Learning and Nonlinear Models - Journal of the Brazilian Society on Computational Intelligence (SBIC), Vol. 23, Iss. 1, pp. 49–66, 2025
https://doi.org/10.21528/lnlm-vol23-no1-art4 © Brazilian Society on Computational Intelligence

ENBASE: ENHANCING IMAGE CLASSIFICATION IN IoT SCENARIOS THROUGH ENTROPY-BASED SELECTION OF NON-IID DATA

Ernesto Gurgel Valente Neto
Graduate Program in Teleinformatics Engineering
Federal University of Ceará, Center of Technology
Campus of Pici, Fortaleza, Ceará, Brazil
Email: gurgelvalente@alu.ufc.br

Solon Alves Peixoto
Federal University of Ceará, Campus of Itapajé
Itapajé, Ceará, Brazil
Email: solon.alves@ufc.br

Julio César Santos dos Anjos
Federal University of Ceará, Campus of Itapajé
Itapajé, Ceará, Brazil
and
Graduate Program in Teleinformatics Engineering
Center of Technology, Campus of Pici, Fortaleza, Ceará, Brazil
Email: jcsanjos@ufc.br

Abstract – This study presents an analysis of the scalability and dispersion of results in Federated Learning (FL) using two algorithms: EnBaSe, based on entropy, and Random, a random selection approach. The Random algorithm ensures that each member of the population has an equal probability of inclusion. At the same time, EnBaSe calculates the information gain and selects the most informative samples for the neural network. Both algorithms were applied in federated learning scenarios with data distributed non-independently and non-identically (Non-IID). The MNIST, Fashion-MNIST, CIFAR-10, and CIFAR-100 datasets were used for the evaluation, representing different levels of computer vision classification. The results show that the EnBaSe algorithm achieves high accuracy while halving computational and energy costs compared to training with all samples from the datasets. In addition, EnBaSe demonstrated greater resilience to variability, showing low variance and a more stable distribution, especially in Internet of things (IoT) environments with limited computational resources.

Keywords – Data Quality, Deep Learning, Entropy, Federated Learning, IoT.

1 Introduction

In the field of Machine Learning (ML), information quality, quantity, and relevance are closely linked to using computational resources. This relationship is evident in two main scenarios: centralized and decentralized. For example, in a centralized scenario, data is aggregated in a single location, involving data transfer, computation, and energy costs. In contrast, in a decentralized scenario, data is processed on devices with limited resources, lower storage capacity, and limited communication bandwidth.

As a result, the large number of devices and high data volume require high-speed processing and analysis to generate valuable insights while complying with the legal requirements to protect private and confidential data [1]. These aspects involve technical and system design challenges [2]. Consequently, data quality is crucial in machine learning [3], particularly in fields with data limitations, such as drug discovery, which often operates with small and limited datasets [4].

In addition to data quality challenges, data heterogeneity in Federated Learning (FL) presents significant challenges, particularly with the Non-Independent and Identically Distributed (non-iid) distributions, which directly affect the accuracy and convergence of the FL models [5]. The presence of non-iid data, which often comes up as a result of temporal dependencies, also poses risks of bias and inconsistent performance in trained models [6]. These challenges affect the selection of clients and developing effective model-fusion methods [7].

To illustrate the influence of entropy on data quality [8], and to address ambiguity [9] and minimize computational costs [10], these works find to minimize the communication delays and reduce the algorithm execution time.

Motivated by these studies on entropy properties, we analyzed the image distributions. We observed that the samples provided long tail behavior with extreme values, which contributed to the dispersion around the mean. The proposed Entropy-Based Selection (EnBaSe) algorithm deals with the dispersion of anomalous values and the distortion of the tails in non-iid data. Also, the approach around data quality evaluation inside the class does not change the distribution dispersion and maintains the sample domain representativity. The impact of the approach is the maintenance of stochastic distribution properties and the decrease of non-significant computation from nodes, which, as a result, enables increased performance of FL on edge computing and saves energy costs.

The main contributions: Building on previous research, this study lies in a comprehensive analysis of the EnBaSe algorithm's performance within the context of FL under non-iid scenarios. The investigation includes a detailed comparison of its distribution, range, and extreme values, and a thorough assessment of the stability, reliability, and scalability of the achieved results.

In addition to this introduction, the structure of this work is organized as follows: Section 1 presents the introduction; Section 2 discusses the background and related work; Section 3 describes the proposed model; Section 4 details the materials and methods; Section 5 presents the evaluations; Section 6 shows the discussion, conclusions and future works.

2 Background and Related Work

Internet of Things (IoT) combines devices with sensors, processing capabilities, and software, enabling data exchange between devices and systems over the Internet. This integration of electronics, communication, and computer engineering creates IoT systems that facilitate interaction with everyday objects [11,12]. Its development is driven by some technologies, as follows:

i. Cloud Computing;

ii. Edge Computing;

iii. Distributed Intelligence;

iv. Security and Privacy;

These advancements in IoT technologies generate vast amounts of data, which have contributed to advancements in ML across various sectors, such as forecasting, data analysis, patient monitoring, healthcare, insurance, transportation, marketing, and automation [13,14], as well as diagnostic systems, personalized prescriptions, and integration with IoT [15], allows devices to learn from data and, at the same time, ensures robust privacy support [16].

2.1 Statistical Nature of Data

Homogeneous data is characterized by sharing quantitative or qualitative properties, which implies acceptable or predictable variation, reflecting a similar trend. Thus, when data has statistically homogeneous characteristics, neural networks tend to converge to optimal values of the Stochastic Gradient Descent (SGD) [17].

Unlike homogeneous data, which exhibit uniformity, heterogeneous data display high variability in types and formats [18]. These data often include variability, outliers, and inconsistencies in format and meaning [19]. For example, the diversity of IoT sensors and data-collection devices results in significant variability and imbalance in the generated data, creating heterogeneous data [20].

The concept of Independent and Identically Distributed (iid) data further underscores the importance of data consistency. Iid data refers to independent observations that follow the same probabilistic distribution, which enhances model convergence and performance by maintaining uniformity across training samples [21]. In ML, such data are often processed on centralized servers, assuming homogeneity. In this context, models tend to converge to a global optimum because of the similarity in sample distributions [22].

Conversely, non-iid data introduce statistical heterogeneity, with variations in sample quantity and distribution [23]. In this scenario, each node or device has a unique dataset. Variability in data quantity and classes across devices introduces bias, complicating model training [24,25].

Entropy, a central concept in information theory, reflects the level of unpredictability or disorder in a communication channel and is a quantitative tool for measuring uncertainty.

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i) \tag{1}$$

In the formula, X represents all possible symbol values, $p(x_i)$ is the probability of the i-th value, and $\log_2 p(x_i)$ represents the logarithm base 2 of $p(x_i)$, with entropy measured in bits.

Entropy quantifies data uncertainty and helps assess redundancy, making it a key measure of informational value [8,26]. Finally, entropy assesses the degree of uncertainty within a set of possible events, quantifying both the information gained and the significance of that information within a system.

Another mathematical model commonly applied in neural networks is a statistical and probabilistic technique used in ML is the **Random sampling** [27], which contributes to promoting diversity in sample selection and allows the neural network to learn different aspects of the data, which leads to more accurate predictions [28]. This technique consists of randomly selecting a set of data so that each element has an equal or known probability of being selected. In this way, the aim is to create a representative set of data, avoiding bias from non-random choices. Such an approach enhances model robustness by decreasing the dependency on specific sample configurations [29].

In addition to promoting robustness, random sampling also mitigates bias and increasing fairness in neural network training, random sampling improves the likelihood of finding global solutions and facilitates capturing general features and patterns, promoting generalization and reducing overfitting [30].

2.2 Related Work

Table 1 represents a summary of the main work related to Deep Learning (DL) and IoT, with machine learning and edge computing. These studies apply different solutions and seek to improve performance and efficiency in federated learning.

Table 1: Summary of Related Work - Motivation.

Author	**Year**	**DL**	**Edge**	**IoT**	**Opt.**	**Adj.**	**Analysis**
Deng et al. [31]	2021	x	x	x	x	x	x
Liu et al. [32]	2021	x		x	x		x
Al-Saedi et al. [33]	2022	x	x	x	x		x
Tan et al. [34]	2022	x			x	x	x
Yu et al. [35]	2022				x	x	x
Wolfrath et al. [36]	2022	x	x	x	x		x
Qi et al. [37]	2022	x			x		x
Zhang et al. [38]	2023	x			x	x	x
Wu et al. [39]	2023			x		x	
Sun et al. [40]	2023				x		x
Tu et al. [41]	2023	x	x	x	x		x
Hossain et al. [42]	2023		x	x	x		x
Tao et al. [43]	2023	x	x	x	x		x
Adjei-Mensah et al. [44]	2024	x			x		x
Wu et al. [45]	2024	x			x	x	x
Hamidi et al. [46]	2024	x	x	x	x	x	x
Our Model	2024	x	x	x		x	x

Note: DL: Deep Learning, Edge: Edge Computing, IoT: Internet of Things, Opt.: Algorithm Optimization, Adj.: Automatic Adjustment, Analysis: Data/Image Analysis or Data Quality Analysis.

FedWNS [41] utilizes node selection based on data distribution, enabling the selection of higher-quality samples in both iid and non-iid environments. While FedWNS emphasizes node selection based on data distribution, this study prioritizes essential data, reducing data transfer requirements and improving the model's performance.

FedAVO [42] enhances communication efficiency in FL by reducing overhead. Unlike FedAVO, EnBaSe focuses on improving data selection and quality by filtering high-quality data, thereby reducing overhead and computational and energy costs.

FedCo [33] utilizes cluster optimization to enhance communication efficiency in FL, effectively managing and reducing communication overhead. This approach improves model accuracy and minimizes the need for data retransmission, enabling FedCo to handle non-iid data variability more effectively.

Additionally, the algorithm proposed by [35] automatically adjusts model weights to optimize performance, enhancing both efficiency and accuracy in learning. This technique is particularly relevant for FL environments, as it improves the effectiveness of model training while reducing the need for frequent manual interventions.

Similarly, preconditioned FL introduces a method to optimize learning through preconditioning, aiming to improve model performance [43]. This approach involves preconditioning the learning environment or data to facilitate training.

An entropy-based approach prioritizes informative data to address heterogeneity and improve client clustering [36]. Similar strategies for client clustering based on non-iid data are explored in [47] and [38]. The EnBaSe filter emphasizes high-quality data before training, reducing data transfer and associated costs.

Building on these methods, the FAIR model filters out low-quality updates based on learning history, optimizing global learning and performance [31]. Similarly, FederaSER mitigates data poisoning by leveraging historical updates to guard against low-quality data and enhance training [32].

In contrast to these approaches, another study introduced a Blockchain-based FL model (BFL) that incorporates mechanisms to improve reputation and quality in model aggregation. This model aims to increase process transparency and reduce the risks of data leakage and manipulation [37]. The security and transparency layers integrated into the aggregation process are essential for ensuring the reliability of the aggregated models.

Moreover, Per-FedAvg utilizes meta-learning to balance global performance with client-specific personalization, effectively mitigating drift in local models [34].

These methods primarily seek to achieve efficiency in communication, optimization in client selection, robustness, and resilience in scenarios where data security is critical. They thus serve as a motivation and basis for future research by pointing out the main challenges discussed in the current literature on IoT and FL. In this way, they present an overview of the state of the art. In parallel, other approaches have been based on analyzing the system's homogeneity and the entropy of the neural network data, as shown in Table 2.

Table 2: Summary of Related Work - Similar approaches.

Author	Year	Application Scenario	Advantages	Disadvantages
Itahara et al. [48]	2021	IoT, FL, and non-IID	Robustness against attacks and noise	Loss of Accuracy
Li et al. [8]	2022	Agricultural pest recognition	Reduces redundancy in datasets	Restricted to multi-class classification
Condori Bustincio et al. [9]	2023	Heterogeneity, communication overhead	Reduces communication overhead	Limited generalization on datasets
Zhang et al. [49]	2023	Protein structure prediction	Robustness in feature selection	High complexity, requires fine-tuning
Orlandi et al. [10]	2023	IoT devices	Mitigates non-IID impacts, reduces energy consumption	Slight reduction in precision
Hamidi et al. [46]	2024	Medical diagnosis	Better Accuracy on unbalanced datasets	Increased complexity

For example, one author applies the concept of entropy in his architecture to differentiate between relevant and irrelevant data. The method generates a disturbed image from a statistical prototype, where the entropy values are used as quality indicators [8]. Another study uses entropy reduction to mitigate ambiguity and improve the accuracy of model outputs, using a method called Entropy Reduction Aggregation (ERA) [48].

Additionally, one author explores strategies to minimize communication overhead and data heterogeneity, using entropy in client selection in IoT devices with non-iid data [9]. In another approach, information theory is integrated into the algorithm, developing a high-precision classifier based on the optimal combination of features [49].

By evaluating the entropy at the edges, FedAvg-BE aims to reduce execution time in FL environments with non-iid data, achieving a 26% reduction for CIFAR-10 [10]. In contrast to FedAvg-BE, which focuses on minimizing execution time, the algorithm proposed in this study, EnBaSe, employs embedded data filtering to select high-quality data prior to processing. Finally, information theory is applied to the FL context to minimize the dispersion of minority classes and monitor class concentration using specific metrics classes and monitoring the concentration of classes using particular metrics [46].

2.3 Problem Overview and Discussion

The FL model employs decentralized ML, addressing challenges such as connection bottlenecks, infrequent updates, network latency, and convergence delays. These factors significantly impact energy consumption, particularly in low-performance devices such as smartphones, tablets, and industrial equipment.

The lack of validation for new datasets increases the convergence time of trained neural network models and doesn't guarantee that the new data will improve the accuracy. Algorithms designed to validate data quality could, therefore, expedite the convergence of trained models and reduce energy costs for devices operating under such constraints.

The model proposed in this study focuses on validating information prior to initiating the neural network training process. By doing so, only the most relevant data is utilized, reducing the volume of information required for training and directly impacting the costs associated with this process.

Finally, the main difference between EnBaSe and the other more advanced models is that EnBaSe functions as an integrated layer in the architecture of the neural network at the edge. Its function is to quantify the information gain for the neural network and select the most predictable and relevant data.

3 Model Evaluated

The EnBaSe algorithm, described in pseudocode 1, was developed for the embedded system studied in this work and uses entropy as the central measure to quantify the uncertainty of images. Images with lower entropy values indicate lower uncertainty, whereas those with higher values indicate more significant uncertainty.

First, the entropy is calculated for each 2D image, generating key-value pairs that associate each image with its entropy value. These pairs were then sorted and divided into two categories based on the median entropy values, creating image classes with varying levels of uncertainty.

where

i. K: Represents the total number of classes in the dataset.

ii. $X_{\textbf{train}}$: Training Dataset.

iii. $Y_{\textbf{train}}$: Labels corresponding to the training set X_{train}.

iv. $X_{\textbf{selected}}$: Subset X_{train} selected by the algorithm based on entropy.

v. $Y_{\textbf{selected}}$: Labels corresponding to subset X_{selected}.

vi. **label**: A class label ($K - 1$, where K is the total number of classes).

vii. **C**: Set of indices belonging to a given class $label$.

viii. **MEntropy**: An array that stores pairs (index, entropy value) for each image in a given class.

ix. **ComputeEntropy(image)**: A function that calculates the entropy of an image.

x. **Median**: Median entropy values in the set $MEntropy$.

Algorithm 1 EnBaSe. Where K denotes the total number of classes.

```
Require: $\mathcal{X}_{\text{train}}, \mathcal{Y}_{\text{train}}, K$
Ensure: Selected classes based on entropy
$\mathcal{X}_{\text{selected}} \leftarrow \emptyset$
$\mathcal{Y}_{\text{selected}} \leftarrow \emptyset$
for label $\leftarrow 0$ to $K-1$ do
    $\mathcal{C} \leftarrow$ Retrieve indices belonging to class label
    $\mathcal{M}_{\text{Entropy}} \leftarrow \emptyset$
    for each sample $\in \mathcal{C}$ do
        $\mathcal{M}_{\text{Entropy}} \leftarrow (\text{key}, \text{ComputeEntropy}(\text{image}))$
    Sort $\mathcal{M}_{\text{Entropy}}$ by ComputeEntropy(image)
    Calculate the median of $\mathcal{M}_{\text{Entropy}}$
    $\mathcal{I}_{\text{Qualified}} \leftarrow \emptyset$
    for each key $\in \mathcal{M}_{\text{Entropy}}$ do
        if key.entropy $\leq$ median then
            Append key.index to $\mathcal{I}_{\text{Qualified}}$
    for $i$ in $\mathcal{I}_{\text{Qualified}}$ do
        Append $\mathcal{X}_{\text{train}}[i]$ to $\mathcal{X}_{\text{selected}}$
        Append $\mathcal{Y}_{\text{train}}[i]$ to $\mathcal{Y}_{\text{selected}}$
return $\mathcal{X}_{\text{selected}}, \mathcal{Y}_{\text{selected}}$
function COMPUTEENTROPY(image)
    $H \leftarrow -\sum_d p(\text{image}) \log_2(p(\text{image}))$
    return $H$
```

xi. **IQualified**: Set of indices of samples with entropy less than or equal to the median, X_{selected} and Y_{selected}.

xii. **H**: Entropy calculated for the image.

xiii. $p(image)$: Probability distribution associated with the image (used in entropy calculations).

Empty lists are created to store the data based on the input dataset and the corresponding labels. The algorithm then iterates through each class, calculates the entropy value of each image, and stores this data as key-value pairs, where each image is associated with its entropy value.

The image is provided to the function as a matrix containing raw values ranging from 0 to 255 or normalized values between 0 and 1. Images in grayscale and Red, Green, Blue (RGB) channels are treated as a unified probability distribution rather than separate distributions across different color channels. Consequently, we generalize the probabilities associated with color channels—intensity for blue and contrast for green and red—allowing for an analysis of pixel complexity without specific channel distinctions. This abstraction enables the measurement of overall pixel variation using a single numerical matrix.

The values obtained from this analysis provide the visual complexity of the images. According to information theory, when entropy has a low variation, close to 0, there is little uncertainty or the patterns are predictable. On the other hand, when the variation is high, close to 1, it can indicate excessive detail or difficult predictability.

We have observed in experiments that some classes of different data sets have extreme entropy values. Therefore, to avoid the average being influenced by these extreme values when separating low entropy from high entropy, we chose the median as the separation criterion because it minimizes the impact of these extreme values, avoiding distortions in the distribution.

4 Materials and Methods

This section provides a detailed overview of the methodological approaches used in this experiment, enabling other researchers to replicate it. The source code is available on GitHub [1] to support the reproducibility. The following sections describe the materials, methodologies, software, and hardware used in these experiments.

4.1 Dataset Description

To address the challenges of image processing and pattern recognition in heterogeneous scenarios, four image datasets were selected to validate the experimental method: MNIST[2], Fashion-MNIST[3], CIFAR-10 and CIFAR-100[4]. This selection allows the experiment to cover both iid and non-iid scenarios. The dataset characteristics are as follows:

i. **MNIST:** Grayscale images of handwritten digits (0-9), used for digit recognition and classification tasks.

[1] https://github.com/ernesto-arq/Entropy-Artificial-Intelligence.git

[2] https://yann.lecun.com/exdb/mnist/

[3] https://github.com/zalandoresearch/fashion-mnist

[4] https://www.cs.toronto.edu/~kriz/cifar.html

ii. **Fashion-MNIST:** Grayscale images of fashion items in ten categories, such as shirts and pants.

iii. **CIFAR-10:** Color images are divided into ten categories, including vehicles and animals.

iv. **CIFAR-100:** Diverse color images encompassing 100 classes providing greater class granularity.

Table 3: Properties of Datasets.

Property	**Fashion-MNIST**	**MNIST**	**CIFAR-10**	**CIFAR-100**
Number of Classes	10 Classes	10 Classes	10 Classes	100 Classes
Image Dimension	28x28	28x28	32x32	32x32
Color Type	Grayscale	Grayscale	RGB	RGB
Total Images/Traning	60,000	60,000	50,000	50,000
Images by Class	6,000	6,000	6,000	600

4.2 Non-IID Experiment Configuration

To simulate a (FL) scenario with non-iid data, the techniques of **feature distribution**, **label distribution skew**, and **quantity skew** have been used [50–53]. These aspects, as described below, contribute to the creation of a heterogeneous FL environment, allowing for diversity in contributions and characteristics:

i. **Feature Distribution Skew:** Variation in data features across clients with the same labels.

ii. **Label Distribution Skew:** Unequal label distributions across clients, creating distinct class groups.

iii. **Quantity Skew:** Imbalances in data volume per client, affecting local model training.

Each dataset underwent 60 experiments combining Normal Execution (Normal), EnBaSe, and Random with FedAvg [54] and FedProx [55], totaling 240 experiments. Data will be distributed across ten virtually simulated devices, which will perform training and send updates for aggregation. The following describes aggregation algorithms for federated environments, representing part of the evaluation methodology:

i. **FedAvg:** Weighted average of client updates, adjusted by data volume.

ii. **FedProx:** Regularizes local losses to minimize deviations from the global model.

Experiments were conducted using 10 clients and 50 epochs to evaluate the algorithm's effectiveness under constrained conditions. FedAvg and FedProx were selected as classical models from the literature to serve as baseline comparisons, providing a more comprehensive understanding of the results. The number of devices and epochs was defined according to the articles published and compared in Table 7. Aiming at IoT applications with low processing power and energy limitations, we limited the number of epochs and the number of devices available for the first experiment. In the second experiment, to test resilience, we increased the number of epochs to 100 and the number of devices to 50. The choice of hyperparameters and architecture was empirical, with simulations carried out until the neural network demonstrated stability and a balance between accuracy and loss.

4.3 Deep Neural Networks

For the non-iid experiment with heterogeneous data, specific normalization parameters were applied. The MNIST and Fashion-MNIST datasets used Convolutional Neural Network (CNN) without Transfer Learning (TL) or Data Augmentation (DA), optimized with SGD. For CIFAR-10 and CIFAR-100, an adapted ResNet-50 model was employed, incorporating DA, and also optimized with SGD. The following normalization parameters[5] were applied:

MNIST and Fashion-MNIST: CNN models with batch size 128, **SGD** optimizer, 50 epochs, without TL or DA. Standard normalization parameters were applied for each dataset.

CIFAR-10 and CIFAR-100: Adapted ResNet-50 model with batch size 128, DA, **SGD** optimizer, and 50 epochs. Normalization parameters for RGB channels were applied for each dataset. The DA techniques used include horizontal inversion, rotation with a limit of up to 15 degrees, and a random affine transformation (0, (0.1; 0.1)). The values were tested empirically, observing better network performance in the face of input variations. Finally, ResNet-50 was selected because its characteristics are more complex than previous models and less computationally demanding than larger models [56].

The neural network architecture configurations are presented in Table 4, with the following components: batch normalization (BN), max pooling (MP), dropout (DP), fully-connected layer (FC), and global average pooling (GAP). The architecture consists of the following layers:

- **Layer 1:** three blocks, each with Conv, BN, ReLU, and MP components;
- **Layer 2:** four blocks, each with Conv, BN, ReLU, and MP components;
- **Layer 3:** six blocks, each with Conv, BN, ReLU, and MP components;
- **Layer 4:** three blocks, each with Conv, BN, ReLU, and MP components.

[5] https://github.com/jeremy313

Table 4: Model Configurations non-iid.

MNIST	Fashion	CIFAR-10	CIFAR-100
CONV-1	CONV-1	ResNet50	ResNet50
MP	MP	BN	BN
CONV-2	CONV-2	LAYER-1	LAYER-1
MP	MP	LAYER-2	LAYER-2
FC-500	FC-500	LAYER-3	LAYER-3
FC-10	FC-10	LAYER-4	LAYER-4
SOFTMAX	SOFTMAX	GAP	GAP
		FC-512	FC-512
		BN	BN
		DP-1	DP-1
		FC-10	FC-100
		SOFTMAX	SOFTMAX

4.4 Performance Evaluation Criteria

The following metrics were used to comprehensively evaluate the performance of the models in this experiment, ensuring a thorough analysis of the results:

i. **Accuracy:** The proportion of correct predictions out of the total samples.

ii. **Recall:** The proportion of true positives correctly identified among all actual positive cases;

iii. **F1-Score:** The harmonic mean of precision and recall used as an additional metric in the FL context;

iv. **Learning Curve:** A measure model of performance over time by comparing training and validation curves to detect overfitting and underfitting;

v. **Precision:** The proportion of true positives out of all positive predictions, which measures the accuracy of positive predictions;

vi. **Boxplot:** Statistical visualization used to analyze data distribution in relation to the median, quartiles, and outliers;

vii. **Variance:** A measure of data variability relative to the mean;

viii. **Standard Deviation:** A measure of data dispersion around the mean, allowing for the analysis of variability;

ix. **Minimum, Mean, and Maximum Accuracy:** Evaluation of model accuracy across different phases, covering the minimum, average, and maximum performance values;

4.5 Performance Evaluation Criteria

Pseudocode 2 illustrates the Random technique used in this work. This technique uses the scikit-learn library [57] to manage the organization and selection of stratified samples by class. For consistency, this selection methodology is referred to as Random throughout the text.

Algorithm 2 Procedure for Splitting the Dataset into Training and Test Subsets.

```
Require: Feature set $X$, labels $y$, test set proportion $test\_size$, shuffling parameter $shuffle = True$, stratification parameter $stratify = None$
Ensure: Training and test subsets: $X_{train}, X_{test}, y_{train}, y_{test}$
Calculate the test set size based on the value of $test\_size$
if $shuffle = True$ then
    Perform random shuffling of the data to ensure random distribution
if $stratify \neq None$ then
    Apply stratified splitting of the data, preserving the class distribution indicated by $stratify$
Split the dataset $X$ and labels $y$ into training and test subsets, respecting the specified test set size
Return the training and test subsets: $X_{train}, X_{test}, y_{train}, y_{test}$
```

As presented in the background Section 2, this algorithm will be used to analyze the EnBaSe results comparatively, evaluating performance and efficiency alongside the global aggregation techniques discussed in Subsection 2.1. Furthermore, in the final results section, the EnBaSe results will be compared with the state-of-the-art.

5 Results

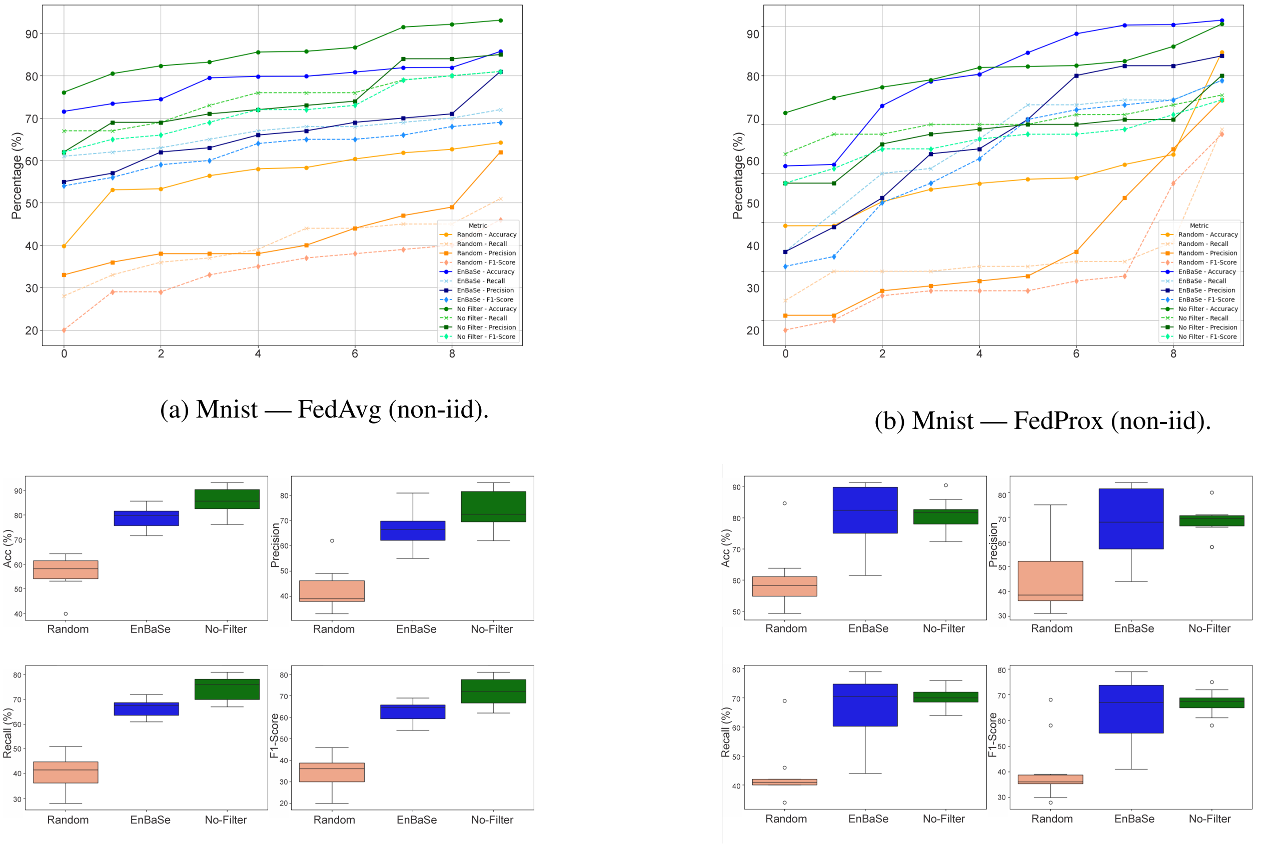
This section presents the results of the embedded EnBaSe algorithm (Section 3), evaluated in an FL environment with FedAvg and FedProx (Subsection 4.2). The skewed data distribution (Subsection 4.2) creates a non-iid scenario, as there are no guarantees about the number of classes or the volume of data on each device. We evaluate performance in three scenarios: (i) Normal-Execution (complete set), (ii) EnBaSe (selects half the set based on entropy), and (iii) Random (random samples), as per Subsections 2.1. The aim is to compare the computing time and effectiveness of EnBaSe, which only processes half of the low entropy data. At the same time, Random uses the same number of samples but is chosen randomly.

In addition, experiments were carried out with EnBaSe on the MNIST, Fashion-MNIST, and CIFAR-100 sets (Subsection 4.1), and the results obtained were compared with state-of-the-art models (Table 7). The following subsections present the results for MNIST (Subsection 5.1), Fashion-MNIST (5.2), CIFAR-10 (5.3), and CIFAR-100 (5.4), respectively.

5.1 MNIST results

A total of 60 experiments were conducted on the MNIST dataset. Of these, 30 experiments were dedicated to training using the complete dataset (Normal), distributed according to the distribution skew, applied to both the EnBaSe and Random, as well as the global aggregation algorithm FedAvg for FL. The additional 30 experiments were consistently conducted using the global aggregation algorithm FedProx. Table 3 provides a detailed description of the MNIST dataset, including its primary specifications. Figures 1 a) and b) represent the results for the MNIST dataset using the FedAvg and FedProx FL aggregation algorithms, respectively.

The EnBaSe algorithm demonstrates adaptability to the heterogeneous (non-iid) scenario. By contrast, the Random exhibited a notable drop in precision across these metrics under the same conditions. Thus, the random selection may be suboptimal in a heterogeneous scenario where data points are interdependent and do not follow the same distribution.



(a) Mnist — FedAvg (non-iid).

(b) Mnist — FedProx (non-iid).

(c) Box Plot Mnist — FedAvg (non-iid).

(d) Box Plot Mnist — FedProx (non-iid).

Figure 1: Results of FedAvg and FedProx on the MNIST dataset.

Thus, the random selection algorithm may be suboptimal in a heterogeneous scenario where data points are interdependent and do not follow the same distribution. This sampling approach may fail to guarantee the representativeness across each node (i.e., device) in various existing distributions. Additionally, when comparing the FedAvg and FedProx techniques, it was observed that FedProx demonstrated greater robustness based on the metric values obtained for this evaluation dataset (i.e., MNIST).

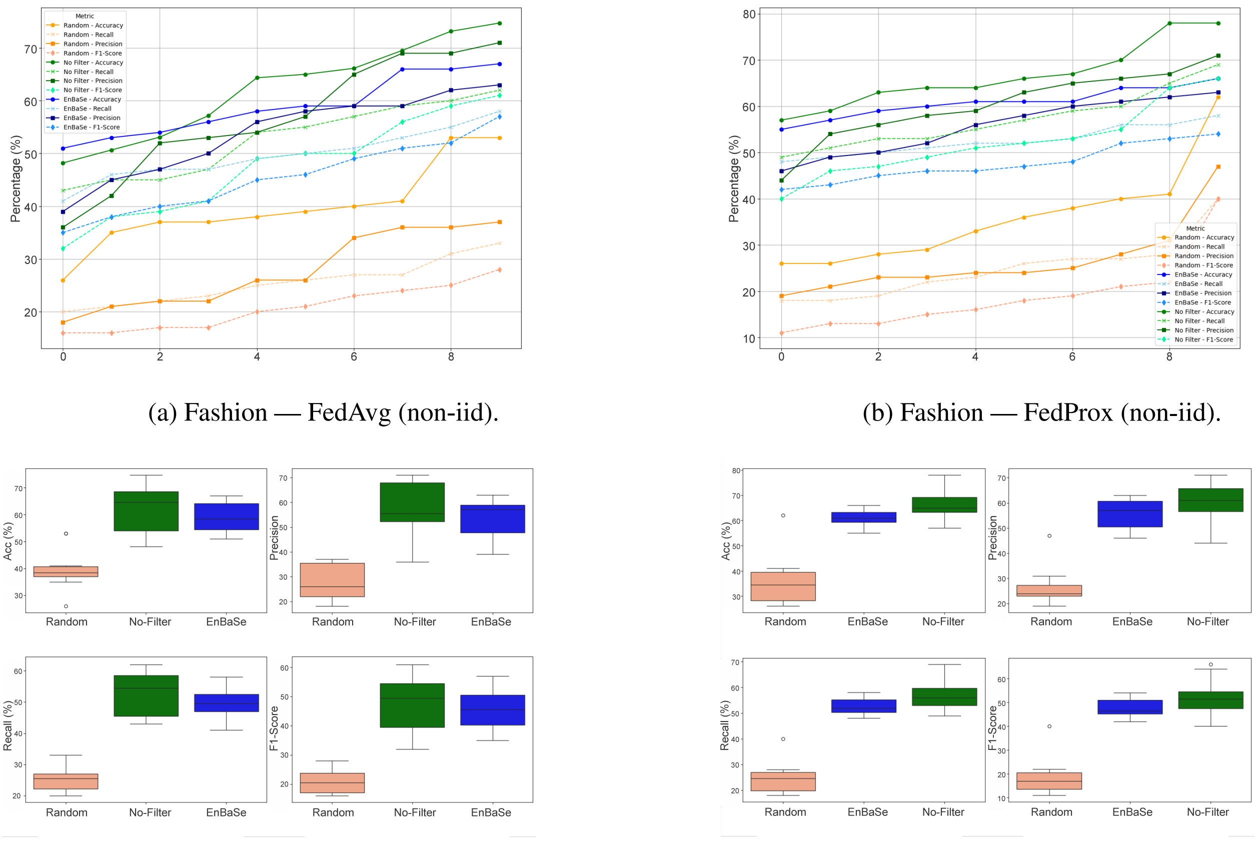
EnBaSe performs training with half of the data while applying quantitative analysis to ensure data quality and demonstrates adaptability to the heterogeneous (non-iid) scenario. In contrast, the Random exhibited a notable drop in precision across these metrics under the same conditions. In Figure 1 c), significant differences can be observed in the accuracy, precision, recall, and F1-score metrics for FedAvg, with the results of the Random diverging more from those of other techniques, as previously noted. In contrast, the results obtained by EnBaSe and Normal were closer.

Regarding the results presented in Figure 1 d), a pattern similar to the previous observation can be seen. For the FedProx algorithm, there was more significant variability in the results, which were distributed almost symmetrically around the median in the boxplot. This variation is noteworthy from the algorithm's perspective because its performance is comparable to that achieved with training on a complete dataset. When training was performed with full data, the results exhibited a more stable distribution with less variation. In contrast, the Random results showed high variability, with a dense concentration of low values and outliers.

5.2 Fashion-MNIST results

For the Fashion-MNIST dataset of grayscale fashion and clothing images, 60 experiments were conducted, divided equally between the FedAvg and FedProx global aggregation algorithms, considering the Normal, EnBaSe and Random configurations. Table 3 gives an overview of the Fashion-MNIST dataset.

In Figure 2 a), for Fashion-MNIST using the FedAvg FL algorithm, the training algorithms with the entire dataset and EnBaSe demonstrate comparable performance in metrics and outcomes. This allowed us to observe the minimum and maximum performances of the model across the experiments. Additionally, it is evident that EnBaSe experiences minimal loss relative to the Normal, proportional to its accuracy and other metrics.



(a) Fashion — FedAvg (non-iid).

(b) Fashion — FedProx (non-iid).

(c) Box Plot Fashion — FedAvg (non-iid).

(d) Box Plot Fashion — FedProx (non-iid).

Figure 2: Results of FedAvg and FedProx on the Fashion-MNIST dataset.

In contrast, if we randomly select data from each class without applying a quality criterion in a non-iid environment, we observe a decrease in performance, with results inferior to those obtained with both Normal and EnBaSe.

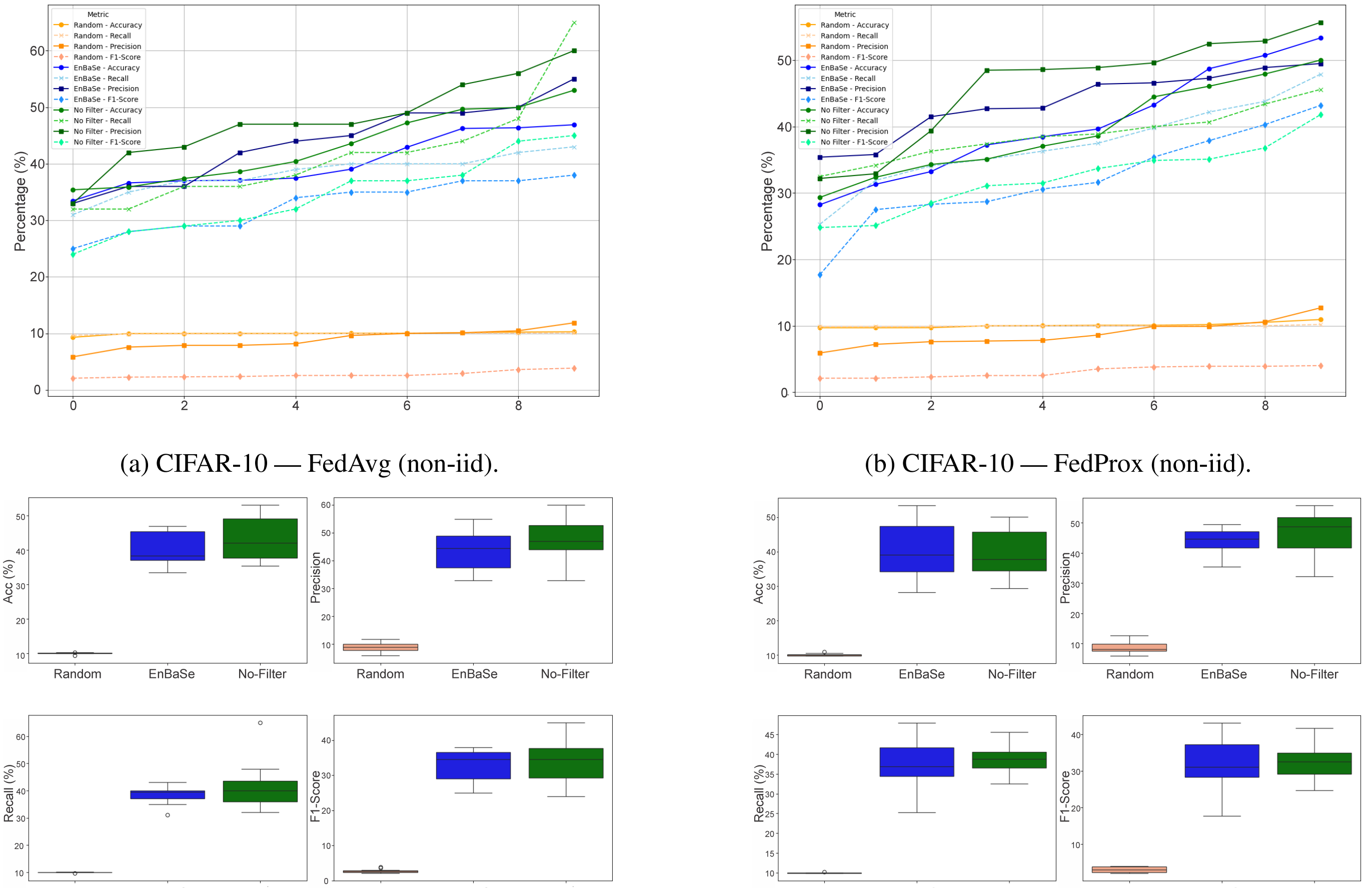
Similarly, in Figure 2 b), which represents FedProx, consistency is observed across different metrics in various simulations. In both cases, the Random yielded the lowest results. Thus, it can be concluded that the overall accuracy remains consistent throughout the experiments, with an emphasis on training with the full dataset (Normal) and EnBaSe, which presented similar results and less pronounced variability.

In the boxplot results shown in Figure 2 c) for FedAvg, the normal execution demonstrated a slight superiority in results, albeit with more significant variability across different training sessions. The EnBaSe algorithm, on the other hand, shows lower variability in results. For the Random, a flattened distribution with low accuracy, dispersed values, and outliers can be observed, highlighting the limitations of this technique in the non-iid scenario.

The same pattern is evident in Figure 2 d) for the FedProx. Thus, it can be concluded that the normal execution maintains a consistently superior performance for both FedAvg and FedProx. EnBaSe also achieved favorable results, whereas the Random technique showed the poorest performance across all metrics.

5.3 CIFAR-10 results

The CIFAR-10 dataset includes ten distinct classes (airplanes, cars, birds, cats, deer, dogs, etc.), each with low resolution and limited visual information, which presents a significant challenge in terms of visual complexity. Table 3 presents the main characteristics of the dataset.



(a) CIFAR-10 — FedAvg (non-iid).

(b) CIFAR-10 — FedProx (non-iid).

(c) Box Plot CIFAR-10 — FedAvg (non-iid).

(d) Box Plot CIFAR-10 — FedProx (non-iid).

Figure 3: Results of FedAvg and FedProx on the CIFAR-10 dataset.

This visual complexity is essential for analyzing the results shown in Figure 3 a), which illustrate the performance of FedAvg. It can be observed that the CIFAR-10 results follow a consistent pattern with the experiments conducted on MNIST and Fashion-MNIST, with a slight improvement in the results for FedAvg in the case of CIFAR-10. This pattern was observed across multiple experiments using different algorithms, neural networks, and hyperparameters.

The pattern observed in Figure 3 a) is similar to that in Figure 3 b) for FedProx. These observations support the hypothesis that, in non-iid scenarios, a quantitative approach to data quality evaluation can achieve results comparable to those obtained with large data volumes.

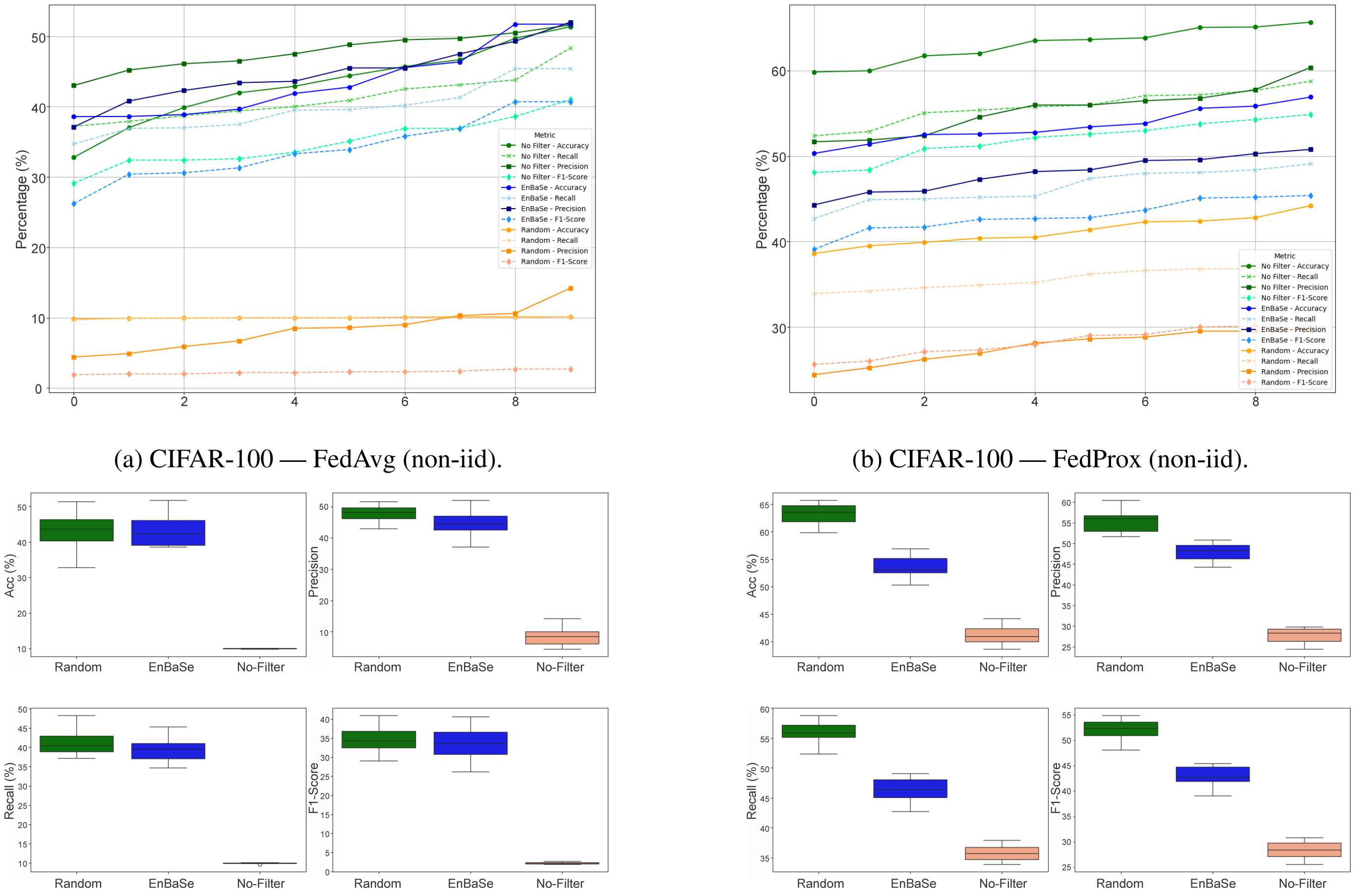
The behavior of EnBaSe demonstrates the feasibility of converging the model and achieving results comparable to the typical execution approach by selecting only half of the CIFAR-10 dataset using a quantitative selection strategy. This pattern, observed with the FedAvg and FedProx algorithms, suggests that the processing time and computational costs can be significantly reduced in low-power IoT environments with minimal performance loss.

Figures 3 c) and 3 d) present boxplots for CIFAR-10 using the FedAvg and FedProx global aggregation models, respectively. A similar and consistent variation is observed with other experiments for the normal and EnBaSe techniques, whereas the Random experiments encountered more significant challenges in handling the classification task.

Additionally, the Random showed outliers outside the central distribution, indicating instances where the model occasionally achieved a performance above the average. However, these results were inferior to those of Normal and EnBaSe. A slightly higher variability is also noticeable for Normal compared with EnBaSe, suggesting that in non-iid scenarios, particularly with CIFAR-10, the EnBaSe demonstrates greater robustness.

5.4 CIFAR-100 results

The following sections present the results obtained for CIFAR-100, which encompasses 100 distinct classes (fish, flowers, vehicles, animals, etc.), representing a challenging benchmark in the field of Computer Vision (CV). This dataset requires neural network models to handle significant complexities owing to the various classes, as summarized in Table 3.



(a) CIFAR-100 — FedAvg (non-iid).

(b) CIFAR-100 — FedProx (non-iid).

(c) Box Plot CIFAR-100 — FedAvg (non-iid).

(d) Box Plot CIFAR-100 — FedProx (non-iid).

Figure 4: Results of FedAvg and FedProx on the CIFAR-100 dataset.

Overall, the best results were obtained with FedAvg, as shown in Figure 4 a), evaluated across various metrics for this dataset. The EnBaSe algorithm selected half of the data, achieving results that were competitive with the Normal approach. Furthermore, by selecting only the most representative and high-quality samples for the dataset, EnBaSe reduces the training time of the neural network by half.

Figure 4 b) presents the results for FedProx on CIFAR-100, replicating the pattern observed in other experiments. A performance level proportional to each method's capacity was also noted for EnBaSe, while FedProx demonstrated a lower performance than FedAvg.

Figure 4 c) presents the best results among global aggregation algorithms, which presents the FedAvg boxplot. There is notable consistency in the accuracy, precision, recall, and F1-score metrics, reflecting the learning plateau achieved by the model.

In the results obtained for the CIFAR-100 dataset using the global aggregation algorithm FedAvg, both Normal and EnBaSe presented consistent and high results, with medians close to each other. In contrast, the Random exhibited a significantly lower performance range.

Regarding precision, both Normal and EnBaSe maintain high consistency, whereas Random consistently yields lower results. This same pattern is observed in the metrics of Recall and F1-Score, indicating that EnBaSe performs an adequate selection of relevant samples compared to Normal. Similar results, with minor differences in the values, can be observed in Figure 4 d), which represents the FedProx boxplot.

5.5 Performance Variability and Dispersion in Federated Learning

Table 5 presents the minimum, mean, and maximum accuracy values obtained by the global model, along with the respective execution times, which indicate computational and energy costs.

Generally, the Random technique exhibits computational costs similar to EnBaSe, but with marked differences in accuracy. In this setting, EnBaSe efficiently balances cost and accuracy across datasets of varying complexity, highlighting its robustness and consistency.

In the MNIST and Fashion-MNIST datasets, Normal and EnBaSe achieve similar accuracy results, despite both EnBaSe and Random using only half of the dataset. However, Random shows a noticeable drop in accuracy, indicating EnBaSe's effectiveness in handling non-iid data.

In more challenging datasets (CIFAR-10 and CIFAR-100), EnBaSe again demonstrates performance close to normal execution, suggesting that not all data samples contribute to model accuracy, and some may even introduce noise. These findings confirm EnBaSe's ability to address data heterogeneity while maintaining an efficient trade-off between performance and computational cost.

Table 5: Performance Analysis between Datasets and non-iid Techniques

Dataset	Technique	Algorithm	Accuracy (%)			Time (s)		
			Min	Avg	Max	Min	Avg	Max
Mnist	FedAvg	Normal	76.10	85.70	93.11	1,074	1,096	1,074
Mnist	**FedAvg**	**EnBaSe**	**71.58**	**78.91**	**85.72**	**577**	**586**	**577**
Mnist	FedAvg	Random	39.79	56.80	64.22	549	555	565
Mnist	FedProx	Normal	73.84	81.73	91.30	1,163	1,180	1,163
Mnist	**FedProx**	**EnBaSe**	**61.52**	**81.26**	**90.51**	**620**	**629**	**620**
Mnist	FedProx	Random	49.28	59.13	80.26	595	606	617
Fashion	FedAvg	Normal	48.20	62.21	74.73	1,046	1,084	1,103
Fashion	**FedAvg**	**EnBaSe**	**51.00**	**58.80**	**67.00**	**547**	**556**	**571**
Fashion	FedAvg	Random	26.00	39.80	53.00	568	574	589
Fashion	FedProx	Normal	57.00	66.40	78.00	1,108	1,125	1,157
Fashion	**FedProx**	**EnBaSe**	**55.00**	**60.80**	**66.00**	**597**	**601**	**609**
Fashion	FedProx	Random	26.00	35.90	62.00	628	632	637
CIFAR-10	FedAvg	Normal	28.82	43.12	53.00	14,676	14,941	15,229
CIFAR-10	**FedAvg**	**EnBaSe**	**33.47**	**40.32**	**46.90**	**8,107**	**8,259**	**8,412**
CIFAR-10	FedAvg	Random	9.33	10.01	10.30	7,933	8,205	8,249
CIFAR-10	FedProx	Normal	29.34	39.52	50.02	15,368	15,662	15,941
CIFAR-10	**FedProx**	**EnBaSe**	**28.24**	**40.42**	**53.36**	**8,276**	**8,413**	**8,564**
CIFAR-10	FedProx	Random	9.70	10.09	10.94	8,329	8,394	8,528
CIFAR-100	FedAvg	Normal	32.79	43.24	51.35	14,936.	15,214	15,279
CIFAR-100	**FedAvg**	**EnBaSe**	**32.74**	**41.60**	**51.72**	**8,123**	**8,131**	**8,140**
CIFAR-100	FedAvg	Random	09.81	10.00	10.12	7,916	8,274	8,339
CIFAR-100	FedProx	Normal	59.88	63.08	65.71	17,649	18,020	18,446
CIFAR-100	**FedProx**	**EnBaSe**	**50.33**	**53.54**	**56.93**	**8,150**	**8,202**	**8,243**
CIFAR-100	FedProx	Random	38.60	41.20	44.20	8,109	8,418	8,648

Table 6 presents the variance and standard deviation results, allowing a comparison of outcome variability. FedAvg generally shows lower variance and dispersion, indicating higher stability as a global aggregation strategy.

For FedProx, there was higher variability, especially in the case of EnBaSe. This suggests that entropy may introduce greater variability in FedProx, which aims to minimize deviations in the model weights. This observation raises hypotheses about the impact of entropy on FedProx variability and convergence.

The first hypothesis is that while FedProx is designed to penalize large deviations from global model weights, introducing entropy increases distributional complexity, making weight adjustments more challenging. In highly heterogeneous data, this penalization intensifies, complicating weight stabilization.

The second hypothesis attributes FedProx's variability to frequent device updates before aggregation, amplifying the results' variability. Moreover, EnBaSe's entropy approximates the variance of normal execution, suggesting adjustments to enhance stability.

5.6 Benchmarking EnBaSe: Comparison with Recent Models

Table 7 shows benchmark results of EnBaSe compared to recent studies, focusing on its impact on accuracy. Benchmark results for CIFAR-10 and CIFAR-100 demonstrate FedProx's performance with EnBaSe over 100 epochs and 50 clients.

In CIFAR-10 the FedCOME model achieved 75.88% accuracy, while AdaFedAdam achieved 72.77%. FedPer++ and FedAvg (Adapted) performed 85.09% and 90.80%, respectively. EnBaSe combined with FedProx achieved 84.46%, placing it close to

Table 6: Performance Metrics of Different Algorithms Across Datasets

Dataset	Technique	Analysis	Algorithm	Acc (%)	Precision	Recall (%)	F1-Score
Mnist	FedAvg	Variance	**EnBaSe**	**19.36**	**55.88**	**13.17**	**25.82**
	FedAvg		Normal	29.73	58.68	27.16	43.21
	FedAvg		Random	49.60	71.61	46.84	52.71
	FedAvg	Std. Dev.	**EnBaSe**	**4.40**	**7.48**	**3.63**	**5.08**
	FedAvg		Normal	5.45	7.66	5.21	6.57
	FedAvg		Random	7.04	8.46	6.84	7.26
	FedProx	Variance	**EnBaSe**	**127.54**	**214.48**	**133**.43	**186.1**
	FedProx		Normal	26.81	41.66	11.38	24.40
	FedProx		Random	100.90	224.77	88.94	158.71
	FedProx	Std. Dev.	**EnBaSe**	**11.29**	**14.65**	**11.55**	**13.64**
	FedProx		Normal	5.18	6.45	3.37	4.94
	FedProx		Random	10.04	14.99	9.43	12.60
Fashion	FedAvg	Variance	**EnBaSe**	**32.99**	**0.01**	**0.00**	**48.27**
	FedAvg		Normal	88.49	0.01	0.01	94.06
	FedAvg		Random	64.77	0.01	0.00	17.79
	FedAvg	Std. Dev.	**EnBaSe**	**5.74**	**0.08**	**0.05**	**6.95**
	FedAvg		Normal	9.41	0.12	0.07	9.70
	FedAvg		Random	8.05	0.07	0.04	4.22
	FedProx	Variance	**EnBaSe**	**11.07**	**36.68**	**10.72**	**17.16**
	FedProx		Normal	49.82	61.34	39.66	62.68
	FedProx		Random	115.88	63.17	43.29	68.40
	FedProx	Std. Dev.	**EnBaSe**	**3.33**	**6.06**	**3.27**	**4.14**
	FedProx		Normal	7.06	7.83	6.30	7.92
	FedProx		Random	10.76	7.95	6.58	8.27
CIFAR-10	FedAvg	Variance	**EnBaSe**	**23.72**	**51.21**	**12.49**	**20.68**
	FedAvg		Normal	41.81	59.29	94.94	48.27
	FedAvg		Random	0.07	3.12	0.02	0.34
	FedAvg	Std. Dev.	**EnBaSe**	**4.87**	**7.16**	**3.53**	**4.55**
	FedAvg		Normal	6.47	7.70	9.74	6.95
	FedAvg		Random	0.26	1.77	0.13	0.59
	FedProx	Variance	**EnBaSe**	**72.06**	**25.25**	**41.56**	**54.67**
	FedProx		Normal	50.80	69.31	15.73	27.96
	FedProx		Random	0.15	3.95	0.01	0.68
	FedProx	Std. Dev.	**EnBaSe**	**8.49**	**5.03**	**6.45**	**7.39**
	FedProx		Normal	7.13	8.33	3.97	5.29
	FedProx		Random	0.39	1.99	0.09	0.82
CIFAR-100	FedAvg	Variance	**EnBaSe**	**26.14**	**18.36**	**12.53**	**21.62**
	FedAvg		Normal	31.98	7.08	11.15	12.48
	FedAvg		Random	0.01	8.87	0.01	0.08
	FedAvg	Std. Dev.	**EnBaSe**	**5.11**	**4.28**	**3.54**	**4.65**
	FedAvg		Normal	5.66	2.66	3.34	3.53
	FedAvg		Random	0.10	2.98	0.11	0.28
	FedProx	Variance	**EnBaSe**	**4.28**	**4.62**	**4.25**	**3.85**
	FedProx		Normal	4.32	7.80	4.10	5.37
	FedProx		Random	2.97	3.68	1.77	3.19
	FedProx	Std. Dev.	**EnBaSe**	**2.07**	**2.15**	**2.06**	**1.96**
	FedProx		Normal	2.08	2.79	2.02	2.32
	FedProx		Random	1.72	1.92	1.33	1.79

Table 7: Benchmark of Models on CIFAR-10 and CIFAR-100 Datasets.

Dataset	Architecture	Author	Model	Acc (%)
CIFAR-10	ConvNet	[58]	FedCOME	75.88
	VGG11	[17]	AdaFedAdam	72.77
	ResNet-50	**Our Model**	**FedProx (EnBaSe)**	**84.46**
	CNN	[59]	FedPer++	85.09
	ResNet-50	[60]	FedAvg (Adapted)	90.80
CIFAR-100	ConvNet	[58]	FedCOME	37.66
	ResNet-56	[46]	Fed-IT	39.29
	CCT-2	[61]	FedAvg-Vanilla	40.36
	ResNet-18	[62]	FedProx(FedFed)	70.02
	ResNet-50	**Our Model**	**FedProx (EnBaSe)**	**72.84**

FedPer++. Although EnBaSe did not surpass FedAvg (Adapted) in accuracy, it stood out for its computational efficiency by reducing processing costs by selecting only half of the data based on entropy without significant losses in accuracy.

In CIFAR-100, the FedCOME, Fed-IT, and FedAvg-Vanilla models showed accuracies of 37.66%, 39.29%, and 40.36%, respectively. FedProx (FedFed) achieved 70.02%, while EnBaSe, combined with FedProx, achieved the best performance, with 72.84%. Given the high number of classes in CIFAR-100, EnBaSe proved capable of selecting relevant samples while maintaining high accuracy with a reduced volume of data.

6 Conclusion

In this study, we analyzed the results, dispersion, and variability of the EnBaSe algorithm. This entropy-based algorithm works as an additional neural network layer, quantifying the network's information gain and selecting the most informative samples.

The overall performance shows the ability to select more representative samples in heterogeneous environments and halve computational costs and, indirectly, energy costs, reinforcing its potential and application in scenarios where computational and energy limitations are critical factors.

Despite the contributions, some limitations were observed, particularly in specific scenarios with variability in the results, in some cases showing more excellent dispersion and variance.

Additionally, EnBaSe demonstrated stability in a scenario with non-iid data, showing limited variation between the results, indicating symmetry in the distributions and creating expected ranges of these results. These observations suggest that the data selection technique, addressing entropy, improves reliability, allowing a scenario with heterogeneous data to have a constant range in the expectation of the results obtained, as shown in Table (5).

Finally, the algorithm showed efficiency when scaled from 10 to 50 devices. Despite the increased number of devices, the model maintained its efficiency, improving accuracy without compromising performance.

In non-iid scenarios, which simulate real-world conditions without ensuring data quantity, quality, or balance, entropy emerges as an effective metric to preserve learning quality, even with reduced data quantities. Thus, EnBaSe efficiently uses computational resources and processing time.

Compared to other state-of-the-art works, EnBaSe obtained results close to the highest accuracy in the state-of-the-art field, halving the computational cost with minimal loss. In this way, the results confirm the application of EnBaSe in federated learning scenarios, highlighting its robustness in efficiently using computational resources, with applications in environments with computational and energy limitations, and dealing with data heterogeneity.

In the comparative results obtained in the state-of-the-art, EnBaSe showed greater accuracy for CIFAR-100, a dataset considered a benchmark in the field of computer vision, when combined with FedProx, surpassing models such as FedCOME, Fed-IT, and FedAvg-Vanilla, which achieved accuracies of 37.66%, 39.29%, and 40.36%, respectively. These results highlight EnBaSe's ability to select relevant and representative samples in highly heterogeneous scenarios with many classes.

Despite the promising results, entropy-based selection showed more significant variability in scenarios with extreme heterogeneity. This limitation can be exploited by combining a hybrid selection approach with adaptive methods to improve the model's stability and reliability while maintaining computational efficiency.

Finally, the results show that EnBaSe reduced the convergence time by around 50%, resulting in lower computational and energy costs. In low-performance experiments (10 devices), the model proved stable, with consistent results and a constant range of variability. When increasing the number of available clients (40 devices), the algorithm maintained its efficiency and demonstrated scalability, showing a constant increase in accuracy.

Future works include validating the refinement hypotheses for the Random and EnBaSe algorithms in non-iid scenarios and comparing them with other global aggregation algorithms for FL. Additionally, exploring a hybrid approach that combines random sampling and entropy could be valuable, as well as using entropy to measure the uncertainty or impurity of the data. In addition, these studies also show the feasibility of creating a kernel to analyze image subsets and compute their entropy levels to segment regions of interest with complex details. Finally, in future studies, we intend to include TL and DA in the MNIST and Fashion-MNIST datasets to analyze more robust results.

Acknowledgements

This study was partly financed by the Coordination for the Improvement of Higher Education Personnel - Brazil (CAPES) - Finance Code 001 and partly by the National Institute of Science and Technology (INCT-Signals) sponsored by Brazil's National Council for Scientific and Technological Development (CNPq) under grant no. 406517/2022-3. Partial support was provided via the CEREIA Project (# 2020/09706-7) São Paulo Research Foundation (FAPESP) and FAPESP–MCTIC-CGI. BR, in partnership with the Hapvida NotreDame Intermedica group.

References

[1] L. S. Vailshery. "Number of IoT connected devices worldwide 2019-2023, with forecasts to 2030". https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/, Jul 2023. Accessed: 12/08/2024.

[2] M.-T. Huynh, M. Nippa and T. Aichner. "Big data analytics capabilities: Patchwork or progress? A systematic review of the status quo and implications for future research". *Technological Forecasting and Social Change*, vol. 197, pp. 122884, 2023. DOI:https://doi.org/10.1016/j.techfore.2023.122884.

[3] A. R. Munappy, J. Bosch, H. H. Olsson, A. Arpteg and B. Brinne. "Data management for production quality deep learning models: Challenges and solutions". *Journal of Systems and Software*, vol. 191, pp. 111359, 2022. DOI:https://doi.org/10.1016/j.jss.2022.111359.

[4] D. van Tilborg, H. Brinkmann, E. Criscuolo, L. Rossen, R. Özçelik and F. Grisoni. "Deep learning for low-data drug discovery: hurdles and opportunities". *Current Opinion in Structural Biology*, vol. 86, pp. 102818, 2024. DOI:https://doi.org/10.1016/j.sbi.2024.102818.

[5] J. C. D. Anjos, K. J. Matteussi, F. C. Orlandi, J. L. Barbosa, J. S. Silva, L. F. Bittencourt and C. F. Geyer. "A survey on collaborative learning for intelligent autonomous systems". *ACM Computing Surveys*, vol. 56, no. 4, pp. 1–37, 2023. DOI:https://doi.org/10.1145/3625544.

[6] M. M. Bassiouni, R. K. Chakrabortty, K. M. Sallam and O. K. Hussain. "Deep learning approaches to identify order status in a complex supply chain". *Expert Systems with Applications*, vol. 250, pp. 123947, 2024. DOI:https://.

[7] J. Wen, Z. Zhang, Y. Lan, Z. Cui, J. Cai and W. Zhang. "A survey on federated learning: challenges and applications". *International Journal of Machine Learning and Cybernetics*, vol. 14, no. 2, pp. 513–535, 2023. DOI:https://doi.org/10.1007/s13042-022-01647-y.

[8] Y. Li, X. Chao and S. Ercisli. "Disturbed-entropy: a simple data quality assessment approach. ICT Express. 2022". DOI:https://doi.org/10.1016/j.icte.2022.01.006.

[9] R. W. Condori Bustincio, A. M. de Souza, J. B. Da Costa and L. Bittencourt. "EntropicFL: Efficient Federated Learning via Data Entropy and Model Divergence". In *Proceedings of the IEEE/ACM 16th International Conference on Utility and Cloud Computing*, pp. 1–6, 2023. DOI:doi.org/10.1145/3603166.3632611.

[10] F. C. Orlandi, J. C. S. Dos Anjos, J. F. d. P. Santana, V. R. Q. Leithardt and C. F. R. Geyer. "Entropy to mitigate non-IID data problem on Federated Learning for the Edge Intelligence environment". *IEEE Access*, vol. 11, pp. 78845–78857, July 2023. DOI:https://10.1109/ACCESS.2023.3298704.

[11] P. Sun, S. Shen, Y. Wan, Z. Wu, Z. Fang and X.-z. Gao. "A survey of iot privacy security: Architecture, technology, challenges, and trends". *IEEE Internet of Things Journal*, 2024. DOI:10.1109/JIOT.2024.3372518.

[12] Y. Y. F. Panduman, N. Funabiki, E. D. Fajrianti, S. Fang and S. Sukaridhoto. "A Survey of AI Techniques in IoT Applications with Use Case Investigations in the Smart Environmental Monitoring and Analytics in Real-Time IoT Platform". *Information*, vol. 15, no. 3, pp. 153, 2024. DOI:https://doi.org/10.3390/info15030153.

[13] J. Bian, A. Al Arafat, H. Xiong, J. Li, L. Li, H. Chen, J. Wang, D. Dou and Z. Guo. "Machine learning in real-time Internet of Things (IoT) systems: A survey". *IEEE Internet of Things Journal*, vol. 9, no. 11, pp. 8364–8386, 2022. DOI:https://10.1109/JIOT.2022.3161050.

[14] T. Mazhar, H. M. Irfan, I. Haq, I. Ullah, M. Ashraf, T. A. Shloul, Y. Y. Ghadi, Imran and D. H. Elkamchouchi. "Analysis of challenges and solutions of IoT in smart grids using AI and machine learning techniques: A review". *Electronics*, vol. 12, no. 1, pp. 242, 2023. DOI:https://doi.org/10.3390/electronics12010242.

[15] S. Balakrishnan, K. Suresh Kumar, L. Ramanathan and S. Muthusundar. "IoT for health monitoring system based on machine learning algorithm". *Wireless Personal Communications*, vol. 124, no. 1, pp. 189–205, 2022. DOI:https://doi.org/10.1007/s11277-021-09335-w.

[16] 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 of Things Journal*, vol. 8, no. 7, pp. 5476–5497, 2020. DOI:10.1109/JIOT.2020.3030072.

[17] L. Ju, T. Zhang, S. Toor and A. Hellander. "Accelerating fair federated learning: Adaptive federated adam". *IEEE Transactions on Machine Learning in Communications and Networking*, 2024. DOI:https://10.1109/TMLCN.2024.3423648.

[18] D. Gao, X. Yao and Q. Yang. "A survey on heterogeneous federated learning". *arXiv preprint arXiv:2210.04505*, 2022. DOI:https://doi.org/10.48550/arXiv.2210.04505.

[19] S. Kamm, S. S. Veekati, T. Müller, N. Jazdi and M. Weyrich. "A survey on machine learning based analysis of heterogeneous data in industrial automation". *Computers in Industry*, vol. 149, pp. 103930, 2023. DOI:https://doi.org/10.1016/j.compind.2023.103930.

[20] C. Xu, Y. Qu, Y. Xiang and L. Gao. "Asynchronous federated learning on heterogeneous devices: A survey". *Computer Science Review*, vol. 50, pp. 100595, 2023. DOI:https://doi.org/10.1016/j.cosrev.2023.100595.

[21] H. Zhu, J. Xu, S. Liu and Y. Jin. "Federated learning on non-IID data: A survey". *Neurocomputing*, vol. 465, pp. 371–390, 2021. DOI:https://doi.org/10.1016/j.neucom.2021.07.098.

[22] Q. Li, Y. Diao, Q. Chen and B. He. "Federated learning on non-iid data silos: An experimental study". In *2022 IEEE 38th international conference on data engineering (ICDE)*, pp. 965–978. IEEE, 2022. DOI:https://DOI:10.1109/ICDE53745.2022.00077.

[23] 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. DOI:https://https://doi.org/10.1016/j.future.2022.05.003.

[24] H. Jamali-Rad, M. Abdizadeh and A. Singh. "Federated learning with taskonomy for non-IID data". *IEEE transactions on neural networks and learning systems*, vol. 34, no. 11, pp. 8719–8730, 2022. DOI:https://10.1109/TNNLS.2022.3152581.

[25] L. Cao. "Beyond iid: Non-iid thinking, informatics, and learning". *IEEE Intelligent Systems*, vol. 37, no. 4, pp. 5–17, 2022. DOI:https://10.1109/TNNLS.2022.3152581.

[26] H. Azami, S. Sanei and T. K. Rajji. "Ensemble entropy: A low bias approach for data analysis". *Knowledge-Based Systems*, vol. 256, pp. 109876, 2022. DOI:https://doi.org/10.1016/j.knosys.2022.109876.

[27] D. Rajput, W.-J. Wang and C.-C. Chen. "Evaluation of a decided sample size in machine learning applications". *BMC bioinformatics*, vol. 24, no. 1, pp. 48, 2023. DOI:https://doi.org/10.1186/s12859-023-05156-9.

[28] N. MacNell, L. Feinstein, J. Wilkerson, P. M. Salo, S. A. Molsberry, M. B. Fessler, P. S. Thorne, A. A. Motsinger-Reif and D. C. Zeldin. "Implementing machine learning methods with complex survey data: Lessons learned on the impacts of accounting sampling weights in gradient boosting". *Plos one*, vol. 18, no. 1, pp. e0280387, 2023. DOI:https://doi.org/10.1371/journal.pone.0280387.

[29] J. Chen, Y. Hao, T. Wang, D. Huang and X. Liu. "Discovery of Stomach Adenocarcinoma Biomarkers by Consensus Scoring of Random Sampling and Machine Learning Modeling". In *2022 10th International Conference on Bioinformatics and Computational Biology (ICBCB)*, pp. 112–115. IEEE, 2022. DOI:https://DOI:10.1109/ICBCB55259.2022.9802469.

[30] J. Qi, T. W. Ko, B. C. Wood, T. A. Pham and S. P. Ong. "Robust training of machine learning interatomic potentials with dimensionality reduction and stratified sampling". *npj Computational Materials*, vol. 10, no. 1, pp. 43, 2024. DOI:https://doi.org/10.1038/s41524-024-01227-4.

[31] Y. Deng, F. Lyu, J. Ren, Y.-C. Chen, P. Yang, Y. Zhou and Y. Zhang. "Fair: Quality-aware federated learning with precise user incentive and model aggregation". In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*, pp. 1–10. IEEE, 2021. DOI:https://DOI:10.1109/INFOCOM42981.2021.9488743.

[32] G. Liu, X. Ma, Y. Yang, C. Wang and J. Liu. "Federaser: Enabling efficient client-level data removal from federated learning models". In *2021 IEEE/ACM 29th international symposium on quality of service (IWQOS)*, pp. 1–10. IEEE, 2021. DOI:https://10.1109/IWQOS52092.2021.9521274.

[33] A. A. Al-Saedi, V. Boeva and E. Casalicchio. "Fedco: Communication-efficient federated learning via clustering optimization". *Future Internet*, vol. 14, no. 12, pp. 377, 2022. DOI:https://doi.org/10.3390/fi14120377.

[34] A. Z. Tan, H. Yu, L. Cui and Q. Yang. "Towards personalized federated learning". *IEEE transactions on neural networks and learning systems*, vol. 34, no. 12, pp. 9587–9603, 2022. DOI:https://DOI:10.1109/TNNLS.2022.3160699.

[35] X. Yu, L. Li, X. He, S. Chen, L. Jiang *et al.*. "Federated learning optimization algorithm for automatic weight optimal". *Computational Intelligence and Neuroscience*, vol. 2022, 2022. DOI:https://doi.org/10.1155/2022/8342638.

[36] J. Wolfrath, N. Sreekumar, D. Kumar, Y. Wang and A. Chandra. "Haccs: Heterogeneity-aware clustered client selection for accelerated federated learning". In *2022 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pp. 985–995. IEEE, 2022. DOI:https://10.1109/IPDPS53621.2022.00100.

[37] J. Qi, F. Lin, Z. Chen, C. Tang, R. Jia and M. Li. "High-quality model aggregation for blockchain-based federated learning via reputation-motivated task participation". *IEEE Internet of Things Journal*, vol. 9, no. 19, pp. 18378–18391, 2022. DOI:https://10.1109/JIOT.2022.3160425.

[38] J. Zhang and Z. Li. "A Clustered Federated Learning Method of User Behavior Analysis Based on Non-IID Data". *Electronics*, vol. 12, no. 7, pp. 1660, 2023. DOI:https://doi.org/10.3390/electronics12071660.

[39] C. Wu, Z. Li, F. Wang and C. Wu. "Learning cautiously in federated learning with noisy and heterogeneous clients". In *2023 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 660–665. IEEE, 2023. DOI:https://10.1109/ICME55011.2023.00119.

[40] Q. Sun, X. Li, J. Zhang, L. Xiong, W. Liu, J. Liu, Z. Qin and K. Ren. "Shapleyfl: Robust federated learning based on shapley value". In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2096–2108, 2023. DOI:https://doi.org/10.1145/3580305.3599500.

[41] C. Tu, S. Zhao and H. Deng. "FedWNS: Data Distribution-Wise Node Selection in Federated Learning via Reinforcement Learning". In *2023 26th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, pp. 600–605. IEEE, 2023. DOI:https://10.1109/CSCWD57460.2023.10152675.

[42] M. Z. Hossain and A. Imteaj. "Fedavo: Improving communication efficiency in federated learning with african vultures optimizer". *arXiv preprint arXiv:2305.01154*, 2023. DOI:https://10.1109/COMPSAC61105.2024.00069.

[43] Z. Tao, J. Wu and Q. Li. "Preconditioned Federated Learning". *arXiv preprint arXiv:2309.11378*, 2023. DOI:https://doi.org/10.48550/arXiv.2309.11378.

[44] I. Adjei-Mensah, X. Zhang, I. O. Agyemang, S. B. Yussif, A. A. Baffour, B. M. Cobbinah, C. Sey, L. D. Fiasam, I. A. Chikwendu and J. R. Arhin. "Cov-Fed: Federated learning-based framework for COVID-19 diagnosis using chest X-ray scans". *Engineering Applications of Artificial Intelligence*, vol. 128, pp. 107448, 2024. DOI:https://doi.org/10.1016/j.engappai.2023.107448.

[45] F. Wu, Z. Li, Y. Li, B. Ding and J. Gao. "Fedbiot: Llm local fine-tuning in federated learning without full model". In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 3345–3355, 2024. DOI:https://.

[46] S. M. Hamidi, R. Tan, L. Ye and E.-H. Yang. "Fed-it: Addressing class imbalance in federated learning through an information-theoretic lens". In *2024 IEEE International Symposium on Information Theory (ISIT)*, pp. 1848–1853. IEEE, 2024. DOI:https://10.1109/ISIT57864.2024.10619204.

[47] X. Yang. "A historical review of collaborative learning and cooperative learning". *TechTrends*, vol. 67, no. 4, pp. 718–728, 2023. DOI:https://doi.org/10.1007/s11528-022-00823-9.

[48] S. Itahara, T. Nishio, Y. Koda, M. Morikura and K. Yamamoto. "Distillation-based semi-supervised federated learning for communication-efficient collaborative training with non-iid private data". *IEEE Transactions on Mobile Computing*, vol. 22, no. 1, pp. 191–205, 2021. DOI:10.1109/TMC.2021.3070013.

[49] Y. Zhang, S. Gao, P. Cai, Z. Lei and Y. Wang. "Information entropy-based differential evolution with extremely randomized trees and LightGBM for protein structural class prediction". *Applied Soft Computing*, vol. 136, pp. 110064, 2023. DOI:doi.org/10.1016/j.asoc.2023.110064.

[50] J. Zhang, Z. Li, B. Li, J. Xu, S. Wu, S. Ding and C. Wu. "Federated learning with label distribution skew via logits calibration". In *International Conference on Machine Learning*, pp. 26311–26329. PMLR, 2022.

[51] T. Sheng, C. Shen, Y. Liu, Y. Ou, Z. Qu, Y. Liang and J. Wang. "Modeling global distribution for federated learning with label distribution skew". *Pattern Recognition*, vol. 143, pp. 109724, 2023. DOI:https://doi.org/10.1016/j.patcog.2023.109724.

[52] A. Li, J. Sun, B. Wang, L. Duan, S. Li, Y. Chen and H. Li. "Lotteryfl: Empower edge intelligence with personalized and communication-efficient federated learning". In *2021 IEEE/ACM Symposium on Edge Computing (SEC)*, pp. 68–79. IEEE, 2021. DOI:https://10.1145/3453142.3492909.

[53] S. X. Lee and G. J. McLachlan. "An overview of skew distributions in model-based clustering". *Journal of Multivariate Analysis*, vol. 188, pp. 104853, 2022. DOI:https://doi.org/10.1016/j.jmva.2021.104853.

[54] B. McMahan, E. Moore, D. Ramage, S. Hampson and B. A. y Arcas. "Communication-Efficient Learning of Deep Networks from Decentralized Data". In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, edited by A. Singh and J. Zhu, volume 54 of *Proceedings of Machine Learning Research*, p. 1273–1282, Fort Lauderdale, FL, USA, 20–22 Apr 2017. PMLR.

[55] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar and V. Smith. "Federated Optimization in Heterogeneous Networks". In *Proceedings of Machine Learning and Systems 2020, MLSys 2020*, edited by I. S. Dhillon, D. S. Papailiopoulos and V. Sze, volume 2, p. 429–450, Austin,TX, USA, mar 2020. mlsys.org.

[56] B. Koonce and B. Koonce. "ResNet 50". *Convolutional neural networks with swift for tensorflow: image recognition and dataset categorization*, pp. 63–72, 2021. DOI:doi.org/10.1007/978-1-4842-6168-2_6.

[57] Scikit-learn. *train_test_split: Split arrays or matrices into random train and test subsets*, 2023. Accessed: 2024-09-05.

[58] S. Zheng, T. Ye, X. Li and M. Gao. "Federated Learning via Consensus Mechanism on Heterogeneous Data: A New Perspective on Convergence". In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7595–7599. IEEE, 2024. DOI:https://10.1109/ICASSP48485.2024.10446892.

[59] J. Xu, Y. Yan and S.-L. Huang. "FedPer++: toward improved personalized federated learning on heterogeneous and imbalanced data". In *2022 International Joint Conference on Neural Networks (IJCNN)*, pp. 01–08. IEEE, 2022. DOI:https://10.1109/IJCNN55064.2022.9892585.

[60] S. Ullah and D.-H. Kim. "Federated Learning convergence on IID features via optimized local model parameters". In *2022 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pp. 92–95. IEEE, 2022. DOI:https://DOI:10.1109/BigComp54360.2022.00028.

[61] M. Morafah, M. Reisser, B. Lin and C. Louizos. "Stable Diffusion-based Data Augmentation for Federated Learning with Non-IID Data". *arXiv preprint arXiv:2405.07925*, 2024. DOI:https://doi.org/10.48550/arXiv.2405.07925.

[62] Z. Yang, Y. Zhang, Y. Zheng, X. Tian, H. Peng, T. Liu and B. Han. "FedFed: Feature distillation against data heterogeneity in federated learning". *Advances in Neural Information Processing Systems*, vol. 36, 2024.