GraphSAGE, adaptive differential privacy, hierarchical federated learning, Multi-agent systems, Cognitive radio, 6G networks, Privacy-preserving artificial intelligence.

1 Introduction

Radio spectrum is an indispensable and valuable resource for mobile communications. With the rapid development of communication technologies and the explosive growth of Internet of Things (IoT) devices, the global communications industry has an increasingly urgent demand for spectrum resources. Traditional spectrum allocation methods statically allocate spectrum by dividing the wireless spectrum into fixed frequency bands, which are exclusively assigned to users by government regulatory authorities. However, authorization-based static allocation makes it difficult to assign dedicated spectrum to emerging wireless services. Moreover, measurement studies have shown that most licensed frequency bands remain idle for the majority of the time, resulting in low spectrum utilization. The contradiction between spectrum scarcity and low spectrum utilization has become an urgent issue to be addressed [1]. Spectrum sharing is considered an important approach to improving spectrum utilization and resource allocation efficiency. It allows different users to share the same spectrum resources at different times. In simple terms, it enables unlicensed users, referred to as secondary users (SUs), to dynamically access idle frequency bands that are not currently occupied by licensed users, referred to as primary users (PUs), thereby significantly improving spectrum utilization [2].

To realize dynamic spectrum sharing, a Dynamic Spectrum Access (DSA) system with sensing, prediction, and decision-making capabilities must be established. As shown in Figure 1, a typical DSA system consists of spectrum sensing, spectrum prediction, and channel access strategy modules. Among them, spectrum sensing serves as the foundation of system operation, whose task is to detect primary user signals and identify available idle spectrum bands. Since subsequent spectrum prediction and channel access strategies rely on spectrum sensing results, the accuracy of spectrum sensing directly determines the overall performance and spectrum utilization efficiency of the DSA system. Spectrum sensing technology is evolving from independent single-node detection to collaborative sensing among multiple secondary users, aiming to improve the reliability of spectrum occupancy decisions. Existing Cooperative Spectrum Sensing (CSS) mainly follows two paradigms. One is Centralized Cooperative Spectrum Sensing (CCSS), in which each SU uploads local sensing results to a fusion center where a global decision is made. The other is Distributed Cooperative Spectrum Sensing (DCSS), where SUs exchange sensing information with each other and reach a consensus decision [3].

In both paradigms, collaboration requires the transmission of sensing information. Such information is often derived directly from raw observations or computed statistics such as energy measurements, local decisions, or fine-grained sensing reports. Consequently, sensitive information, such as the spectrum occupancy status of primary users, may be inferred during transmission and sharing, leading to privacy leakage risks [4]. To address privacy and security challenges associated with spectrum sensing data transmission, Federated Learning (FL), as an emerging distributed machine learning paradigm, provides a new solution for spectrum sensing model training. FL enables users to train models locally and only upload model updates to a central server for aggregation, without sharing raw data, thereby providing a certain degree of privacy protection. However, studies have shown that FL itself is vulnerable to privacy leakage. Malicious attackers may infer whether a specific primary user participated in training through membership inference attacks [5], reconstruct the spectrum usage patterns and geographic locations of primary users through model inversion attacks [6], or accurately recover raw spectrum samples from shared gradients through gradient inference attacks [7].

In spectrum sensing scenarios, such attacks may expose sensitive information such as communication frequencies, transmission power levels, and activity patterns of primary users, thereby making them vulnerable to malicious interference, precise tracking, or unauthorized spectrum occupation. Therefore, to prevent the misuse of primary user information, privacy protection mechanisms must be incorporated into federated spectrum sensing. Among privacy-preserving techniques for FL, Differential Privacy (DP) provides a rigorous privacy protection framework. Its key advantage lies in mathematically provable privacy guarantees, whose effectiveness does not depend on an attacker's background knowledge, thereby effectively defending against inference attacks under various adversarial assumptions [8]. However, traditional DP mechanisms generally employ a uniform privacy budget and apply the same level of noise injection to all data regardless of sensitivity. In spectrum sensing scenarios, such a one-size-fits-all strategy fails to reflect the varying privacy risks associated with different spectrum states and may adversely affect sensing reliability. For example, if the noise level is excessively high, the model may incorrectly classify occupied spectrum as idle, causing secondary users to access frequency bands currently used by primary users and increasing interference risks. Conversely, if the noise level is too low, highly sensitive samples may not receive sufficient protection, leading to potential privacy leakage [9]. Therefore, this paper introduces a Personalized Differential Privacy (PDP) mechanism into the federated spectrum sensing framework. Users can select different privacy protection levels according to the sensitivity of their local spectrum data. On one hand, this satisfies personalized privacy requirements; on the other hand,

compared with uniform DP budget allocation, it reduces overall noise injection and improves spectrum sensing accuracy [10]. However, PDP is constrained by the Non-Independent and Identically Distributed (Non-IID) characteristics and label dependency of spectrum sensing data. First, local data collected by secondary users in different regions exhibit significant differences. Secondary users in rural areas or during nighttime periods are more likely to collect “idle” or “low-occupancy” samples, whereas users in urban centers or daytime periods tend to collect more “busy” or “high-occupancy” samples. As a result, data collected within the same physical environment may approximately follow an Independent and Identically Distributed (IID) distribution, while data across different environments exhibit strong Non-IID characteristics.

In FL, Non-IID data cause local model updates to diverge, resulting in slow convergence or even divergence of the global model. Furthermore, privacy budgets in spectrum data are often closely related to labels. For example, in vehicular networks, “high-occupancy” signals may contain sensitive information such as vehicle trajectories and are more susceptible to location inference attacks, requiring stronger privacy protection through larger privacy budgets. In contrast, “low-occupancy” signals often correspond to environmental noise and therefore require less stringent privacy protection. Such label-dependent privacy budget allocation introduces heterogeneous perturbation noise on top of existing Non-IID characteristics, further aggravating model drift and making global model convergence even more difficult. Therefore, there is an urgent need to develop a personalized differential privacy federated learning framework for cooperative spectrum sensing that simultaneously satisfies differentiated privacy requirements, mitigates model drift, and improves classification performance. To address these challenges from the perspective of optimizing spectrum sensing models, this paper proposes a Federated Learning framework integrating Personalized Differential Privacy and a Rebalanced Clustering Strategy, referred to as RebalFL. The objective is to jointly achieve secure spectrum sharing and privacy protection under Non-IID spectrum data environments.

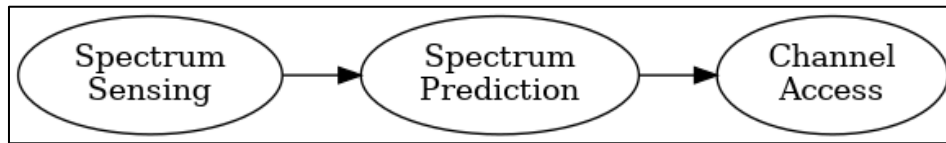


Figure 1: DSA_Architecture

Specifically, this paper first introduces the PDP mechanism, allowing different clients to configure personalized privacy budgets according to their privacy requirements and sample characteristics. This ensures reliable privacy protection while minimizing the impact of noise injection on model performance. Secondly, a rebalanced clustering strategy is designed to partition client data into distribution-balanced clusters, where asynchronous updates are performed within each cluster to alleviate model drift caused by Non-IID data.

The main contributions of this paper are summarized as follows:

- A federated learning framework integrating personalized differential privacy and a rebalanced clustering strategy is proposed to address model drift caused by data label imbalance and label dependency in spectrum sensing tasks. The framework dynamically balances privacy protection and model performance, significantly improving the generalization capability of the global model under Non-IID scenarios.
- A personalized differential privacy mechanism is introduced to systematically prevent privacy leakage during training. Participating devices can flexibly configure privacy protection levels according to their own privacy requirements. While ensuring privacy security, the mechanism effectively reduces overall noise injection and improves model usability.
- A KL-divergence-based rebalanced clustering strategy is designed to partition clients into clusters with balanced data distributions. Distributed training is then performed within each balanced cluster, effectively mitigating model drift caused by Non-IID data and improving robustness in heterogeneous environments.
- Experimental results demonstrate that RebalFL maintains high model accuracy under various Non-IID scenarios and differentiated privacy budget settings. In mildly Non-IID spectrum sensing tasks, RebalFL improves the F1-score by 16.65% compared with PDPFL. Under extremely Non-IID settings on the MNIST dataset, RebalFL significantly outperforms three baseline methods in terms of model accuracy, demonstrating its effectiveness in mitigating model drift while maintaining high predictive performance.

2 Related Work

This section systematically reviews existing studies related to this research, aiming to summarize the development of spectrum sensing from traditional centralized and distributed cooperative spectrum sensing methods to federated learning-based spectrum sensing and privacy protection mechanisms, thereby establishing the foundation for this study.

2.1 Current Status of Cooperative Spectrum Sensing Technologies

The primary objective of spectrum sensing is to extract observable features related to primary user occupancy from received signals and determine the occupancy status of the target frequency band, thereby providing a basis for secondary user access decisions. As multipath fading and shadowing effects in wireless environments become increasingly severe, single-node sensing is prone to missed detections and false alarms. Consequently, research has gradually shifted from independent single-node detection to cooperative spectrum sensing and further evolved into distributed cooperative spectrum sensing, which eliminates the need for a fixed fusion center and allows nodes to exchange information and iteratively reach consensus decisions, thereby improving the robustness and reliability of sensing results.

As a result, a series of classical analytical detectors based on statistical theory have been developed. Their common characteristic is the detection of manually designed statistical features. However, traditional methods have several limitations. For example, Energy Detection (ED), which is based on energy thresholds, does not require prior knowledge but is highly sensitive to noise interference and has difficulty distinguishing noise from legitimate signals [11]. Matched Filtering (MF) achieves optimal performance when prior sequences are known and synchronization conditions are satisfied, but its applicability is limited for unknown or heterogeneous primary users [12]. Cyclostationary Feature Detection (CFD) utilizes cyclic characteristics such as spectral correlation to resist noise and performs effectively under low Signal-to-Noise Ratio (SNR) conditions, but it has high computational complexity and is sensitive to sampling and synchronization errors [13]. Covariance- and eigenvalue-based methods utilize correlation structures to alleviate certain noise uncertainties; however, their information sources remain limited to low-order statistical correlations [14].

Therefore, although traditional methods can characterize explicitly modelable features such as energy, correlation, and cyclic properties, a limited number of statistical measures cannot adequately capture joint time-frequency structures, temporal dependencies, and nonlinear decision boundaries in complex scenarios involving multiple primary users, strong non-stationary interference, evolving occupancy patterns, and hardware imperfections. Consequently, their robustness and transferability remain limited [15]. To improve sensing accuracy in scenarios where precise statistical models are unavailable or wireless channel environments vary significantly, researchers have increasingly turned to data-driven approaches in recent years. Existing studies have extensively introduced machine learning and deep learning methods. These approaches automatically extract high-dimensional discriminative features from massive datasets. For example, Support Vector Machines (SVMs) construct optimal hyperplanes in high-dimensional spaces for binary classification tasks [16]. Convolutional Neural Networks (CNNs) extract local spatial texture features from time-frequency representations [17]. Long Short-Term Memory (LSTM) networks capture long-term temporal dependencies within signal sequences [18]. Overall, the advantages of machine learning methods lie in their ability to characterize complex statistical properties and automate feature extraction processes, making them particularly suitable for scenarios where traditional statistical models cannot accurately represent the underlying data distributions. However, in cooperative spectrum sensing, regardless of whether analytical detectors, machine learning models, or deep learning models are employed, the collaborative process inevitably requires the transmission and sharing of sensing information. In CCSS, secondary users upload raw observation data or decision statistics to the fusion center, whereas DCSS requires neighboring nodes to exchange fine-grained sensing reports. Existing studies have pointed out that such sensing data or decision statistics may be exploited to infer sensitive information, such as the locations of secondary users, thereby introducing privacy leakage risks [19].

2.2 Current Status of Federated Learning-Based Cooperative Spectrum Sensing

Federated Learning (FL) is a distributed machine learning architecture that has been initially applied in wireless communication systems and provides a promising solution to address privacy concerns in cooperative spectrum sensing tasks [20].

Chen et al. [21] addressed the issue of limited training performance caused by insufficient data volume and labeled samples at individual spectrum sensors by proposing an FL-based multi-source cooperative spectrum sensing framework. Through cross-node knowledge sharing, the proposed framework improved detection accuracy. Song et al. [22] proposed a cooperative dynamic spectrum access technique based on FL and multi-agent deep reinforcement learning, allowing multiple users to collaboratively optimize system performance without sharing training data. Shi et al. [23] further demonstrated the feasibility of extending traditional cooperative spectrum sensing into an FL framework, providing new insights for building efficient and privacy-preserving distributed spectrum monitoring systems. However, in practical wireless communication systems, user devices and spectrum data are often Non-Independent and Identically Distributed (Non-IID). To address this challenge, Li et al. [24] introduced a proximal term into the optimization process of FL and proposed a proximal federated learning algorithm. Wang et al. [25] proposed a normalized averaging-based FL algorithm to achieve faster convergence of training errors. Although FL alleviates privacy risks associated with centralized data collection to some extent, it does not inherently provide comprehensive privacy protection mechanisms. Sensitive information may still be exposed during the transmission and aggregation of model parameters.

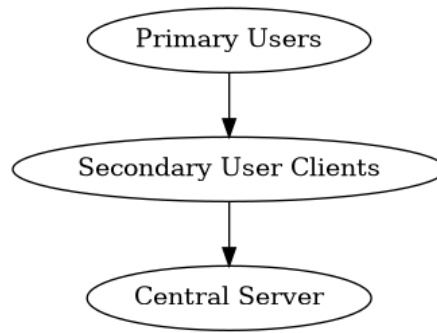


Figure 2: System Model

2.3 Current Status of Differential Privacy-Based Federated Learning

As a rigorous mathematical privacy protection framework, Differential Privacy (DP) has been widely applied in FL to mitigate privacy leakage risks during model training. Traditional DP mechanisms achieve privacy protection by injecting noise into model gradients or parameters before transmission [26], theoretically defending against various inference attacks. However, these methods generally employ a uniform privacy budget, applying the same privacy constraints to all samples [27–28]. This one-size-fits-all strategy not only ignores the diverse privacy requirements of different users but may also impose excessive privacy protection on certain data, resulting in unnecessary noise injection and degraded performance of the global FL model. To address these limitations, researchers have proposed Personalized Differential Privacy (PDP), which allows participants to define personalized privacy budgets according to their privacy preferences. Based on the granularity of privacy budget allocation, existing PDP mechanisms can be categorized into client-level PDP [29] and record-level PDP [30]. The former, such as the client-level PDP scheme proposed by Liu et al. [29], allocates different privacy budgets to different clients to accommodate diverse privacy requirements across institutions. The latter further refines privacy protection to individual data records [30], enabling personalized privacy control for each record and allowing different data samples to possess distinct privacy budgets. Sampling mechanisms play a crucial role in PDP-based FL tasks. Studies have shown that introducing random sampling can effectively prevent catastrophic forgetting caused by differences in privacy budgets [31]. Based on this observation, Boenisch et al. [32] proposed a binary-search-based sampling probability optimization algorithm to determine the optimal sampling probability for each record under a given privacy budget. However, when privacy budgets are continuously distributed or span a wide range, the computational overhead of this algorithm increases significantly, limiting its applicability in large-scale FL scenarios. To address this issue, Liu et al. [30] further proposed a Simulation-Curve Fitting (SCF) algorithm that directly estimates the nonlinear mapping relationship between sampling probabilities and privacy budgets, thereby improving computational efficiency. Although PDP mechanisms can improve the overall utility of FL models, they exhibit significant limitations in scenarios where data are Non-IID and privacy budgets are label-dependent. Existing studies generally assume that client data are independently and identically distributed and fail to consider the realistic situation of label imbalance across clients, which can lead to model drift. Furthermore, PDP mechanisms may amplify these differences, causing classification models to become biased toward majority classes and consequently weakening the overall generalization capability of the global model. Therefore, this paper proposes RebalFL, which introduces a rebalanced clustering strategy to effectively mitigate model drift caused by Non-IID data distributions and label-dependent characteristics.

Table 1: Major Mathematical Symbols and Their Definitions

Symbol	Definition
ϵ	Differential Privacy Budget
δ	Differential Privacy Relaxation Parameter
ω	Local Model Parameters

Symbol	Definition
T	Total Number of Communication Rounds
τ	Number of Local Training Iterations
q	Sampling Probability
σ	Standard Deviation of Gaussian Noise
$KL(\cdot \ \cdot)$	Kullback–Leibler (KL) Divergence
S_0	Central Server
$S = \{S_1, S_2, \dots, S_k\}$	Set of Edge Servers
$C = \{C_1, C_2, \dots, C_i\}$	Set of Clients

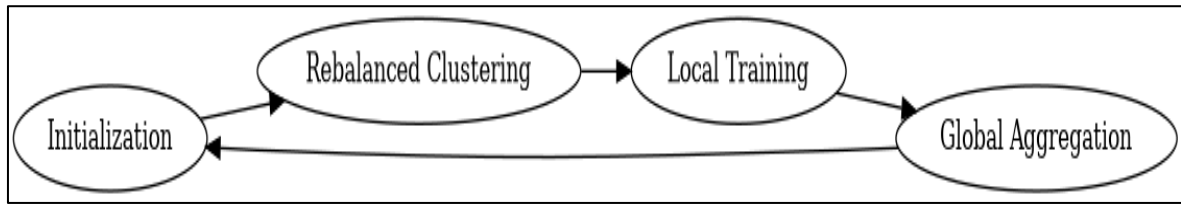


Figure 3: RebalFL Framework

2.4 Preliminary Knowledge

To facilitate subsequent reading and understanding, Table 1 summarizes the main mathematical symbols used in this paper and their meanings.

2.4.1 Machine Learning-Based Spectrum Sensing Model

The core task of spectrum sensing is to determine whether a primary user signal exists in a target frequency band based on the observation signal received by a secondary user. This problem is typically modeled as a binary hypothesis testing problem. Let the discrete received sequence within an observation window be represented as $r[k]_{k=1}^N$. Then, the binary hypothesis testing can be expressed as

$$\{H_0: r[k] = n[k] \quad H_1: r[k] = s[n] + n[k]\}$$

where $n[k]$ denotes additive noise and $s[k]$ denotes the primary user signal. Spectrum sensing makes decisions by constructing a statistical metric $T[r]$ and comparing it with a threshold λ . When $T[r] \geq \lambda$, the spectrum state is determined as occupied H_1 ; otherwise, it is determined as idle H_0 . The corresponding detection performance is usually represented by the detection probability and false alarm probability as

$$P_d = Pr(H_1), \quad P_f = Pr(T(r) \geq \lambda | H_0)$$

In traditional analytical detectors, common statistical metrics include energy statistics and cyclostationary feature statistics. Taking the energy statistic as an example, it can be obtained by aggregating the energy of the received signal in either the frequency domain or time domain, as shown in Eq

$$T_E(r) = \frac{1}{N} \sum_{\{k=1\}}^N |r[k]|^2$$

The key characteristic of such methods is that the design of statistical metrics usually depends on prior assumptions regarding the statistical characteristics of signals and noise, and the threshold setting is highly sensitive to noise uncertainty. To reduce dependence on precise statistical models and manually designed features, machine learning methods have been widely introduced into spectrum sensing in recent years, transforming the “occupied or not”

decision into a supervised classification problem. Specifically, a feature vector x is first constructed from the observation sequence r , where features may include energy, power spectral density, time-frequency representations, and so forth. Subsequently, a classifier $f_\theta(\cdot)$ is trained using the labeled dataset $((x_i, y_i)_{i=1}^M)$, where $y_i = \{0, 1, \dots, i\}$ represents the data label. The training process can be abstracted as minimizing the empirical risk:

$$\min_{\theta} \frac{1}{M} \sum_{i=1}^M l(f_\theta(x_i), y_i)$$

where $l(\cdot)$ denotes the classification loss function. During the inference stage, for a new observation r^* , the corresponding feature vector x^* is calculated and the prediction result $\hat{y} = f_\theta(x^*)$ is output as the occupancy decision.

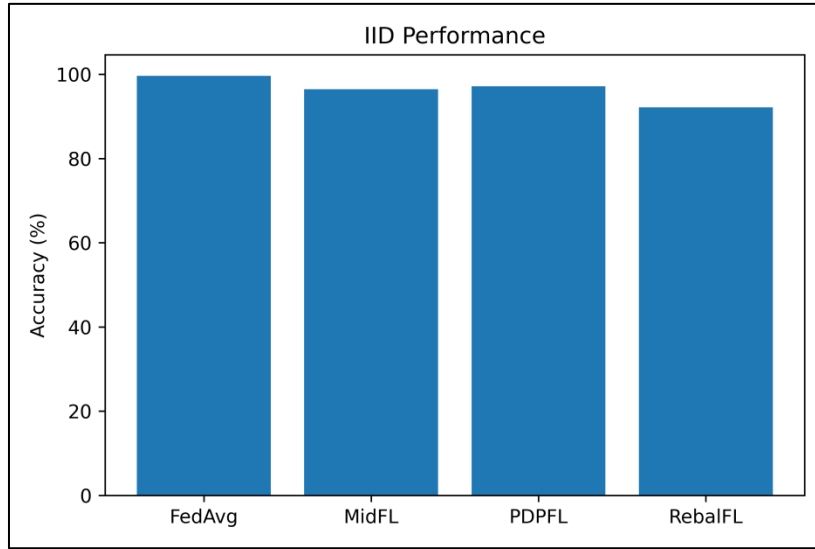


Figure 4: IID

2.5 Federated Learning-Based Cooperative Spectrum Sensing Framework

Spectrum sensing is the core mechanism of Dynamic Spectrum Access (DSA) systems, used to determine whether a specific frequency band is occupied by a primary user, thereby providing reliable spectrum access opportunities for secondary users and improving spectrum resource utilization [33]. However, high-accuracy spectrum sensing models often rely on large amounts of historical data, which are difficult for a single node to collect independently. To overcome this data island problem, Federated Learning (FL), as a distributed machine learning paradigm, provides a new training framework for spectrum sensing. The FL-based spectrum sensing framework consists of one central server and (I) secondary-user clients $C = \{C_1, C_2, \dots, C_I\}$. During the spectrum sensing stage, each secondary-user client collects a local spectrum sensing dataset $D = \{D_1, D_2, \dots, D_I\}$ which is stored locally. During the training stage, the central server coordinates all secondary-user clients to jointly train a global spectrum sensing model $\omega^* \in R^d$. The entire training process follows the standard FL workflow and consists of the following four core steps.

2.6 Model Distribution

At the beginning of the t^{th} training round, the central server broadcasts the current global model parameters ω^t to all participating secondary-user clients.

2.7 Local Training

After receiving the model parameters, each secondary-user client C_i independently trains the model using its local dataset D_i and updates the local model parameters ω_i^t through gradient descent:

$$\omega_i^{t+1} = \omega_i^t - \eta l_i(\omega_i^t; D_i)$$

where η denotes the learning rate, ω_i^t represents the local model of client C_i , and $l_i(\omega_i^t; D_i)$ denotes the local loss function of client C_i .

2.8 Model Upload

After completing local training, each secondary-user client uploads the updated model parameters ω_i^{t+1} to the central server.

2.9 Global Aggregation

After receiving model updates from all participating clients, the central server performs weighted averaging to aggregate the global model and obtain a new round of global model parameters:

$$\omega^{t+1} = \sum_{i=1}^I \frac{n_i}{N} \omega_i^{t+1}$$

where n_i represents the local data size of client C_i , and N denotes the total number of samples across all participating clients. The above process is repeated until the model converges or the predefined number of training rounds is reached. The final global spectrum sensing model ω^* is then obtained and broadcast by the central server to all secondary-user clients for deployment.

2.10 Personalized Differential Privacy Mechanism

Differential Privacy (DP) is a data-driven privacy protection approach. Integrating DP into FL can further reduce privacy leakage risks. Personalized Differential Privacy (PDP) is a variant of traditional DP that aims to provide different levels of privacy protection for each data record in a dataset, thereby preventing attackers from inferring sensitive information from shared model updates. To this end, this section first reviews the basic concepts of DP and then introduces PDP and the noise mechanism used in this paper.

Definition 1: (ϵ, δ) -DP

For a randomized mechanism A and the set of all possible outputs $o \in \text{range}(A)$, given neighboring datasets D and D' , if for any output query the following condition holds, then the randomized mechanism A satisfies (ϵ, δ) – DP [34]:

$$\Pr[A(D) \in o] \leq e^\epsilon \Pr[A(D') \in o] + \delta$$

where $\Pr[A(D) \in o]$ denotes the probability that the output of mechanism A on dataset D falls within set o . The parameter ϵ is the privacy budget and serves as a measure of privacy protection strength. A larger privacy budget indicates weaker privacy protection, whereas a smaller privacy budget indicates stronger privacy protection. The parameter δ is the relaxation parameter representing the probability that mechanism A does not satisfy (ϵ, δ) -DP.

Definition 2: $(\epsilon_{i,j}, \delta)$ – PDP

For a randomized mechanism A and the set of all possible outputs $o \in \text{range}(A)$, given neighboring datasets D and D_{-j} , if for any output query the following condition holds, then the randomized mechanism A satisfies $(\epsilon_{i,j}, \delta)$ – PDP [35]:

$$\Pr[A(D) \in o] \leq e^{\epsilon_{i,j}} \Pr[A(D_{-j}) \in o] + \delta$$

Definition 3: Gaussian Mechanism

The Gaussian mechanism protects privacy by adding noise following a Gaussian distribution to the query results. For a given query function (f) , the output on dataset (D) can be expressed as

$$M(D) = f(D) + N(0, \sigma^2)$$

where $N(0, \sigma^2)$ is the noise term following a Gaussian distribution with mean 0 and variance σ^2 . To satisfy (ϵ, δ) – PDP, an appropriate value of σ must be selected. A common setting is

$$\sigma = \sqrt{2 \ln \ln \left(\frac{1}{\delta} \right) \frac{\Delta f}{\varepsilon}}$$

where Δf denotes the sensitivity of the function.

2.11 Problem Definition

2.11.1 System Model

This paper considers a multi-region wireless spectrum sensing scenario. Within each region, a large number of Primary Users (PUs) are authorized to communicate over different frequency bands according to their service requirements, continuously generating real spectrum signals. To perceive spectrum occupancy states, several spectrum sniffing nodes are deployed as Secondary User (SU) clients, which are responsible for local spectrum sensing model training. All secondary users communicate with a cloud-based central server through wireless networks. The central server is responsible for coordinating the training and distribution of the global spectrum sensing model in the Federated Learning (FL) framework.

The architecture of the FL-based cooperative spectrum sensing system is illustrated in Fig. 2. The system consists of three types of entities: Primary Users, Secondary User Clients, and a Central Server. The functions of each entity are summarized as follows.

Primary Users: Primary users are the authorized holders of spectrum bands. Their activity data are collected and labeled by secondary user clients through spectrum sensing technologies and serve as the original dataset for spectrum sensing model construction.

Secondary User Clients: Each secondary user client collects spectrum sensing data through local sensors. During the FL process, clients receive the global model from the central server, perform local training tasks, and upload only local model updates to the central server, ensuring that raw data never leave the local device.

Central Server: The central server is responsible for initializing and aggregating the global model. During each training iteration, the central server aggregates local model updates received from multiple secondary user clients and updates the global model using algorithms such as Federated Averaging (FedAvg).

2.12 Threat Model

This paper assumes that both secondary user clients and the central server are honest-but-curious participants. This means that they strictly follow the predefined training protocol but may attempt to infer private information from shared data during the learning process [36]. In the FL framework, secondary user clients can access the global model updates generated in each training round. By analyzing the received global model, a client may attempt to infer sensitive information belonging to other participants, thereby launching membership inference attacks [37]. As the maintainer of the global model, the central server has access to model updates from all participating clients. By analyzing these model updates, the server may attempt to reconstruct or infer the local data of specific clients [38].

Design Objectives

This paper focuses on cooperative spectrum sensing scenarios, with the following design objectives.

Privacy Protection: To address the diverse privacy protection requirements of different primary users, the framework should allow different data samples to be assigned personalized privacy budgets according to user preferences. The proposed scheme must be theoretically proven to satisfy Personalized Differential Privacy (PDP) and provide rigorous privacy guarantees.

Mitigation of Model Drift in Non-IID Scenarios: The framework should effectively prevent model drift caused by imbalanced data distributions or heterogeneous privacy protection strengths, thereby avoiding classification bias toward specific categories and improving model robustness in Non-IID environments.

No Accuracy Degradation Under Privacy Protection: After introducing the PDP mechanism, the framework should ensure that the global model does not suffer significant performance degradation due to noise perturbation. Its final performance should outperform traditional Differential Privacy (DP)-based approaches while maintaining the reliability and accuracy of spectrum sensing model outputs.

3 Proposed Method

To achieve the aforementioned objectives, this paper proposes a Federated Learning framework that integrates Personalized Differential Privacy (PDP) and a Rebalanced Clustering Strategy, referred to as RebalFL. Under the premise of providing personalized privacy protection, RebalFL effectively mitigates model drift caused by the Non-IID characteristics of distributed data. Consequently, the global model trained through the federated learning process can achieve high classification accuracy while preserving user privacy.

3.1 Overview of the Proposed Framework

The overall architecture of RebalFL is illustrated in Fig. 3 and consists of three core entities: a Central Server, multiple Edge Servers, and Secondary User (SU) Clients. The execution of RebalFL is an iterative process that can be divided into the following four stages.

System Initialization: The central server initializes the global model parameters, constructs and broadcasts a sampling probability estimator, and each secondary user client performs local privacy budget allocation and sampling probability computation.

Client Rebalanced Clustering: Based on the data distribution of participating clients, the central server calculates the Kullback–Leibler (KL) divergence among clients and constructs multiple client clusters with balanced data distributions. An edge server is assigned to manage each cluster.

Local Balanced Model Training: Within each cluster, the corresponding edge server organizes clients to perform asynchronous chain-based model training and transmission. During local training, each client applies the PDP protection mechanism. After training, the edge server submits the locally balanced model to the central server.

Global Model Aggregation and Update: The central server collects model updates from all edge servers and aggregates them using the Federated Averaging (FedAvg) algorithm to generate a new global model. The process then returns to the first stage and iterates until model convergence or a predefined stopping criterion is reached.

3.2 System Initialization

The system initialization stage is jointly completed by the central server and secondary user clients, providing the foundation for privacy preservation and efficient model training. This stage mainly consists of central server initialization and client initialization. **Central Server Initialization:** During the central server initialization phase, the parameters of the global spectrum sensing model are first initialized. Subsequently, the central server constructs a sampling probability estimator using a Simulation-Curve Fitting (SCF) algorithm. Through numerical simulation, the server investigates the upper bound of the actual privacy cost that a single data record can achieve after completing a full federated training process under different sampling probabilities. Simulation results indicate that the relationship between the privacy budget and the sampling probability approximately follows an exponential function: $\varepsilon \approx e^{aq+b} + c$. Based on this relationship, the sampling probability estimator is derived as $F(\varepsilon) = q = \frac{\ln(\varepsilon - c) - b}{a}$ where $a \neq 0, \varepsilon \in (0, 10], c \in (0, \varepsilon)$. The sampling probability estimator enables the direct computation of the corresponding optimal sampling probability for any given privacy budget. **Client Initialization:** During the client initialization phase, each client C_i first assigns a personalized privacy budget $\varepsilon_{i,j}$ to every local spectrum sensing record $d_{i,j}$ according to its data sensitivity level or specific privacy requirements. Subsequently, each client utilizes the sampling probability estimator to calculate the corresponding optimal sampling probability $q_{i,j}$ for every spectrum sensing record in its local dataset. This initialization process establishes the foundation for personalized privacy protection while ensuring efficient and privacy-aware federated training in subsequent stages.

3.3 Client Rebalanced Clustering

Due to the significant differences in signal environments experienced by secondary user clients located in different geographical regions, the label distributions of their sensed data also vary considerably. To mitigate the data imbalance problem caused by such heterogeneity, this section introduces a client rebalanced clustering strategy, where an edge server is assigned to each cluster that approximates the global balanced distribution. The process consists of the following two stages.

Client Data Distribution Upload: In the client data upload stage, each client first locally calculates the category labels and corresponding sample counts within its dataset. Subsequently, Differential Privacy (DP) noise is added to the statistical information. The perturbed data distribution is then uploaded to the central server. As a result, the central server only obtains noise-perturbed distribution information rather than the actual contents of local client datasets.

This approach enables visibility of data distribution characteristics while ensuring that the privacy of the original spectrum sensing data remains protected.

Client Rebalanced Clustering: During the client rebalanced clustering stage, edge server scheduling is employed to construct balanced client clusters, thereby mitigating model drift caused by heterogeneous data distributions across clients. To achieve this objective, the system first introduces multiple edge servers and adopts a greedy allocation strategy to assign clients sequentially. Specifically, after examining the data distributions of all unassigned clients, each edge server selects the client whose inclusion results in a cluster distribution that is closest to a uniform distribution. The optimal selection is determined by minimizing the Kullback–Leibler (KL) divergence between the current cluster distribution and the target uniform distribution, as shown in Line 4 of Algorithm 1. When an edge server reaches its maximum client capacity, the central server creates a new edge server and continues the allocation process until all clients have been assigned.

Algorithm 1: KL Divergence-Based Client Rebalanced Clustering

Input: Client set $C = \{C_1, C_2, \dots, C_I\}$, Edge server set $S = \emptyset$, Global balanced distribution P_u

Output: Edge server set with assigned clients $S = \{S_1, S_2, \dots, S_K\}$

repeat

Create edge server S_k

for ($|C| > 0$) and ($|S_k| < \gamma$) do

$$S_k \leftarrow \underset{C_i \in C}{\operatorname{argmin}} D_{KL}(P_{S_k} + P_{C_i} || P_u),$$

Add client C_i to edge server S_k

Remove client C_i from C

end for

$$S \leftarrow S \cup S_k$$

return S

Assume that the total number of clients is n , and the number of data label categories is L . During the rebalanced clustering process, identifying the optimal client assignment requires traversing the candidate client set and computing KL divergence values. The total number of KL divergence computations is $\sum_{i=1}^n (n - i + 1) = \frac{n(n+1)}{2}$. The computational complexity of a single KL divergence calculation is $O(L)$. Therefore, the overall time complexity of Algorithm 1 is $O(n^2L)$. Since L is typically a very small constant in practical spectrum sensing scenarios, and the clustering algorithm is executed only once during the initialization phase, the overall framework maintains good scalability even when handling large-scale client participation. The current algorithm adopts a locally optimal greedy selection strategy. Under extreme data distribution conditions, it may fail to obtain the globally optimal clustering result. Specifically, clusters constructed earlier may consume critical category samples, making it difficult for subsequent clients to achieve an ideal balanced distribution.

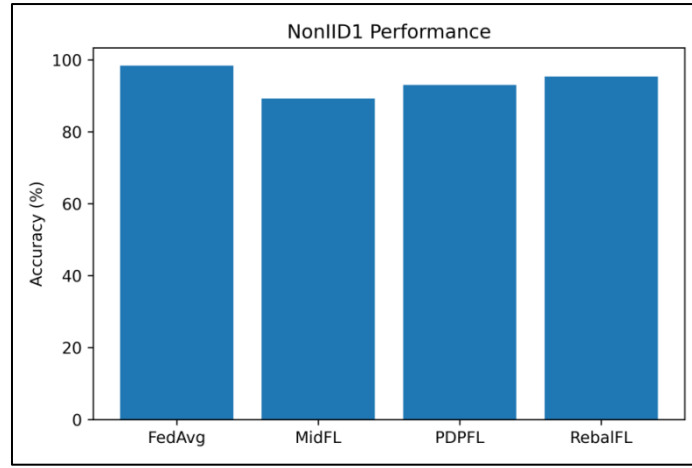


Figure 5: NonIID1

3.4 Local Balanced Model Training

After the clustering process is completed, each edge server organizes model training among the clients within its managed cluster. The model starts from the first client, is sequentially transmitted and updated among clients, and is finally collected by the edge server for local aggregation. This section focuses on the client-side local training process based on Personalized Differential Privacy (PDP), which consists of the following four steps.

3.4.1 Step 1: Model Distribution

The edge server distributes the current cluster-level local model ω_r to the first participating client.

3.4.2 Step 2: Record-Level Poisson Sampling

Before training begins, client C_i independently performs Poisson sampling on each local data record $d_{i,j}$ according to its record-specific sampling probability $q_{i,j}$. Specifically, $\beta_{i,j} \sim \text{Ber}(q_{i,j})$ and the mini-batch training set for the current round is constructed as $D_i = \{d_{i,j} \mid \beta_{i,j} = 1\}$. Through the Poisson sampling mechanism, privacy amplification can be achieved, thereby reducing the overall privacy cost.

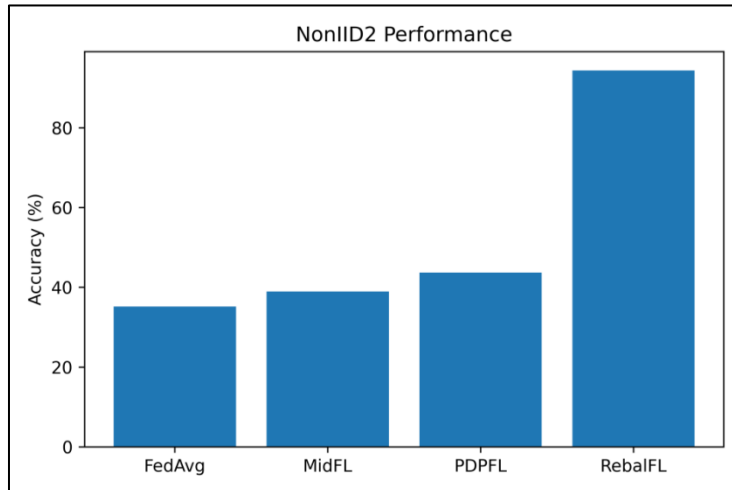


Figure 6: NonIID2

3.4.3 Step 3: Local Model Training and Update

During local model training, the client injects Gaussian noise $N(0, \sigma^2)$ into the gradients and continuously tracks the cumulative privacy expenditure $\rho_{i,j}$ for each data record. When the privacy budget of a particular record is exhausted, that record is excluded from subsequent training rounds. After completing local training, the client forwards the updated local model ω_i^{r+1} to the next client in the chain or directly to the corresponding edge server.

3.4.4 Step 4: Model Submission

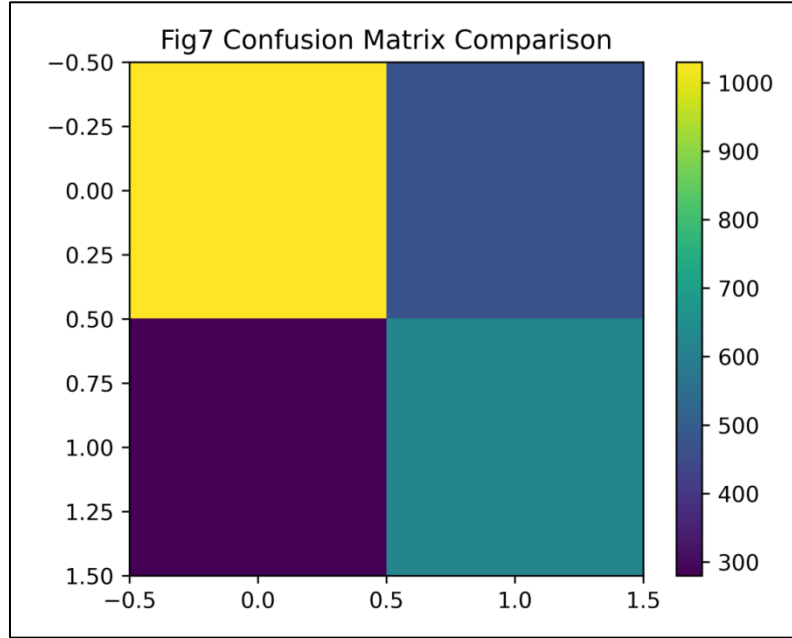


Figure 7: Confusion Matrix

After receiving the model update from the final client in the cluster, the edge server determines whether the maximum number of local iterations τ has been reached. If the local iteration limit has not been reached, the model is returned to the first client, and Step 3 is repeated. Otherwise, the locally trained model is submitted to the central server, completing one training round. The process then returns to Step 1 and continues iteratively until model convergence or the predefined global training round limit T is reached. After training is completed, RebalFL guarantees that the cumulative privacy expenditure $\rho_{i,j}$ of every data record throughout the federated learning process does not exceed its predefined privacy budget $\varepsilon_{i,j}$. Consequently, each data point in the system is provided with provable $(\varepsilon_{i,j}, \delta)$ – PDP protection.

3.5 Security Analysis

The objective of this section is to demonstrate that RebalFL satisfies $(\varepsilon_{i,j}, \delta)$ -Personalized Differential Privacy during cooperative spectrum sensing and can effectively defend against inference attacks launched by honest-but-curious central servers and secondary-user clients. Specifically, even if the honest-but-curious central server or participating clients can access global model parameters or gradient updates, they cannot infer the existence of any individual local data record.

3.6 Sensitivity Analysis

Sensitivity analysis is first conducted to ensure that the output distributions of RebalFL on any pair of neighboring datasets D_i and D'_i remain statistically indistinguishable, thereby concealing the presence or absence of individual records. To limit the influence of a single record on model updates, RebalFL introduces a gradient clipping mechanism during local training. Assume that client C_i possesses neighboring datasets D_i and D'_i , and let $f(D_i)$ denote the gradient update function computed on dataset D_i . Before performing gradient descent, the L_2 – norm of each sample gradient g is constrained by a threshold C : $|g|_2 \leq C$. According to the properties of gradient clipping, the upper bound of the L_2 -sensitivity Δ_s of the stochastic gradient descent function can be derived as

$$\Delta_s = \max_{D_i, D'_i} |f(D_i) - f(D'_i)|_2 = \eta \max |g_i - g'_i| = \frac{\eta}{|D_i|} 2C \leq C$$

The above result guarantees that the influence of adding or removing any individual data record on model parameter updates is strictly bounded by the threshold C . This demonstrates that the training process of RebalFL exhibits strong stability with respect to individual data records, and the dependency of model updates on any single record is tightly controlled, thereby satisfying the sensitivity requirements of differential privacy.

3.7 Privacy Analysis

Privacy analysis is performed under the framework of Rényi Differential Privacy (RDP) to derive the upper bound of privacy loss in RebalFL. By analyzing the privacy amplification effect introduced by Poisson sampling, the cumulative

privacy expenditure over multiple training rounds is computed. Subsequently, the conversion theorem between RDP and PDP is applied to prove that RebalFL satisfies $(\varepsilon_{i,j}, \delta) - PDP$. RebalFL adopts a Poisson-subsampled Gaussian mechanism to achieve differential privacy. For a data record $d_{i,j}$, let the sampling probability be $q_{i,j}$, and let the Gaussian noise standard deviation be σ .

RDP quantifies privacy leakage through the Rényi divergence between the output distributions of neighboring datasets, enabling more precise and rigorous privacy accounting. For Rényi divergence of order $\alpha > 1$, the privacy loss incurred by a single local iteration on record $d_{i,j}$ is upper-bounded by

$$\rho_{iter}(\alpha, q_{i,j}) \leq \frac{1}{\alpha - 1} \ln \{ (1 - q_{i,j})^{\alpha-1} (\alpha q_{i,j} - q_{i,j} + 1) + \alpha (1 - q_{i,j})^{\alpha-1} q_{i,j}^l e^{(l-1)\rho(l)} \}$$

where $\rho(\alpha) = \frac{\alpha}{2\sigma^2}$. From Equation (12), it can be observed that a smaller sampling probability $q_{i,j}$ results in a smaller single-step privacy loss ρ_{iter} , demonstrating the privacy amplification effect of subsampling.

Cumulative Privacy Loss: In federated learning, model training involves multiple communication rounds and local iterations. Assume that the total number of communication rounds is T , and each round contains τ local iterations. According to the adaptive composition theorem of RDP, the total privacy loss is equal to the sum of privacy losses incurred at each step. Therefore, for record $d_{i,j}$, the cumulative RDP privacy loss after completing all training rounds is

$$\rho_{\{total\}}(\alpha) = \sum_{t=1}^T \sum_{r=1}^{\tau} \rho_{iter}^{t,r}(\alpha) = T\tau \rho_{iter}(\alpha, q_{i,j})$$

Conversion from RDP to PDP: Finally, using the conversion lemma between RDP and Differential Privacy, for a given relaxation parameter $\delta > 0$, the cumulative RDP parameter can be transformed into $(\varepsilon_{i,j}, \delta) - PDP$ as follows:

$$\varepsilon_{\{i,j\}} = \min_{\alpha > 1} (\rho_{\{total\}}(\alpha) + \frac{\ln(\frac{1}{\delta})}{\alpha - 1})$$

Through the above derivation, it can be proven that RebalFL can accurately control the cumulative privacy expenditure $\rho_{\{total\}}$ of each record $d_{\{i,j\}}$ throughout $T\tau$ training iterations, ensuring that the final privacy consumption remains strictly bounded by the user-specified personalized privacy budget $\varepsilon_{\{i,j\}}$. Therefore, RebalFL satisfies $(\varepsilon_{\{i,j\}}, \delta) -$ Personalized Differential Privacy (PDP) and provides rigorous privacy protection guarantees for cooperative spectrum sensing in federated learning environments. Here is the scientific English translation of the provided section without changing the technical meaning.

4 Experimental Evaluation

This section presents the experimental design and performance analysis of RebalFL. The primary objective is to evaluate whether RebalFL can effectively mitigate model drift caused by Non-IID data characteristics while simultaneously providing personalized privacy protection, thereby significantly improving the convergence stability and final prediction accuracy of the global model during federated training.

4.1 Experimental Setup

4.1.1 Datasets and Model Description

To comprehensively evaluate the effectiveness of RebalFL, experiments were conducted using both the MNIST dataset and a simulated spectrum sensing dataset. Specifically, the MNIST dataset was first utilized to simulate a three-class spectrum occupancy classification task. Subsequently, a simulated spectrum sensing dataset was employed to perform a binary spectrum availability classification task that more closely resembles real-world spectrum sensing scenarios. The datasets and models used in the experiments are described as follows.

MNIST Dataset: Samples with labels 0, 1, and 2 from the MNIST dataset were selected and mapped to the spectrum sensing categories of low occupancy, moderate occupancy, and high occupancy, respectively. Each sample is a 28×28 grayscale image. The resulting subset contains 18,623 training samples and 3,147 testing samples, with a relatively balanced class distribution. Therefore, it is suitable for evaluating the fundamental classification performance of the proposed method on a general classification task.

Simulated Spectrum Sensing Dataset: To ensure the validity of the simulation experiments in cognitive radio environments, the dataset generation methodology follows the approach presented in [41]. The dataset is constructed based on a binary hypothesis testing framework. The energy statistic of the received signal is extracted as the primary classification feature, simulating the channel-state decision process performed by practical energy detectors.

Specifically: Label 0 represents an idle channel state. Label 1 represents a busy channel state. The training set contains 5,000 samples for each class, while the testing set consists of 1,500 idle samples and 900 busy samples. The overall training-to-testing ratio is approximately 8:2.

Model Configuration: A Convolutional Neural Network (CNN) is adopted as the core classification model. The convolutional and pooling layers are responsible for feature extraction from the input data. For each global communication round: Number of global communication rounds: 15, Number of local training epochs per client: 50, Batch size: 128, Learning rate: 0.1.

4.1.2 Privacy Budget Allocation Strategies

To evaluate the performance of RebalFL under different privacy budget allocation schemes, multiple privacy budget settings were designed for different tasks. **MNIST Three-Class Classification Task:** For samples belonging to labels 0, 1, and 2, privacy budgets ε_1 , ε_2 , and ε_3 were assigned, respectively. Three allocation strategies were considered.

Fixed Budget: Each class is assigned a predetermined and constant privacy budget: $\varepsilon_1 = 0.1, \varepsilon_2 = 1.0, \varepsilon_3 = 5.0$

Multi-Distribution Budget: The privacy budgets of the three classes are independently sampled from three different Gaussian distributions: $N_1(0.1, 0.01)$, $N_2(1.0, 0.05)$, $N_3(5.0, 0.5)$. This setting simulates scenarios where privacy requirements fluctuate across different clients.

Long-Tail Distribution Budget: The privacy budgets of all samples collectively follow a Pareto distribution with: Shape parameter = 1.0, Minimum value = 0.1. This configuration models a long-tail privacy preference distribution where most samples require stronger privacy protection, while a small number of samples require relatively weaker protection.

Simulated Spectrum Sensing Binary Classification Task. A fixed-budget allocation strategy is adopted: Class 0: ($\varepsilon_1 = 0.5$), Class 1: ($\varepsilon_2 = 0.05$). This setting is designed to evaluate model performance under significantly different privacy protection strengths.

4.2 Baseline Methods

To comprehensively evaluate the proposed approach, three representative baseline methods were selected for comparison. **FedAvg** [42] FedAvg is the classical Federated Averaging algorithm and does not employ any privacy protection mechanism. It serves as the performance upper-bound reference. **PDPFL** [30] PDPFL incorporates Personalized Differential Privacy (PDP), allowing different data records to adopt customized privacy budgets according to their privacy requirements. **MidFL** [43] adopts a uniform DP-FedAvg mechanism, where the privacy budget is set as the average of all privacy budgets and identical privacy protection strength is applied to all client data.

4.3 Accuracy Evaluation of RebalFL Under Privacy Protection

This section presents the experimental results of RebalFL and the three baseline methods on the MNIST three-class classification task. The primary objective is to investigate whether RebalFL can maintain high classification accuracy under different client data heterogeneity scenarios and privacy budget allocation strategies. All reported results represent the average values obtained from three independent experimental runs. Thirty secondary-user clients are deployed, and three data heterogeneity scenarios are constructed. **IID Scenario:** All clients possess samples from the three classes in a balanced manner, forming an ideal IID data distribution. **Mildly Heterogeneous Scenario (Non-IID1)** 10 clients contain samples from only one label. 10 clients contain samples from two labels. The remaining 10 clients contain samples from all three labels. This setting simulates commonly observed real-world data imbalance situations. **Severely Heterogeneous Scenario (Non-IID2)** Each client contains samples from only one class.

4.4 Performance Comparison Under the IID Scenario

Experiments were conducted under three privacy budget settings: Fixed-budget distribution, Multi-distribution budget, Long-tail budget distribution. The results are presented in Table 2 and Figure 4, where the performance of the proposed method is highlighted in bold. RebalFL maintains relatively stable training curves and high classification accuracy across all settings. Although its final accuracy under fixed-budget and multi-distribution settings is slightly lower than certain baselines, RebalFL still demonstrates strong overall performance and exhibits superior robustness under the extreme long-tail budget scenario.

Table 2: Performance Comparison Under the IID Scenario

Privacy Budget Setting	Method	Accuracy	Runtime (s)
Fixed Budget	FedAvg	99.65%	218
	MidFL	96.44%	482
	PDPFL	97.15%	630
	RebalFL	92.13%	620
Multi-Distribution	FedAvg	99.65%	218
	MidFL	75.18%	243
	PDPFL	89.47%	276
	RebalFL	85.28%	275
Long-Tail Budget	FedAvg	99.65%	218
	MidFL	74.40%	230
	PDPFL	76.84%	259
	RebalFL	83.27%	256

Under the extreme long-tail budget setting, FedAvg achieves the highest accuracy because no privacy noise is injected. Among privacy-preserving approaches, RebalFL achieves an accuracy of 83.27%, significantly outperforming PDPFL (76.84%) and MidFL (74.40%). This improvement can be attributed to the combined effect of the personalized differential privacy mechanism and the rebalanced clustering strategy. First, compared with uniform differential privacy mechanisms, personalized differential privacy reduces the overall amount of injected noise. Consequently, both RebalFL and PDPFL outperform MidFL. Second, the rebalanced clustering strategy effectively alleviates the uneven participation of low-budget samples caused by privacy budget disparities, allowing both high-budget and low-budget data to contribute more evenly to model training. As a result, RebalFL significantly outperforms PDPFL. From the perspective of runtime, RebalFL requires more training time than FedAvg and MidFL under both fixed-budget and multi-distribution settings. This overhead mainly arises from the additional computations introduced by the PDP mechanism and the rebalanced clustering strategy. However, considering the substantial gains in model accuracy and stability, the additional computational cost remains acceptable.

4.5 Performance Comparison Under the Mildly Heterogeneous Non-IID1 Scenario

Experiments were conducted under the Non-IID1 scenario using the three privacy budget settings: Fixed budget, Multi-distribution budget, Long-tail budget. The results are shown in Table 3 and Figure 5, with the proposed method highlighted in bold. Compared with the IID scenario, all privacy-preserving methods experience varying degrees of accuracy degradation, indicating that mild data heterogeneity exacerbates model drift and consequently reduces classification performance.

Table 3: Performance Comparison of Different Methods in the Non-IID1 Scenario

Privacy Budget Setting	FedAvg [42] Accuracy	FedAvg Time (s)	MidFL [43] Accuracy	MidFL Time (s)	PDPFL [30] Accuracy	PDPFL Time (s)	RebalFL (Proposed) Accuracy	RebalFL Time (s)
Fixed Budget	98.41%	235	89.25%	469	93.02%	610	95.36%	626
Multi-Distribution Budget	98.41%	235	69.48%	231	74.61%	268	88.30%	276
Long-Tail Budget	98.41%	235	64.96%	217	73.17%	243	85.66%	256

Although FedAvg maintains the highest overall accuracy due to the absence of privacy noise, significant oscillations are observed during the early stages of training. In contrast, RebalFL consistently exhibits smoother convergence curves and achieves the best overall performance among all privacy-preserving methods under every privacy budget configuration. Under the fixed-budget setting, RebalFL achieves an accuracy of 95.36%, outperforming MidFL (89.25%) and PDPFL (93.02%). Under the multi-distribution budget setting, the accuracies of MidFL and PDPFL decrease to 69.48% and 74.61%, respectively, whereas RebalFL still maintains a relatively high accuracy of 88.30%. Notably, under the long-tail budget setting, the accuracy of MidFL further declines to 64.96%, while PDPFL achieves 73.17%. RebalFL, however, still obtains a high accuracy of 85.66%, substantially mitigating the performance degradation caused by mild data heterogeneity.

Overall, the superior performance of RebalFL can be attributed to the synergistic effect of two key design components. First, RebalFL adopts a Personalized Differential Privacy (PDP) mechanism combined with a sampling strategy, ensuring that low-budget data records are not prematurely excluded from the training process due to early exhaustion of their privacy budgets. This effectively mitigates the issue of “data forgetting” and enables the global model to benefit from a more comprehensive contribution of training data. Second, the proposed rebalancing clustering strategy directly alleviates model drift by correcting data distribution disparities among clients. Consequently, RebalFL achieves higher performance and more stable convergence under mildly heterogeneous data distributions.

4.6 Performance Comparison in the Severely Heterogeneous Non-IID2 Scenario

In the severely heterogeneous Non-IID2 scenario, the data distribution disparity among secondary-user clients is further amplified. The class distributions of individual clients are almost completely non-overlapping, representing an extreme heterogeneous data setting. The experimental results are presented in Table 4 and Figure 6, where the results of the proposed RebalFL framework are highlighted in bold. Under this environment, the conventional FedAvg algorithm experiences severe performance degradation due to model drift, with classification accuracy dropping dramatically to 35.18%. Under the long-tail privacy budget setting, MidFL achieves only 43.21% accuracy, while PDPFL further declines to 34.38%. In contrast, RebalFL demonstrates remarkable stability under identical settings, consistently outperforming all baseline methods. Even in the most challenging long-tail privacy budget scenario, RebalFL maintains an accuracy of 82.07%. These results demonstrate that the proposed client rebalancing clustering strategy can effectively identify and group secondary-user clients with complementary data distributions. By mitigating model drift at its source, RebalFL is able to maintain high classification performance across a wide range of Non-IID environments.

Table 4: Performance Comparison of Different Methods in the Non-IID2 Scenario

Privacy Budget Setting	FedAvg [42] Accuracy	FedAvg Time (s)	MidFL [43] Accuracy	MidFL Time (s)	PDPFL [30] Accuracy	PDPFL Time (s)	RebalFL (Proposed) Accuracy	RebalFL Time (s)
Fixed Budget	35.18%	223	38.95%	514	43.72%	642	94.42%	624
Multi-Distribution Budget	35.18%	223	45.24%	227	44.17%	256	84.73%	280
Long-Tail Budget	35.18%	223	43.21%	240	34.38%	280	82.07%	256

4.7 Effectiveness Analysis of RebalFL in Mitigating Model Drift

This section presents the experimental setup and performance evaluation of RebalFL and PDPFL [30] on the simulated spectrum sensing binary classification task. The evaluation is conducted using precision, recall, F1-score, accuracy, and confusion matrix analysis, with the primary objective of investigating whether RebalFL can effectively mitigate model drift caused by Non-IID data distributions and the Personalized Differential Privacy mechanism. A total of 20 secondary-user clients are configured to simulate a mildly heterogeneous environment. Among them, 10 clients are mixed-data clients, each containing 70% Class-0 (idle state) samples and 30% Class-1 (busy state) samples. The remaining 10 clients are single-class clients, where five clients contain only Class-0 samples and the other five contain only Class-1 samples. For the simulated spectrum sensing binary classification task, RebalFL and PDPFL are compared using four evaluation metrics: precision, recall, F1-score, and accuracy. The results are summarized in Table 5. The results indicate that although PDPFL maintains a very high precision of 100.00% for Class-0 samples, its overall performance remains limited. In particular, its ability to identify Class-1 samples is significantly weaker, achieving a recall of only 47.78%, which results in an overall F1-score of merely 64.66%. By contrast, RebalFL achieves superior performance across all

evaluation metrics. The overall classification accuracy increases to 88.13%, while the F1-score improves to 81.31%, demonstrating a more balanced and reliable classification capability.

The confusion matrices shown in Figure 7 provide a more intuitive comparison. As illustrated in Figure 7(a), PDPFL incorrectly classifies 470 true Class-1 samples as Class-0 samples. In comparison, Figure 7(b) shows that RebalFL significantly enhances the recognition of Class-1 samples, reducing the number of misclassified Class-1 instances to only 280. These experimental findings clearly demonstrate that the proposed rebalancing clustering strategy effectively corrects model drift caused by heterogeneous privacy budget allocations. While improving overall classification performance, RebalFL substantially enhances the recognition capability for strongly protected privacy-sensitive data. For practical spectrum sensing applications, RebalFL provides more reliable detection of busy-spectrum states associated with primary users, thereby reducing the likelihood of interference during spectrum access decisions and offering greater practical value in real-world cognitive radio environments. Here is the English translation in scientific paper format without changing the technical meaning.

Table 5: Performance Comparison on Simulated Spectrum Sensing Binary Classification Task

Method	Precision	Recall	F1-Score	Accuracy
PDPFL [30]	100.00%	47.78%	64.66%	80.42%
RebalFL (Proposed Scheme)	99.20%	68.89%	81.31%	88.13%

5 Conclusion

With the increasing scarcity of spectrum resources, Federated Learning (FL) has emerged as a promising approach for building efficient and privacy-preserving cooperative spectrum sensing systems. However, achieving reliable privacy protection and high model accuracy simultaneously under Non-IID data distributions and heterogeneous privacy requirements remains a critical challenge. To address this issue, this paper proposed a Federated Learning framework that integrates Personalized Differential Privacy and a Rebalancing Clustering Strategy, referred to as RebalFL. Specifically, a record-level Personalized Differential Privacy (PDP) mechanism was introduced, allowing clients to flexibly allocate privacy budgets according to the sensitivity of individual data records. This mechanism provides rigorous privacy guarantees while significantly reducing the negative impact of excessive noise injection on model utility. Furthermore, a KL-divergence-based client rebalancing clustering strategy was designed to construct distribution-balanced training clusters, thereby effectively mitigating the model drift problem caused by Non-IID data distributions. Experimental results demonstrate that RebalFL achieves superior privacy-preserving performance and model utility under various Non-IID scenarios. The proposed framework consistently maintains high classification accuracy while satisfying personalized privacy requirements, highlighting its effectiveness for privacy-preserving cooperative spectrum sensing applications. In practical future deployments, Federated Learning systems may face various malicious threats, including Byzantine attacks and data poisoning attacks. Although the proposed framework was not specifically designed to defend against such adversarial behaviors, its Personalized Differential Privacy mechanism introduces noise that can partially dilute abnormal gradients uploaded by malicious clients, thereby mitigating the impact of certain poisoning attacks. Moreover, the proposed rebalancing clustering strategy groups clients according to their data distributions. By incorporating client trustworthiness evaluation mechanisms in future work, it may become possible to identify and isolate anomalous participants within clusters, thereby enhancing the robustness of the system against distributed attacks.

Future research will further investigate optimization techniques for Differential Privacy in Federated Learning and explore the applicability of the proposed framework to a broader range of wireless communication scenarios, including spectrum prediction, dynamic spectrum management, and resource allocation. Such efforts are expected to further promote the adoption of privacy-preserving Federated Learning technologies for secure and efficient spectrum resource sharing in next-generation wireless communication systems.

Statements on compliance with ethical standards

Disclosure of conflict of interest

The authors declare no conflicts of interest regarding this manuscript

References

- [1] H. Zhang, Y. Nie, J. Cheng, V. C. M. Leung, and A. Nallanathan, "Sensing time optimization and power control for energy efficient cognitive small cell with imperfect hybrid spectrum sensing," *IEEE Transactions on Wireless Communications*, vol. 16, no. 2, pp. 730–743, Feb. 2017.
- [2] Y. Zeng, Y.-C. Liang, A. T. Hoang, and R. Zhang, "A review on spectrum sensing for cognitive radio: Challenges and solutions," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, no. 1, pp. 1–15, 2010.
- [3] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 116–130, First Quarter 2009.
- [4] A. Ghasemi and E. S. Sousa, "Spectrum sensing in cognitive radio networks: Requirements, challenges and design trade-offs," *IEEE Communications Magazine*, vol. 46, no. 4, pp. 32–39, Apr. 2008.
- [5] N. Carlini, C. Liu, Ú. Erlingsson, J. Kos, and D. Song, "The secret sharer: Measuring unintended neural network memorization and extracting secrets," in *Proc. USENIX Security Symposium*, 2019, pp. 267–284.
- [6] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Artificial Intelligence and Statistics (AISTATS)*, 2017, pp. 1273–1282.
- [7] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2019, pp. 14774–14784.
- [8] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Foundations and Trends in Theoretical Computer Science*, vol. 9, nos. 3–4, pp. 211–407, 2014.
- [9] R. C. Geyer, T. Klein, and M. Nabi, "Differentially private federated learning: A client level perspective," *arXiv:1712.07557*, 2018.
- [10] A. Erlingsson, V. Pihur, and A. Korolova, "RAPPOR: Randomized aggregatable privacy-preserving ordinal response," in *Proc. ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2014, pp. 1054–1067.
- [11] D. Cabric, A. Tkachenko, and R. W. Brodersen, "Spectrum sensing measurements of pilot, energy, and collaborative detection," in *Proc. IEEE Military Communications Conference (MILCOM)*, 2006, pp. 1–7.
- [12] H. Urkowitz, "Energy detection of unknown deterministic signals," *Proceedings of the IEEE*, vol. 55, no. 4, pp. 523–531, Apr. 1967.
- [13] W. A. Gardner, "Exploitation of spectral redundancy in cyclostationary signals," *IEEE Signal Processing Magazine*, vol. 8, no. 2, pp. 14–36, Apr. 1991.
- [14] Y. Zeng and Y.-C. Liang, "Eigenvalue-based spectrum sensing algorithms for cognitive radio," *IEEE Transactions on Communications*, vol. 57, no. 6, pp. 1784–1793, Jun. 2009.
- [15] S. Haykin, "Cognitive radio: Brain-empowered wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, Feb. 2005.
- [16] M. Saber, A. E. Rharras, R. Saadane, and A. Chehri, "An optimized spectrum sensing implementation based on SVM, KNN and TREE algorithms," in *Proc. 15th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, 2019, pp. 383–389.
- [17] C. Liu, J. Wang, X. Liu, Y. Liang, and Y. Chen, "Deep CM-CNN for spectrum sensing in cognitive radio," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, pp. 2306–2321, Oct. 2019.
- [18] S. Solanki, V. Dehalwar, J. Choudhary, and S. B. Goyal, "Spectrum sensing in cognitive radio using CNN-RNN and transfer learning," *IEEE Access*, vol. 10, pp. 113482–113492, 2022.
- [19] H. Li, T. Hu, J. Chen, and Y. Wang, "Privacy preserving algorithm for spectrum sensing in cognitive vehicle networks," *Chinese Journal of Electronics*, vol. 33, no. 1, pp. 30–42, Jan. 2024.
- [20] J. Konečný, H. B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," *arXiv:1610.02527*, 2016.
- [21] Z. Chen, Y. Xu, H. Wang, and Y. Li, "Federated learning-based cooperative spectrum sensing in cognitive radio," *IEEE Communications Letters*, vol. 26, no. 2, pp. 330–334, Feb. 2022.
- [22] Y. Song, H. Chang, and L. Liu, "Federated dynamic spectrum access through multi-agent deep reinforcement learning," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, 2022, pp. 3466–3471.

- [23] Y. Shi, Y. E. Sagduyu, and T. Erpek, "Federated learning for distributed spectrum sensing in NextG communication networks," arXiv:2204.03027, 2022.
- [24] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine Learning and Systems*, vol. 2, pp. 429–450, 2020.
- [25] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020, pp. 7611–7623.
- [26] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proc. ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2016, pp. 308–318.
- [27] N. Papernot et al., "Scalable private learning with PATE," in *Proc. International Conference on Learning Representations (ICLR)*, 2018.
- [28] R. Shokri and V. Shmatikov, "Privacy-preserving deep learning," in *Proc. ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2015, pp. 1310–1321.
- [29] J. Liu, J. Lou, L. Xiong, J. Ren, and J. Niu, "Projected federated averaging with heterogeneous differential privacy," *Proceedings of the VLDB Endowment*, vol. 15, no. 4, pp. 828–840, 2021.
- [30] Z. Jorgensen, T. Yu, and G. Cormode, "Conservative or liberal? Personalized differential privacy," in *Proc. IEEE International Conference on Data Engineering (ICDE)*, 2015, pp. 1023–1034.
- [31] B. Niu, Y. Chen, B. Wang, and H. Li, "AdaPDP: Adaptive personalized differential privacy," in *Proc. IEEE INFOCOM*, 2021, pp. 1–10.
- [32] F. Boenisch, A. Dziedzic, C. Mühl, and N. Papernot, "Have it your way: Individualized privacy assignment for DP-SGD," in *Advances in Neural Information Processing Systems*, vol. 36, 2023.
- [33] M. A. Aygül, H. A. Çırpan, and H. Arslan, "Machine learning-based spectrum occupancy prediction: A comprehensive survey," *Frontiers in Communications and Networks*, vol. 6, 2025.
- [34] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Proc. Theory of Cryptography Conference (TCC)*, 2006, pp. 265–284.
- [35] Z. Jorgensen, T. Yu, and G. Cormode, "Personalized differential privacy," *Journal of Privacy and Confidentiality*, vol. 7, no. 2, pp. 1–28, 2015.
- [36] Z. Liu, T. Li, V. Smith, and V. Sekar, "Enhancing the privacy of federated learning with sketching," arXiv:1911.01812, 2019.
- [37] J. Hayes, L. Melis, G. Danezis, and E. De Cristofaro, "LOGAN: Membership inference attacks against generative models," *Proceedings on Privacy Enhancing Technologies*, vol. 2019, no. 1, pp. 133–152, 2019.
- [38] Z. Wang, M. Song, Z. Zhang, Y. Song, Q. Wang, and H. Qi, "Beyond inferring class representatives: User-level privacy leakage from federated learning," in *Proc. IEEE INFOCOM*, 2019, pp. 2512–2520.
- [39] I. Mironov, K. Talwar, and L. Zhang, "Rényi differential privacy of the sampled Gaussian mechanism," arXiv:1908.10530, 2019.
- [40] I. Mironov, "Rényi differential privacy," in *Proc. IEEE Computer Security Foundations Symposium (CSF)*, 2017, pp. 263–275.
- [41] Y. Arjoune and N. Kaabouch, "On spectrum sensing, a machine learning method for cognitive radio systems," in *Proc. IEEE International Conference on Electro Information Technology (EIT)*, 2019, pp. 333–338.
- [42] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. AISTATS*, 2017, pp. 1273–1282.
- [43] R. C. Geyer, T. Klein, and M. Nabi, "Differentially private federated learning: A client level perspective," arXiv:1712.07557, 2018.