Al-Rafidain Journal of Computer Sciences and Mathematics (RJCM)

www.csmj.uomosul.edu.iq

A Survey on Federated Learning: Fundamentals, Challenges and Client Selection Methods

Alaa Amjad Mala Baker[1] and Manar Younis Kashmola[2]

[1]Computer Science Department, College of Computer Science and Mathematics, University of Mosul, Iraq
[2]College of Information Technology, University of Ninevah, Iraq
Email: alaa.23csp52@student.uomosul.edu.iq[1] and manar.kashmola@uoninevah.edu.iq[2]

Article information

Article history:
Received 31 May, 2025
Revised 17 August, 2025
Accepted 15 November, 2025
Published 25 December, 2025

Keywords:
Client selection,
federated learning,
machine learning.

Correspondence:
Alaa Amjad Mala Baker
Email:
alaa.23csp52@student.uomosul.edu.iq

Abstract

Federated learning (FL) represents a transformative shift in machine learning, moving from conventional centralized approaches to a distributed framework that emphasizes data privacy and security. The FL server transmits an initial model to clients, which they train locally on their private data. After training, the server aggregates model updates from each client to update the global model. Selecting the best clients in FL is critical to improving the convergence speed and accuracy of the final model, which requires careful client selection approaches. The client selection phase of FL faces numerous challenges that impact overall training performance, including statistical heterogeneity and system heterogeneity resulting from the diversity of client data and resources. Communication costs present another challenge, especially in networks with limited client communication resources. Additionally, selecting trustworthy clients represents another challenge, as selecting malicious clients creates a significant risk within the FL training process. Moreover, the fairness challenge entails providing fair opportunities for all clients to participate in training. To address these challenges, we offer solutions that utilize effective techniques, approaches, and client selection methods. This survey presents a taxonomy of modern client selection methods in FL, highlighting the improvements in FL performance and effectiveness achieved through these methods, including greedy selection, reinforcement learning-based selection, multi-armed bandit-based selection, clustering-based selection, and reputation & security-based selection. Subsequently, it offers a general comparison between these methods in terms of their core ideas, advantages, limitations, and use cases. Finally, future and potential trends in client selection and improving performance in FL are identified.

DOI: 10.33899/rjcsm.v19i2.60317, ©Authors, 2025, College of Computer Science and Mathematics, University of Mosul, Iraq.

1. Introduction

Researchers and businesses alike have shown a growing interest in machine learning (ML) over the past years [1]. Mobile devices, such as smartphones and tablets, are supplied with advanced sensors that generate a huge amount of data. ML can utilize this data for training by transferring it to a central server. On the other hand, the approach raises significant concerns regarding the potential for privacy violations of data owners [2], [3].

Federated Learning (FL) is an innovative technology designed to develop ML models to protect data privacy. The model is trained in FL without sending data to an FL server, this is achieved by having the participating devices (also referred to as clients) collaborate through a decentralized approach to train the model using data stored locally on each device. This approach is not always feasible with conventional ML techniques, which require data collection from each device [4].

Within the FL framework, client selection represents a crucial phase that involves choosing devices to participate in the iterative training rounds. The FL server typically randomly selects a set of clients to train the model. [5]. Random selection of clients in FL degrades performance because it does not consider the client devices' data quality and computational resources (such as CPU, storage, and power) in the selection process [6], [7]. Current client selection methods attempt to improve the random client selection method. Therefore, this research delineates the latest client selection methods, explains their foundational concepts and salient qualities, and demonstrates how they enhance the performance of FL.

Choosing the optimal clients among the candidates in the training process is crucial in enhancing the effectiveness of FL. FL encounters several challenges through the client chosen process, including statistical heterogeneity, system heterogeneity, communication costs, privacy concerns, and fairness [5], [8]. Despite the development in client selection methods, numerous future trends and research opportunities in this area exist that can contribute to enhancing the performance of FL. This survey addresses the subsequent research questions:

• Research Question 1: What is the process for selecting clients for FL, and how are clients selected?

• Research Question 2: What are the main challenges associated with client selection and its addressing strategies in FL?

• Research Question 3: What are the modern client selection methods, and what enhancements do they contribute to FL's performance?

• Research Question 4: What are the future and potential trends in client selection and improving FL performance?

The main contributions of this survey are to identify the challenges of client selection in FL and propose strategies for addressing them, discuss client selection methods in FL in light of the latest developments, and highlight their impact on improving FL performance.

The following sections of this survey are organized as follows. Section 2 presents the literature review. Section 3 presents the basic principles of FL and client selection. The challenges of client selection and solution are shown and discussed in section 4. Selecting clients' methods is discussed in Section 5. Section 6 presents the discussion of the survey. Finally, future research directions are highlighted in Section 7, whereas Section 8 concludes the research contributions and findings.

2. Literature Review

Numerous relevant works exist on client selection and challenges in FL. Nevertheless, while these works are essential, they do not provide an exhaustive overview of client selection methods, challenges, and future trends. Several studies have discussed the FL challenges regardless of the client selection methods [9], [10], [11]. Hosseinzadeh et al. [12] presented a systematic literature review (SLR) on FL in the IoT field, examining evaluation factors and offering a potential client selection approach, but did not further explore the subject. Abreha et al. [13] discussed client selection to a minor extent by presenting some methods related to model convergence.

Selecting clients with non-IID (Independent and Identically Distributed) data during the training process can reduce model accuracy and training bias. Ma et al. [14] studied the solution of non-IID data in FL through data optimization and client selection. However, the survey just links to pertinent studies and does not delve deeper. Lei et al. [15] studied the principles, challenges, and opportunities associated with client selection in FL. The authors focused on statistical and system heterogeneity. The study provided the selection methods using utility functions for each client, which were categorized into statistical and system utility. However, it is not a comprehensive survey, as it just gives general selection criteria. Wafa et al. [16] discuss client selection strategies to address the problems faced by the global server in aggregating model parameters from client devices, particularly the variability in client involvement behaviors. However, the study focused on only three client selection methods.

Therefore, current surveys are restricted in scope and depth, lacking the ability to offer an accurate evaluation of client selection methods, challenges, and prospective trends. To address these constraints, our research provides a detailed overview of advanced client selection methods based on key principles and concepts. In addition, this research investigates the challenges of client selection methods and their prospective trends that can enhance FL performance and efficiency. This survey promotes the future development of client selection methods and inspires academics and practitioners to gain a greater awareness of the research issue.

3. Background

3.1. Federated learning

Federated learning is a rapidly developing paradigm in the field of ML. It has attracted significant attention from academics to investigate its potential and applications [17]. The basis of FL involves an approach that allows participants, including devices or nodes, to train a shared model on their local data in a decentralized and autonomous manner on-site rather than transmitting the data to a central server. After that, only model updates are sent to the FL server. All updates from participating clients are aggregated by the FL server to create a global model [18], as shown in Figure 1. For example, Google uses FL in its Gboard keyboard to learn the best word suggestions without sending user messages to the servers. Each phone learns locally from the user's typing and then shares only the updates [19]. FL has several main categories, depending on data distribution across parties, as shown in Figure 2 [20]. These encompass horizontal FL, characterized by similar features across parties but distinct user data; vertical FL, where parties share identical users yet possess different features. In addition, transfer FL is applicable in the absence of shared features or users and relies on knowledge transfer. Moreover, personalized FL, which seeks to tailor the model to the specific requirements of each party, and decentralized FL, wherein models are exchanged without necessitating a central server [21], [22].

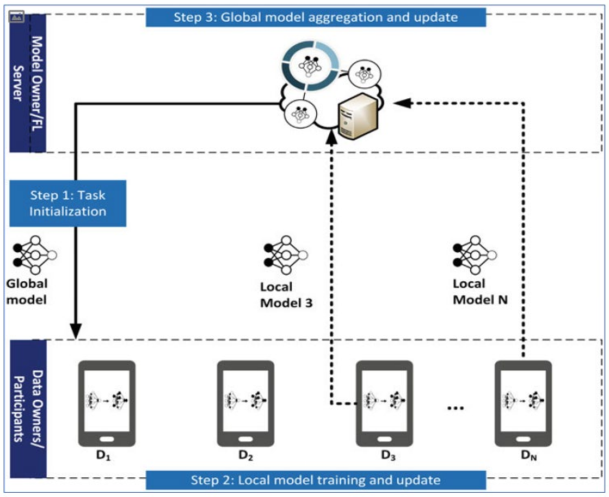

Figure 1. Federated learning system overview.

There are three main phases of FL training, as illustrated below [5], [23]:

1-Initialization phase: The server determines the FL task, such as the training process and the target application. Additionally, it creates an initial global model and specifies its hyperparameters, including the learning rate. After selecting clients among candidates for model training and update, the server distributes the initial global model to clients.

2-Local training phase: Subsequent to receiving the initial global model, clients train this model on their local data and find the optimal parameters that reduce their individual loss functions. After that, the FL server receives the updated local model parameters.

3-Aggregation phase: Finally, the FL server assembles updated local model parameters from clients to create an updated global model and sends it again to the selected clients.

The second and third phases will be repeated in several rounds until the model converges or the predefined target training accuracy is reached. FL has been applied in several areas, like Internet of Things (IoT), vehicular networks, healthcare, and attack detection [24], [25], [5], [26], [27].

3.2. Client selection

Client selection is an important element in FL, significantly impacting the efficacy of the training process, the final model accuracy, and the duration of training [28]. Several studies have confirmed that the FL performance deteriorates when clients are selected randomly [29]. Different client selection methods aim to choose the best clients during each round, thereby enhancing the FL's efficacy, accelerating training, and improving model quality [30]. However, the client's selection process faces several challenges, which will be discussed in the next section.

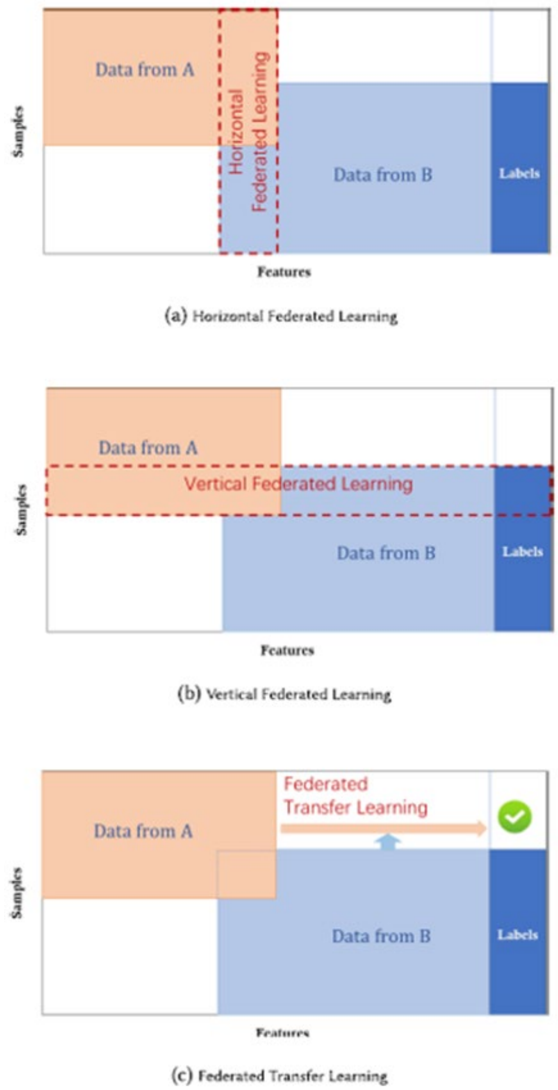

Figure 2. Categories of federated learning.

4. Client selection challenges

The client selection process in FL faces many challenges that may impact its performance, as shown below:

4.1. Statistical Heterogeneity

This challenge is also known as data heterogeneity. Clients in FL may have different data distributions and skew in terms of labels, features, quality, and quantity due to differing data-gathering motivations and behaviors, as shown in Figure 3[31]. In applications of the real world, client data distributions typically do not reflect the population distribution, indicating non-independent and identically distributed data (non-IID). For example, in the healthcare field, patient characteristics, disease prevalence, and diagnostic techniques might differ markedly among hospitals, leading to non-IID data distributions [32].

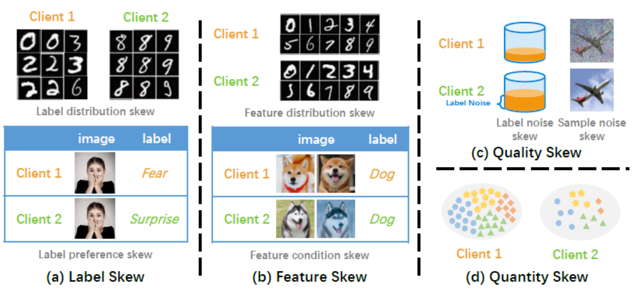

Figure 3. The different skew patterns in statistical heterogeneity.

Selecting clients with non-IID data in the training process degrades model accuracy and slows convergence, leading to biased global models and reduced generalization capabilities [33]. Conventional FL algorithms typically assume that data is IID among clients, which is rarely the case in practical situations. Diverse client selection methodologies have been suggested to address this challenge, including clustering-based client selection, adaptive client selection, and quality-based client selection [34], [35], [36].

4.2. System Heterogeneity

This challenge is also known as device heterogeneity or client heterogeneity. The client devices participating in the training have different capabilities and resources with respect to computation, storage, power, and memory. Client device heterogeneity can have many effects, such as different local training times [37]. In synchronous FL, the straggler problem arises when the duration of each training round is dictated by the slowest client, requiring the server to wait for updates from this client to finish training for each round [38].

In addition, client devices may have restricted processing capacity and battery life, which can impact the increased dropout rate or clients' ability to participate in training [37]. Client selection strategies must consequently account for the heterogeneity of devices and their limited resources. Research has been presented to address the heterogeneity of client hardware resources by relying on client selection methods, such as hierarchical federated learning-based client selection, reputation-based client selection, and clustering-based client selection [39], [40], [41].

4.3. Communication Cost

The clients and the FL server cooperate in numerous communication rounds during the FL training process to achieve the target accuracy level. In FL settings, exchanging only the parameters of the model between the FL server and clients, instead of raw client data, reduces communication resource consumption [42]. However, communication resources are still a serious challenge given the significant number of participant clients in training, and each update potentially includes parameters millions, such as those seen in convolutional neural networks (CNNs). Therefore, the significant dimensionality of the updates may incur communication resource consumption [5], [43] .
The Communications cost challenge has several key issues as shown below:

a. Limited communication capacities

Devices may have limited communication capabilities, such as transmission power and bandwidth, which can affect the efficiency of client selection and lead to a long convergence time [43].

b. Wireless Networks Dynamicity

Client network resources in dynamic wireless networks may constantly change, leading to connectivity issues. In addition, fading of channels in wireless networks can lead to the loss of certain client updates. The dynamic nature of networks and high-mobility environments may result in client unavailability [44].

c. Network latency

High latency might adversely affect client selection efficiency and hinder global model convergence [45]. The communication cost in FL can be mitigated using specific strategies and methods, including quantization, compression strategies, reducing communication rounds, reducing the size of transmitted updates, and hierarchical federated learning [46], [47], [48].

4.4. Security and privacy concerns

FL aims to maintain clients' privacy by enabling them to share only the parameters of the trained model, rather than clients' local raw data. On the other hand, the results of certain recent research have shown the possibility of privacy and security problems if FL clients or servers behave maliciously [49]. Selecting malicious clients creates a significant risk within the FL training process. By manipulating their data and local models, malicious clients can compromise the security of the FL model, ultimately resulting in the disclosure of sensitive information to other clients [50]. Privacy considerations are essential in FL, and client selection is vital for maintaining the confidentiality of client data. The core principle of FL is the separation of model training from direct access to raw training data, thus reducing privacy and security threats. Client selection processes may unintentionally disclose sensitive information if not carefully constructed with privacy considerations in mind. Therefore, there is a need to implement techniques and strategies capable of preventing the selection of harmful clients and methods to exclude them from participating in training [49], [50], such as:

- Reputation-based client selection.
- Selection of trustworthy clients.
- Secure multi-party computing.
- Secure aggregation.

Additional techniques can be used to maintain the security and privacy of the FL learning procedure and mitigate the influence of malicious clients [51], [52]. These techniques include:

- Homomorphic encryption.
- Differential privacy.
- Blockchain.
- Perturbation-based defense.
- Knowledge distillation.
- Adversarial machine learning.

4.5. Fairness

Fairness in an FL system means providing equal selection opportunities for all clients in the training. Unfairness can arise when clients with high abilities are always selected while clients with weaker abilities are excluded from the selection process or have limited opportunities to participate [53].

Unfairness in FL may lead to several problems, such as [54], [55] :

- Selection bias.
- Degraded model accuracy.
- Biased updates and unwanted effects.
- The selected clients do not accurately reflect the overall data distribution.

Client selection bias may be observed through the under-representation of particular client groups, the over-representation of others, or the systematic exclusion of clients possessing specified attributes. When specific client groups are persistently underrepresented in the training process, the resultant model may demonstrate inadequate performance on their data, resulting in unfair outcomes [56]. Fairness is an emerging research challenge in FL. Several techniques have been developed to ensure fairness [57], [58], [59] such as:

- Active learning-based client selection.
- Stratified sampling-based client selection.
- Diversity-based client selection.
- Reinforcement learning- based client selection.

Table 1 summarizes the impact of challenges on FL performance and client selection, while **Table 2** presents solutions to the client selection challenges.

Table 1. The impact of challenges on federated learning performance and client selection.

Ref. No	Challenge	Impact
[33]	Statistical Heterogeneity	• Degradation of the model accuracy • Slowed down the convergence • Biased global models • Reduction of generalization capabilities
[37], [38]	System Heterogeneity	• The stragglers' problem arises when the slower client's training duration determines the training duration for each round • Increasing clients' dropout rate in training
[5], [43], [44], [45]	Communication Cost	• Communication resource consumption • Affects the efficiency of client selection and leads to a long convergence time • Loss of specific client updates • Client unavailability • Hinder of global model convergence
[50]	Security and privacy concerns	• Compromise the security of the FL model • Disclosure of sensitive client information
[54], [55]	Fairness	• Selection bias • Degradation of the model accuracy • Biased updates and unwanted effects • The selected clients do not accurately reflect the overall data distribution

Table 2. Solutions to client selection challenges.

Ref. No	Challenge	Solutions
[34], [35], [36]	Statistical Heterogeneity	• Clustering-based client selection • Quality-based client selection • Adaptive client selection
[39], [40], [41]	System Heterogeneity	• Hierarchical federated learning-based client selection • clustering-based client selection • Reputation-based client Selection
[46], [47], [48]	Communicat-ion Cost	• Quantization • Compression strategies • Reduction of communication rounds • Reduction of transmitted updates size • Hierarchical federated learning-based client selection
[49], [50],	Security and privacy concerns	• Reputation-based clients selection • Selection of trustworthy clients

[51], [52]		• Secure multi-party computing • Secure aggregation • Other techniques
[57], [58], [59]	Fairness	• Active learning-based client selection • Stratified sampling-based client selection • Diversity-based client selection • Reinforcement learning-based client selection

5. Client selection methods

Researchers have suggested several studies regarding client selection in FL. In this section, we classify them based on their main principles.

5.1. Greedy selection

This method denotes a strategy for selecting clients in FL in a greedy manner to improve the global model's efficacy. In other words, it prioritizes clients with higher quality scores, which provide the most advantageous updates to the global model, rather than choosing them randomly.

Pranava et al. [60] proposed a client selection technique in a greedy manner called GREEDYFED, in each communication round, the clients who contribute the most to training are selected to achieve high communication efficiency in FL. This biased approach uses the Shapley value to address applications that have timing restrictions on connections with the parameter server. This method achieved rapid convergence with high accuracy in heterogeneous environments.

Jingyuan et al. [61] presented a study on client selection in Over-the-Air federated learning to improve communication efficiency in terms of transmission power. This study tackles signal aggregate errors and straggler client selection using the proposed approach called FedAirAoI, which is implemented in several stages. Initially, in each communication round, client priorities are determined using Lyapunov optimization. Secondly, clients of the highest priority are selected using a greedy algorithm. Then, the time-average MSE is minimized by addressing the transmission power optimization problem and determining the normalization factor for the selected set of clients. The results of this study showed enhanced model performance, ensuring timely updates and achieving a balance between training efficiency and fairness.

The greedy method has also been used in several studies as part of FL's client selection process. Nacho & Yonetani [62] proposed a FedCS protocol for client selection in FL from heterogeneous clients using a greedy approach in the mobile edge environment. Clients that required minimal time for loading and updating the model were selected, yielding results that completed the training process quickly and produced high-performance learning models. Zhang et al. [63] presented a method for selecting and scheduling clients with fairness guarantees in FL via multi-criteria system. A greedy algorithm is proposed to address the initial client set selection problem, an optimization problem that aims to maximize the total scores of the selected clients. This research result proved that the proposed strategy could enhance the quality of the trained model.

5.2. Reinforcement learning-based selection

Reinforcement Learning (RL) is a type of ML that effectively manages and enhances complex decision-making processes by interacting with a specific environment. It enables an agent to acquire knowledge about a situation by trying different actions. Asadullah et al. [64] presented a method for client selection that combines an RL approach with reputation and trust mechanisms. The authors used Q-learning, derived from reinforcement learning, to select clients with high reputations and trust. This research showed the potential for enhancing the model's accuracy, generalization, convergence speed, and mitigating malicious attacks.

Hongwei et al. [65] proposed FedPRL to address the heterogeneity of systems and data in heterogeneous environments. The authors of this study employed strategies based on RL, improving global model contributions, and evaluating client quality to select participants in the FL training process. The proposed study enhanced the efficiency of the training process and the global model generalization, and it could be integrated into the healthcare field.

The potential of deep reinforcement learning in client selection has emerged for the FL training process. Xutao et al. [66] proposed a client selection strategy for the FL training process utilizing the Deep Reinforcement Learning-based double DQN (DDQN) algorithm in heterogeneous environments. The proposed method demonstrated excellent effectiveness on the non-IID dataset while reducing the number of communication rounds and epochs.

Utilizing trust-based deep reinforcement learning, Ghaith et al. [67] developed a method for selecting the most suitable clients in terms of the time spent on training and the resources consumed. The authors applied this method to COVID-19 detection, attaining an effective balance between model execution time and detection accuracy relative to other methods.

5.3. Multi-Armed Bandit-based selection

The multi-armed bandit (MAB) approach is utilized in many studies to select clients in FL. During each training round, the player (representing the server in FL) selects one of the multiple arms (representing the clients in FL) to receive a reward with different probabilities of improving the global model. The MAB attempts to balance exploration (trying out new clients) and exploitation (selecting clients that yield the highest reward for the model's improvement).

Wenchao et al. [68] introduced an MAB-based online client scheduling method, which was used without knowledge of clients' statistical characteristics and wireless channel information. This method aims to minimize the latency of training using the upper confidence bound policy with non-IID client data; in contrast, the virtual queue technique and upper confidence bound policy are used for IID client data. Bo et al. [69] developed a method for scheduling clients for a wireless FL system using a MAB to reduce training latency and the training rounds. The proposed method is implemented without previous knowledge of the client's computing power and the condition of the wireless channel.

Elia et al. [70] developed a decentralized client selection methodology using a non-fixed multi-armed bandit. The client selection issue is framed as a sequential decision-making problem in which the decision to participate in training is determined independently by the clients rather than the server. The suggested method seeks to balance energy consumption and the efficacy of the global model. The researchers demonstrated the advantages of this strategy compared to the random method in reducing energy consumption and the number of rounds. Dan et al. [71] introduced an algorithm called Bandit Scheduling for FL (BSFL) for client selection. This algorithm also uses an MAB approach to improve learning performance by reducing training latency and preserving the model's generalization ability.

5.4. Clustering-based selection

This type of selection groups clients into clusters based on similarities, such as data type and size, device resources, performance, and behavior. Then, clients from each cluster are randomly selected or selected based on specific performance to participate in the FL training process. This approach aims to achieve a more balanced participation in model training, improve model convergence acceleration, and reduce energy costs.

Zhe et al. [72] proposed COCS (Context-aware Online Client Selection) for client selection in hierarchical federated learning. In this proposed context-aware approach, the network operator makes an online decision to select clients to participate in training through client and edge server couples. The proposed approach demonstrates high performance in the experiments that were conducted. Duanxiao et al. [73] proposed an approach called HCSFed for client selection using a clustering technique. This approach accelerates the global model convergence rate by minimizing the variance between clustered model updates. This proposed approach provides a strong convergence guarantee by minimizing variance and being more efficient than other methods.

Abdullatif et al. [74] introduced the clustered federated multitask learning approach by proposing a method for selecting and scheduling clients in two stages. The first stage involves the fairness principle of clustering clients into clusters at the beginning of training. The second stage involves greedily selecting clients with the best resources and the lowest arrival time from each cluster. This approach ensures enhancements in convergence rate and diminished training duration.

Minghong et al. [75] propose improving client selection during intermittent training participation via hierarchical FL to reduce energy and latency costs and accelerate model convergence. The proposed method was implemented through two plans. The first plan identified clients with a higher propensity to participate in the following training rounds. The second plan is considered a backup plan that selects backup clients when the clients specified in the first plan are unavailable.

5.5. Reputation and security-based selection

Malicious client models can degrade FL performance and compromise sensitive data privacy. Selecting reliable clients based on their reputation and security standards leads to the safety and security of the FL system, as well as improving the quality of the resulting model. Qinnan et al. [76] presented a strategy for selecting trusted clients in FL based on reputation assessment and leveraging blockchain technology. The reliability of clients is evaluated based on their historical reputation, which is stored on the blockchain. This approach improves model convergence and accuracy, as well as the potential to prevent privacy leakage of the client's reputation values. In FL settings where malicious clients and non-IID data are common, Rafael et al. [77] introduced a resilience-orientated method to client selection. The client selection mechanism in the suggested approach relies on the entropy and size of the client data. Malicious clients are identified, and their updates are eliminated from the aggregation through a centroid-based kernel alignment method. This method offers superior performance, stability, and flexibility.

Tao et al. [78] developed a method to maintain the integrity and security of an FL system when selecting clients by countering unreliable models of malicious clients. This method relies on two factors for client selection: the first is the security score, derived from the client's past performance; the second is the fairness score, determined during the aggregation process by measuring the client's participation rate. This study's results confirm that it effectively selects reliable clients fairly. William et al. [79] presented a client selection method that ensures privacy and offers high performance in network anomaly detection. The proposed method uses a fault tolerance strategy integrated with differential privacy. System constraints and model performance dynamically control the number of clients selected to participate in training. This approach improves model accuracy, reduces training time, and reduces noise by achieving differential privacy. It also handles client failures through a fault tolerance strategy.

6. Discussion

Several challenges may impact on the efficiency of the client selection process in FL. When clients have significantly disparate data distributions, their local model updates may be inconsistent with one another and the global model. The averaging of these updates may lead to slower convergence and a non-ideal global model. Moreover, non-IID data can result in client models diverging from the global model, causing instability and reduced accuracy. The diversity of client devices presents a complex issue for FL environments, resulting in varying local training durations that affect the overall convergence and fairness of the global model.

The computational variety among participating devices, resulting from differences in processor power, memory capacity, and energy resources, directly influences the time needed for each client to finish its local training iteration. The practical application of FL encounters another substantial challenge, especially when devices have restricted communication capabilities, which adversely impacts client selection efficiency and extends convergence durations. The selection of malicious clients poses a significant risk to the FL training process. Malicious clients can manipulate their training datasets to impair the model's performance or modify the weights of their local models before submission, adversely affecting the global model. Fairness in FL requires equal selection possibilities for all participating clients during the training process, avoiding situations where specific clients are consistently prioritized while others are marginalized. Disparities may emerge when clients with higher computing resources or high-quality data are persistently favored. In contrast, those with fewer capabilities or less representative data are marginalized or afforded minimal participation opportunities.

The selection of clients in FL is a significant challenge due to budget constraints and client heterogeneity. Researchers have investigated multiple techniques to tackle this challenge; client selection can significantly influence model quality, convergence speed, and fairness.

Greedy algorithms optimize client selection by considering resource availability and performance indicators, enhancing model accuracy and decreasing training latency in heterogeneous environments. Greedy client selection methods present a potential approach for addressing the issues of client heterogeneity and resource limitations in FL, resulting in enhanced model performance and efficiency. RL can be employed in personalized FL to address the challenges of data and system heterogeneity. Developing an FL client selection mechanism that balances exploitation and exploration is a complex task; deep reinforcement learning can facilitate optimal decision-making in intricate dynamic environments. Combining greedy selection techniques with RL strategies, exemplified by the enhanced DDQN algorithm, facilitates dynamic adaptation to client performance, hence augmenting the efficacy of client selection in FL.

MAB methods learn clients' status (e.g., latency distribution, generalization capability) to minimize training latency while preserving model generalization. MAB balances the exploration of diverse clients with the exploitation of acquired information to identify the most rewarding subset in each round. When MAB is implemented in a decentralized manner, clients can participate in the training without dependence on a central server, which may result in a more balanced approach between model accuracy and energy consumption. MAB-based client selection provides a flexible and efficient solution to the issues of client heterogeneity and resource limitations in FL. MAB algorithms optimize diverse objectives by balancing exploration and exploitation, including reducing training latency, enhancing model generalization, and reducing energy usage.

Clustering-based client selection in FL involves categorizing clients into clusters according to specific criteria and then selecting clients to participate in the training from each cluster. Clustering can be employed to select edge nodes to participate in clustered federated multitask learning, hence minimizing training latency and accelerating the convergence rate. Clustering-based client selection enhances FL by reducing variance, tackling heterogeneity, assuring fairness, and optimizing edge node selection.

A reliable reputation assessment framework can be developed to ensure high-quality client selection in FL, which includes assessing the reliability of candidate clients to guarantee a reliable FL system. Reputation can be utilized to identify reliable and trustworthy clients based on their historical behaviors. In general, reputation and security-based client selection can enhance FL by identifying reliable clients, mitigating privacy breaches, fostering mutual trust, and ensuring fairness. Blockchain, security metrics, and RL can be utilized to execute reputation and security-based client selection processes.

Table 3 illustrate the comparison of key improvements, models, and datasets in the surveyed clients' selection methods, while **Table 4** provides a general comparison of these methods in terms of core idea, advantages, limitations/disadvantages, and use cases.

7. Future research directions

Advancements significantly influence the trajectory of FL in client selection methodologies, which are pivotal in addressing the inherent challenges of heterogeneous data distributions and resource limitations prevalent in decentralized environments. There are several future directions for client selection and improving performance in FL. The subsequent sections delineate essential elements of these directions.

7.1. Transfer Learning-based client selection

This selection method can evaluate the potential clients' contributions despite little information on their historical input.

7.2. Fairness-based client selection

There are future directions to ensure equitable participation of all clients during model training and reduce bias in client selection.

Table 3. Comparison of client selection methods in federated learning.

Ref. No	Client selection method	Key Improvement	ML model	Dataset
[60]	Greedy selection	• Improve convergence speed and accuracy	MLP classifier, CNN	MNIST, Fashion-MNIST, CIFAR10
[61]		• Improve model performance • Ensure timely updates • Mitigate stragglers' impact	ResNet-18	CIFAR-10, CIFAR-100
[62]		• Provide high-performance ML models with fast training	DNN	CIFAR-10, Fashion MNIST
[63]		• Improve model quality • Ensure fairness in client selection	CNN	MNIST, CIFAR-10
[64]	RL-based selection	• Enhance model accuracy and fairness • Improve model convergence speed	CNN	MNIST
[65]		• Improve efficiency and accuracy in heterogeneous environments	MobileNet-v244, gated recurrent unit (GRU)	CIFAR-10, CIFAR-100, Fashion-MNIST, MobiAct
[66]		• Reduce communication rounds • Improve convergence speed • Strength in heterogeneous environments	DNN	CIFAR-10, CIFAR-100, NICO, Tiny ImageNet
[67]		• Improve accuracy • Improve execution time	CNN	X-ray images of COVID-19 patients
[68]	Multi-Armed Bandit-based selection	• Improve client scheduling • Enhance learning efficiency	Multinomial logistic regression	MNIST
[69]		• Reduce training rounds • Improve accuracy • Achieve lower training loss	Multilayer perceptron	MNIST
[70]		• Reduce the number of rounds • Reduce energy consumption	ResNet-18	CIFAR-10, CIFAR-100
[71]		• Minimize training latency • Maintain model generalization	Linear regression, CNN	Fashion-MNIST, CIFAR-10
[72]	Clustering-based selection	• High performance • Provide theoretical assurances for both strongly convex and non-convex HFL	Logistic regression, CNN	MNIST, CIFAR-10
[73]		• Accelerate convergence • Effectiveness in various settings • Require fewer rounds to achieve target accuracy	Fully connected network, Logistic regression	MNIST, CIFAR-10, FMNIST
[74]		• Improve convergence speed • Ensuring correct clustering	CNN,DNN	FEMNIST, CIFAR-10
[75]		• Enhance learning efficiency • Improve model accuracy • Reduce system costs	CNN, LeNet5, ResNet18	real-world EUA, MNIST, FMNIST , CIFAR-10, CIFAR-100
[76]	Reputation and security-based selection	• Reduce costs • Improve model accuracy and convergence speed	CNN, MLP	MNIST, CIFAR-10
[77]		• Enhance model accuracy • Resilience against malicious clients	CNN	CIFAR-10, FMNIST, MNIST

[78]		• Enhance model security and fairness • Mitigate data-poisoning attacks	CNN	MNIST, FMNIST
[79]		• Improve accuracy • Reduce training time	NN	UNSW-NB15, ROAD

Table 4. General comparison of clients' selection methods in terms of the core idea, advantages, limitations/ disadvantages, and use cases.

Ref. No	Client selection method	Core Idea	Advantages	Limitations/ disadvantages	Use Cases
[60], [61], [62], [63]	Greedy selection	Select clients based on available resources and performance indicators.	• Reduce training latency • Improve model accuracy • Improve model quality	• Repeatedly exclude weaker clients • clients selection bias • Heterogeneous client characteristics lead to ineffective training processes and complicate the selection process	• Static and heterogeneous environments with limited resources • Small to medium-scale environments • Uncomplicated applications
[12] [64], [65], [66], [67]	RL-based selection	Enhances complex decision-making processes by interacting with a specific environment and uses RL/DRL algorithms to balance exploitation and exploration in client selection.	• Adapt to dynamic environments. • Achieve near-optimal client selection over extended training horizons • Handle system and data heterogeneity • Strength in heterogeneous environments	• High computational cost • Complex design and train	• Dynamic environments. • Large-scale systems • Scenarios with high heterogeneity • IoT and edge computing
[68], [69], [70], [71]	Multi-Armed Bandit-based selection	Learns the client's status (e.g., latency, generalization capability) and balances exploration and exploitation in client selection.	• Enhance generalization and accuracy • Reduce energy consumption • Minimize training latency • Flexible (centralized architecture or decentralized architecture)	• Difficult to maintain an ideal exploration–exploitation balance • Sensitive to fast-changing client conditions	• Heterogeneous resource environments • Systems requiring flexibility • Applications with unstable data, such as mobile or sensor networks
[16], [72], [73], [74], [75]	Clustering-based selection	Groups clients into clusters and selects representatives from each group.	• Ensure fairness. • Accelerate convergence • Optimize edge node selection. • Sometimes used to reduce variance and tackling heterogeneity	• Require precise clustering criteria • Complexity when updating clusters	• Large-scale FL systems • Edge computing applications With Hierarchical FL • Environments with many clients • When clients are highly heterogeneous in terms of data distribution
[76], [77], [78], [79]	Reputation and security-based selection	Selects clients based on trustworthiness, previous behavior, and security metrics (e.g., blockchain, reputation scores, differential privacy).	• Ensure reliable clients. • Mitigate privacy/security risks. • Enhance fairness and trust.	• Require an accurate reputation framework • Overhead for monitoring and evaluation	• Security applications • Privacy-sensitive domains (healthcare, finance) • When there is a risk of malicious clients or data/model poisoning attacks • Scenarios requiring client filtering based on past reputation or trustworthiness

7.3. Dynamic environment

Future research is poised to delve deeper into adaptive client selection strategies that dynamically adjust to the fluctuating availability and computational capabilities of participating clients. Deep reinforcement learning techniques can be applied to client selection in dynamic environments, particularly those with volatile resources.

7.4. Integrating Uncertainty Factors

Future research may explore the incorporation of uncertainty in clients' local computational and communicative resources, enhancing the robustness of FL systems.

7.5. Developing client selection algorithms

Client selection algorithms can be developed that rely on several criteria, including statistical similarity, data quality, client availability, and system and communication resources, to determine which clients to include in the FL training. In addition to developing clients' selection mechanisms in real-world environments such as smart cities.

7.6. Incentive mechanisms

Develop incentive mechanisms, such as game-theoretic models, to encourage clients to participate in training, especially those with limited resources or less reliable connections, which provide features to ensure equity and inclusion in FL.

7.7 Scalability

Future research may explore hierarchical federated learning architectures to improve scalability in large-scale environments with numerous clients. Enhancing communication and aggregation among multi-tier servers can reduce latency and improve convergence efficiency.

7.8 robustness

Future research may incorporate blockchain-based consensus techniques to improve robustness against malicious or unreliable clients. Moreover, lightweight blockchain frameworks may be investigated to reconcile security with efficiency in extensive FL systems.

Conclusion

Federated learning enables clients, including nodes or devices, to independently train a model on their local data rather than transmitting that data to a centralized server. In this survey, we presented the challenges facing the client selection process, including statistical heterogeneity, system heterogeneity, and the communication costs for participating clients during training. Moreover, selecting malicious clients presents a significant risk to FL's performance. Furthermore, unfairness can arise when highly capable clients are always selected, while less capable clients are excluded from the selection process. Selecting the best clients in FL is critical to improving the convergence speed and accuracy of the final model, this requires careful client selection approaches. This survey presents the latest client selection methods in FL, including greedy selection, reinforcement learning-based selection, multi-armed bandit-based selection, clustering-based selection, and reputation and security-based selection. We then present a general comparison of these methods, which acts as a guide for researchers when choosing a client selection method in FL. Finally, this survey presents opportunities for future directions in client selection methods and improving performance in FL.

Acknowledgement

The authors would express they're thanks to college of computer science and mathematics, University of Mosul to support this report.

Conflict of interest

None.

References

[1] H. A. Altememi and Y. F. Al-Irhaim, "A Comparative Study for Speech Summarization Based on Machine Learning: A Survey," *AL-Rafidain Journal of Computer Science and Mathematics*, vol. 16, no. 2, pp. 89–96, 2022, doi: 10.33899/csmj.2022.176595.

[2] S. AbdulRahman, H. Tout, H. Ould-Slimane, A. Mourad, C. Talhi, and M. Guizani, "A survey on federated learning: The journey from centralized to distributed on-site learning and beyond," IEEE Internet Things J, vol. 8, no. 7, pp. 5476–5497, 2020.

[3] W. Hadeed and D. Abdullah, "Real-Time Based Big Data and E-Learning: A Survey and Open Research Issues," AL-Rafidain Journal of Computer Sciences and Mathematics, vol. 15, no. 2, pp. 225–243, 2021.

[4] B. Liu, N. Lv, Y. Guo, and Y. Li, "Recent advances on federated learning: A systematic survey," Neurocomputing, p. 128019, 2024.

[5] W. Y. B. Lim, J. S. Ng, Z. Xiong, D. Niyato, and C. Miao, Federated learning over wireless edge networks. Springer, 2022.

[6] S. K. Lo, Q. Lu, C. Wang, H.-Y. Paik, and L. Zhu, "A systematic literature review on federated machine learning: From a software engineering perspective," ACM Computing Surveys (CSUR), vol. 54, no. 5, pp. 1–39, 2021.

[7] B. Xu, W. Xia, J. Zhang, T. Q. S. Quek, and H. Zhu, "Online client scheduling for fast federated learning," IEEE Wireless Communications Letters, vol. 10, no. 7, pp. 1434–1438, 2021.

[8] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated Learning: Challenges, Methods, and Future Directions," Aug. 2019, doi:

10.1109/MSP.2020.2975749.

[9] G. Shirvani, S. Ghasemshirazi, and B. Beigzadeh, "Federated Learning: Attacks, Defenses, Opportunities and Challenges," in 2024 11th International Symposium on Telecommunications (IST), IEEE, 2024, pp. 470–479.

[10] I. Schoinas et al., "Federated learning: Challenges, SoTA, performance improvements and application domains," IEEE Open Journal of the Communications Society, 2024.

[11] E. Dritsas and M. Trigka, "Federated Learning for IoT: A Survey of Techniques, Challenges, and Applications," Journal of Sensor and Actuator Networks, vol. 14, no. 1, p. 9, 2025.

[12] M. Hosseinzadeh, A. Hemmati, and A. M. Rahmani, "Federated learning-based IoT: a systematic literature review," International Journal of Communication Systems, vol. 35, no. 11, p. e5185, 2022.

[13] H. G. Abreha, M. Hayajneh, and M. A. Serhani, "Federated learning in edge computing: a systematic survey," Sensors, vol. 22, no. 2, p. 450, 2022.

[14] X. Ma, J. Zhu, Z. Lin, S. Chen, and Y. Qin, "A state-of-the-art survey on solving non-iid data in federated learning," Future Generation Computer Systems, vol. 135, pp. 244–258, 2022.

[15] L. Fu, H. Zhang, G. Gao, M. Zhang, and X. Liu, "Client selection in federated learning: Principles, challenges, and opportunities," IEEE Internet Things J, vol. 10, no. 24, pp. 21811–21819, 2023.

[16] W. Bouras, M. Benouis, K. E. Heraguemil, B. Brahim, and S. Akhrouf, "Analysis Study of Participant Selection Methods in Federated Learning," in 2024 2nd International Conference on Electrical Engineering and Automatic Control (ICEEAC), IEEE, 2024, pp. 1–6.

[17] M. Aledhari, R. Razzak, R. M. Parizi, and F. Saeed, "Federated learning: A survey on enabling technologies, protocols, and applications," IEEE Access, vol. 8, pp. 140699–140725, 2020.

[18] C. Papadopoulos, K.-F. Kollias, and G. F. Fragulis, "Recent Advancements in Federated Learning: State of the Art, Fundamentals, Principles, IoT Applications and Future Trends," Future Internet, vol. 16, no. 11, p. 415, 2024.

[19] Z. Xu et al., "Federated learning of gboard language models with differential privacy," arXiv preprint arXiv:2305.18465, 2023.

[20] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," Knowl Based Syst, vol. 216, p. 106775, 2021.

[21] W. Riviera, I. B. Galazzo, and G. Menegaz, "FLavourite: Horizontal and Vertical Federated Learning setting comparison," IEEE Access, 2025.

[22] A. Z. Tan, H. Yu, L. Cui, and Q. Yang, "Towards personalized federated learning," IEEE Trans Neural Netw Learn Syst, vol. 34, no. 12, pp. 9587–9603, 2022.

[23] P. Kairouz et al., "Advances and open problems in federated learning," Foundations and trends® in machine learning, vol. 14, no. 1–2, pp. 1–210, 2021.

[24] A. Qayyum, K. Ahmad, M. A. Ahsan, A. Al-Fuqaha, and J. Qadir, "Collaborative federated learning for healthcare: Multi-modal covid-19 diagnosis at the edge," IEEE Open Journal of the Computer Society, vol. 3, pp. 172–184, 2022.

[25] T. Zhang, L. Gao, C. He, M. Zhang, B. Krishnamachari, and A. S. Avestimehr, "Federated learning for the internet of things: Applications, challenges, and opportunities," IEEE Internet of Things Magazine, vol. 5, no. 1, pp. 24–29, 2022.

[26] I. M. Ahmed and M. Y. Kashmoola, "Investigated insider and outsider attacks on the federated learning systems," in 2022 IEEE International Conference on Communication, Networks and Satellite (COMNETSAT), IEEE, 2022, pp. 438–443.

[27] I. M. Ahmed and M. Y. Kashmoola, "CCF based system framework in federated learning against data poisoning attacks," Journal of Applied Science and Engineering, vol. 26, no. 7, pp. 971–979, 2022.

[28] F. Lai, X. Zhu, H. V Madhyastha, and M. Chowdhury, "Oort: Efficient federated learning via guided participant selection," in 15th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 21), 2021, pp. 19–35.

[29] Y. J. Cho, J. Wang, and G. Joshi, "Client selection in federated learning: Convergence analysis and power-of-choice selection strategies," arXiv preprint arXiv:2010.01243, 2020.

[30] B. Guler and A. Yener, "Sustainable Federated Learning," Feb. 2021, [Online]. Available: http://arxiv.org/abs/2102.11274

[31] M. Ye, X. Fang, B. Du, P. C. Yuen, and D. Tao, "Heterogeneous federated learning: State-of-the-art and research challenges," ACM Comput Surv, vol. 56, no. 3, pp. 1–44, 2023.

[32] J. Li, T. Mahmoodi, and H.-K. Lam, "Distributed Learning in Heterogeneous Environment: federated learning with adaptive aggregation and computation reduction," in ICC 2023-IEEE International Conference on Communications, IEEE, 2023, pp. 1976–1981.

[33] V. N. Iyer, "A review on different techniques used to combat the non-IID and heterogeneous nature of data in FL," arXiv preprint arXiv:2401.00809, 2024.

[34] Z. Huang, "Cluster-Based Two-Stage Client Selection Strategy for Personalized Federated Learning," in 2024 5th International Conference on Information Science, Parallel and Distributed Systems (ISPDS), IEEE, 2024, pp. 136–139.

[35] H. An and C. Zhu, "Optimal Client Selection for Wireless Federated Learning," in 2024 IEEE 16th International Conference on Advanced Infocomm Technology (ICAIT), IEEE, 2024, pp. 54–59.

[36] W. Marfo, D. K. Tosh, and S. V Moore, "Adaptive client selection in federated learning: A network anomaly detection use case," arXiv preprint arXiv:2501.15038, 2025.

[37] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," IEEE Signal Process Mag, vol. 37, no. 3, pp. 50–60, 2020.

[38] S. Reddi et al., "Adaptive federated optimization," arXiv preprint arXiv:2003.00295, 2020.

[39] L. Jiang, X. B. Zhai, J. Zhu, and J. Li, "Resource-Constrained Client Selection via Cluster-Driven Federated Learning for Internet of Things," in 2024 IEEE Wireless Communications and Networking Conference (WCNC), IEEE, 2024, pp. 1–6.

[40] H. An and C. Zhu, "Optimal Client Selection for Wireless Federated Learning," in 2024 IEEE 16th International Conference on Advanced Infocomm Technology (ICAIT), IEEE, 2024, pp. 54–59.

[41] J. Kang, Z. Xiong, D. Niyato, Y. Zou, Y. Zhang, and M. Guizani, "Reliable federated learning for mobile networks," IEEE Wirel Commun, vol. 27, no. 2, pp. 72–80, 2020.

[42] P. Kairouz et al., "Advances and open problems in federated learning," Foundations and trends® in machine learning, vol. 14, no. 1–2, pp. 1–210, 2021.

[43] O. R. A. Almanifi, C.-O. Chow, M.-L. Tham, J. H. Chuah, and J. Kanesan, "Communication and computation efficiency in federated learning: A survey," Internet of Things, vol. 22, p. 100742, 2023.

[44] Y. Zhou et al., "The role of communication time in the convergence of federated edge learning," IEEE Trans Veh Technol, vol. 71, no. 3, pp. 3241–3254, 2022.

[45] W. Deng, X. Chen, X. Li, and H. Zhao, "Adaptive federated learning with negative inner product aggregation," IEEE Internet Things J, vol. 11, no. 4, pp. 6570–6581, 2023.

[46] O. Shahid, S. Pouriyeh, R. M. Parizi, Q. Z. Sheng, G. Srivastava, and L. Zhao, "Communication efficiency in federated learning: Achievements and challenges," arXiv preprint arXiv:2107.10996, 2021.

[47] Y. Wang, L. Lin, and J. Chen, “Communication-efficient adaptive federated learning,” in International conference on machine learning, PMLR, 2022, pp. 22802–22838.

[48] J. Shu et al., “Clustered federated multitask learning on non-IID data with enhanced privacy,” IEEE Internet Things J, vol. 10, no. 4, pp. 3453–3467, 2022.

[49] S. Yu and L. Cui, Security and Privacy in Federated Learning. Springer, 2023.

[50] S. Almutairi and A. Barnawi, “Federated learning vulnerabilities, threats and defenses: A systematic review and future directions,” Internet of Things, vol. 24, p. 100947, 2023.

[51] A. Q. Saeed et al., “Integrating Three Machine Learning Algorithms in Ensemble Learning Model for Improving Content-based Spam Email Recognition,” Journal of Soft Computing and Data Mining, vol. 5, no. 2, pp. 188–196, 2024.

[52] I. M. Ahmed and M. Y. Kashmoola, “Threats on machine learning technique by data poisoning attack: A survey,” in International Conference on Advances in Cyber Security, Springer, 2021, pp. 586–600.

[53] T. H. Rafi, F. A. Noor, T. Hussain, and D.-K. Chae, “Fairness and privacy preserving in federated learning: A survey,” Information Fusion, vol. 105, p. 102198, 2024.

[54] T. Gafni, N. Shlezinger, K. Cohen, Y. C. Eldar, and H. V. Poor, “Federated learning: A signal processing perspective,” IEEE Signal Process Mag, vol. 39, no. 3, pp. 14–41, 2022.

[55] Y. Huang and C. Hu, “Toward data heterogeneity of federated learning,” arXiv preprint arXiv:2212.08944, 2022.

[56] M. Fang, J. Liu, N. Z. Gong, and E. S. Bentley, “Aflguard: Byzantine-robust asynchronous federated learning,” in Proceedings of the 38th Annual Computer Security Applications Conference, 2022, pp. 632–646.

[57] Y. Sun, S. Si, J. Wang, Y. Dong, Z. Zhu, and J. Xiao, “A fair federated learning framework with reinforcement learning,” in 2022 International Joint Conference on Neural Networks (IJCNN), IEEE, 2022, pp. 1–8.

[58] S. Kim, S. Bae, H. Song, and S.-Y. Yun, “Re-thinking federated active learning based on inter-class diversity,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 3944–3953.

[59] C. Lu, W. Ma, R. Wang, S. Deng, and Y. Wu, “Federated learning based on stratified sampling and regularization,” Complex & Intelligent Systems, vol. 9, no. 2, pp. 2081–2099, 2023.

[60] P. Singhal, S. R. Pandey, and P. Popovski, “Greedy shapley client selection for communication-efficient federated learning,” IEEE Networking Letters, vol. 6, no. 2, pp. 134–138, 2024.

[61] J. Liu, Z. Chang, and Y.-C. Liang, “Age-Based Device Selection and Transmit Power Optimization in Over-the-Air Federated Learning,” arXiv preprint arXiv:2501.01828, 2025.

[62] T. Nishio and R. Yonetani, “Client selection for federated learning with heterogeneous resources in mobile edge,” in ICC 2019-2019 IEEE international conference on communications (ICC), IEEE, 2019, pp. 1–7.

[63] M. Zhang, H. Zhao, S. Ebron, R. Xie, and K. Yang, “Multi-criteria client selection and scheduling with fairness guarantee for federated learning service,” arXiv preprint arXiv:2312.14941, 2023.

[64] A. Tariq, A. Lakas, F. Sallabi, T. Qayyum, M. A. Serhani, and E. Barka, “Empowering trustworthy client selection in edge federated learning leveraging reinforcement learning,” in Proceedings of the Eighth ACM/IEEE Symposium on Edge Computing, 2023, pp. 372–377.

[65] H. Yang, J. Li, M. Hao, W. Zhang, H. He, and A. K. Sangaiah, “An efficient personalized federated learning approach in heterogeneous environments: a reinforcement learning perspective,” Sci Rep, vol. 14, no. 1, p. 28877, 2024.

[66] X. Meng, Y. Li, J. Lu, and X. Ren, “An optimization method for non-IID federated learning based on deep reinforcement learning,” Sensors, vol. 23, no. 22, p. 9226, 2023.

[67] G. Rjoub, O. A. Wahab, J. Bentahar, R. Cohen, and A. S. Bataineh, “Trust-augmented deep reinforcement learning for federated learning client selection,” Information Systems Frontiers, vol. 26, no. 4, pp. 1261–1278, 2024.

[68] W. Xia, T. Q. S. Quek, K. Guo, W. Wen, H. H. Yang, and H. Zhu, “Multi-armed bandit-based client scheduling for federated learning,” IEEE Trans Wirel Commun, vol. 19, no. 11, pp. 7108–7123, 2020.

[69] B. Xu, W. Xia, J. Zhang, T. Q. S. Quek, and H. Zhu, “Online client scheduling for fast federated learning,” IEEE Wireless Communications Letters, vol. 10, no. 7, pp. 1434–1438, 2021.

[70] E. Guerra, M. Miozzo, and P. Dini, “Fully decentralized client selection for energy-efficient federated learning,” in 2023 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech), IEEE, 2023, pp. 595–601.

[71] D. Ben Ami, K. Cohen, and Q. Zhao, “Client selection for generalization in accelerated federated learning: A multi-armed bandit approach,” IEEE Access, 2025.

[72] Z. Qu, R. Duan, L. Chen, J. Xu, Z. Lu, and Y. Liu, “Context-aware online client selection for hierarchical federated learning,” IEEE Transactions on Parallel and Distributed Systems, vol. 33, no. 12, pp. 4353–4367, 2022.

[73] D. Song et al., “Fast heterogeneous federated learning with hybrid client selection,” in Uncertainty in Artificial Intelligence, PMLR, 2023, pp. 2006–2015.

[74] A. M. Albaseer, M. Abdallah, A. Al-Fuqaha, A. M. Seid, A. Erbad, and O. A. Dobre, “Fair selection of edge nodes to participate in clustered federated multitask learning,” IEEE Transactions on Network and Service Management, vol. 20, no. 2, pp. 1502–1516, 2023.

[75] M. Wu et al., “Towards Seamless Hierarchical Federated Learning under Intermittent Client Participation: A Stagewise Decision-Making Methodology,” arXiv preprint arXiv:2502.09303, 2025.

[76] Q. Zhang, Q. Ding, J. Zhu, and D. Li, “Blockchain empowered reliable federated learning by worker selection: A trustworthy reputation evaluation method,” in 2021 IEEE wireless communications and networking conference workshops (WCNCW), IEEE, 2021, pp. 1–6.

[77] R. Veiga et al., “A robust client selection mechanism for federated learning environments,” Journal of the Brazilian Computer Society, vol. 30, no. 1, pp. 444–455, 2024.

[78] T. Wan, S. Feng, W. Liao, N. Jiang, and J. Zhou, “A Secure and Fair Client Selection Based on DDPG for Federated Learning,” International Journal of Intelligent Systems, vol. 2024, no. 1, p. 2314019, 2024.

[79] W. Marfo, D. K. Tosh, and S V Moore, “Adaptive client selection in federated learning: A network anomaly detection use case,” arXiv preprint arXiv:2501.15038, 2025.

Abbreviations

BSFL	Bandit Scheduling for Federated learning
CNNs	Convolutional Neural Networks
COCS	Context-aware Online Client Selection
DNN	Deep Neural Network

FL	Federated learning
IOT	Internet of Things
MAB	Multi-Armed Bandit
ML	Machine learning
MLP	Multilayer perceptron
NN	Neural Network
non-IID	Non-independent and -identical Distributed
RL	Reinforcement Learning