IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 17 May 2025, accepted 2 June 2025, date of publication 10 June 2025, date of current version 23 July 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3578301

RESEARCH ARTICLE

Adding Data Quality to Federated Learning Performance Improvement

ERNESTO GURGEL VALENTE NETO[1], SOLON ALVES PEIXOTO JR.[2], VALDERI REIS QUIETINHO LEITHARDT[3], (Senior Member, IEEE), JUAN FRANCISCO DE PAZ SANTANA[4], AND JULIO C. S. DOS ANJOS[5,6]

[1]PPGETI, Federal University of Ceará, Fortaleza 60020-181, Brazil
[2]Department of Data Science, Federal University of Ceará Campus Itapajé, Itapajé 62600-000, Brazil
[3]Instituto Universitário de Lisboa (ISCTE-IUL), ISTAR, 1649-026 Lisboa, Portugal
[4]Expert Systems and Applications Laboratory, University of Salamanca, 37008 Salamanca, Spain
[5]Department of Data Science, Federal University of Ceará Campus Itapajé, Graduate Program in Teleinformatics Engineering (PPGETI/UFC), Technological Center, Campus of Pici, Fortaleza, Ceará 60455-970, Brazil
[6]National Institute of Science and Technology (INCT-Signals), Campus of Pici, Fortaleza, Ceará 60455-970, Brazil

Corresponding author: Ernesto Gurgel Valente Neto (gurgelvalente@alu.ufc.br)

This work was supported in part by the Coordination for the Improvement of Higher Education Personnel-Brazil (CAPES) under Grant 001; in part by the National Funds under FCT-Fundação para a Ciência e Tecnologia, I.P., within the scope of projects under Grant UIDB/04466/2020 and Grant UIDP/04466/2020; in part by the National Institute of Science and Technology (INCT-Signals), sponsored by Brazil's National Council for Scientific and Technological Development (CNPq) under Grant 406517/2022-3; and in part by the CEREIA Project São Paulo Research Foundation (FAPESP), FAPESP-Ministério da Ciência, Tecnologia e Inovação (MCTI)-Comitê Gestor da Internet no Brasil (CGI.br) in partnership with the Hapvida NotreDame Intermedica Group under Grant 2020/09706-7.

ABSTRACT Massive data generation from Internet of Things (IoT) devices increases the demand for efficient data analysis to extract relevant and actionable insights. As a result, Federated Learning (FL) allows IoT devices to collaborate in Artificial Intelligence (AI) training models while preserving data privacy. However, selecting high-quality data for training remains a critical challenge in FL environments with non-independent and identically distributed (non-iid) data. Poor-quality data introduces errors, delays convergence, and increases computational costs. This study develops a data quality analysis algorithm for both FL and centralized environments to address these challenges. The proposed algorithm reduces computational costs, eliminates unnecessary data processing, and accelerates the convergence of AI models. The experiments utilized the MNIST, Fashion-MNIST, CIFAR-10, and CIFAR-100 datasets, and performance evaluation was based on main literature metrics, including accuracy, recall, F1 score, and precision. Results show a maximum observed execution time reduction of up to 56.49%, with an accuracy loss of approximately 0.50%.

INDEX TERMS Data quality, deep learning, federated learning, IoT, IID, non-IID.

I. INTRODUCTION

Advances in Artificial Intelligence (AI) and the Internet of Things (IoT) have a measurable impact in multiple sectors, enabling intelligent, data-driven applications. However, this development presents challenges for data security and mobility [1], particularly in the healthcare sector, where these technologies play a crucial role in protecting patient privacy and ensuring compliance with confidentiality regulations [2].

In scenarios where data protection is paramount, approaches to effective monitoring and management are particularly relevant, ensuring regulatory compliance and the safety of patient data [3], [4]. In this context, the ability to process large amounts of data efficiently and the collaboration between neural networks are increasingly regarded as essential for various applications [5], [6].

From this perspective, Federated Learning (FL) has demonstrated potential in the literature, allowing IoT devices to collaborate to create a neural network without sharing raw data. Edge devices share their local model variables by

The associate editor coordinating the review of this manuscript and approving it for publication was Thomas Canhao Xu.

updating a global model, which in turn updates local devices until a global minimum is reached [7].

Despite its promise, this approach introduces several challenges, such as handling Independent and Identically Distributed (iid) and Non-Independent and Identically Distributed (non-iid) datasets [8], [9], [10]. It also addresses issues such as mitigating the high communication costs due to data dispersion and the complexity of selecting clients during model training. These factors affect learning convergence and prolong execution time, which leads to increased computational demands and a more computationally intensive process on edge devices, which have hardware and energy limitations.

Furthermore, establishing incentives for collaboration and managing device heterogeneity remains a significant challenge [11], [12]. Consequently, addressing computational demands requires automating cost management, increasing resource availability [13], capturing temporal dependencies [14], and ensuring data quality at the edge [15].

In this context, state-of-the-art studies have explored solutions that demonstrate the significance of signal processing [16], with a focus on device heterogeneity and data privacy protection. These approaches aim to optimize resource efficiency and improve the training of FL models while ensuring robust privacy support [11], [17].

Ultimately, these studies highlight fundamental issues in managing large data volumes, high communication costs, and the complexity of data selection. Furthermore, the nature of non-iid data introduces additional complexity to the training models.

Current FL approaches do not evaluate the quality of the data at the edge, leading to low-quality inputs and inefficient training. Addressing this limitation, this study proposes an entropy-based data selection method to optimize FL performance with the **following:**

i. providing a data quality analysis algorithm on edge to select data with the highest informational value, maintaining both class balance and accuracy levels comparable to the original FL models;
ii. reducing unnecessary data processing with low data quality to save energy on the edge; and
iii. improving FL the computation performance by the reducing execution time by 50% in IoT devices.

In general, at the top layer, the **FL Model and Aggregation Server** orchestrate clients (nodes), detect rare events, and ensure resistance to poisoning attacks or failures, such as communication cost [18]. The responsibilities may include techniques such as feature extraction [19], dynamic regularization [20], node selection [21], client clustering [22], client sampling [23], client contributions [24], and adaptive selection [25], as well as layers of security, increased fairness in collaboration between clients, and poison attack mitigation and defense mechanisms [26], [27].

For instance, **upper-layer aspects** – such as client orchestration and connection problem management – are commonly addressed in studies of global aggregation algorithms on the aggregation server in FL approaches. Alternatively, designers may integrate these aspects with broader solutions (e.g., cryptography, blockchain, and connection management).

In addition, the taxonomy in the Continuum of the Internet of Things (CIoT) [28], [29] assigns these responsibilities to the orchestrator, including managing connectivity, network resources, resource allocation, network management, and security across distributed edge, fog, and cloud layers. Some responsibilities are shared between the FL algorithms and the orchestrator, especially in scenarios where coordination, data availability, and system resilience are crucial for distributed training and aggregation processes.

This work proposes an **agnostic algorithm** (i.e., a new AI layer as a data preparation step in the edge) **without interfering** with the **FL** algorithm execution **or the server** where aggregation occurs. The algorithm is based on edge data quality evaluation, which removes data without relevant information for training and improves the convergence of training algorithms by selecting the most significant information from the input data based on the entropy metric.

The **algorithm operates at the edge** and selects the best aggregated information by processing on the edge devices. This design choice ensures that the proposed approach does not depend on or interfere with the functionalities of the **FL Model or Aggregation Server**. As a result, the algorithm can be integrated into different FL pipelines regardless of the server's adopted aggregation strategies or management. Due to this, rare event detection, FL security, attacks that compromise the integrity or reliability of the neural model hosted on the aggregation server, and communication issues **are outside the scope of this work.**

The remainder of this paper is structured as follows. Section III provides an appropriate background, and Section III covers Related Work. Section IV presents the proposed model in detail. Section V presents the methodology and the evaluated scenarios. Section VI presents the evaluations and the results achieved. Section VIII presents the final considerations, Section IX outlines future directions, and finally, Section Appendix A contains more detailed information about the state-of-the-art.

II. PRELIMINARIES AND BACKGROUND

This section describes and contextualizes three concepts: 1) Iid data with centralized learning, where independent and identically distributed data are collected centrally. 2) Non-iid data with FL addresses decentralized data, reflecting uneven distribution and non-independent data. 3) Entropy measures uncertainty or disorder. Finally, it introduces the Related Work and outlines the primary challenges addressed in this study.

A. IID AND NON-IID DATA

Iid data refers to observations where each sample is drawn from the same probability distribution, independently of others. For example, in a production system that consistently

generates data following a distribution, it becomes possible to obtain statistical inferences about these characteristics.

The iid property assumes that each observation is independent of the others [30], meaning that one sample does not influence another. Such an assumption allows statistical tools, such as the law of large numbers and the central limit theorem, to generalize the results. However, these assumptions are often not satisfied in the real world. Data are commonly correlated or exhibit sample heterogeneity [31].

These conditions violate the iid data assumptions, where the observations do not follow the same probabilistic distribution, and the observations may exhibit high correlation with each other or be strongly correlated rather than having more distant relationships. Furthermore, dynamic environments may render iid assumption invalid or inapplicable.

The distributions change over time, leading to an effect referred to in the literature as concept drift, which represents a shift or evolution in the data that invalidates the created AI model. Figure 1 presents a visual representation of these concepts.

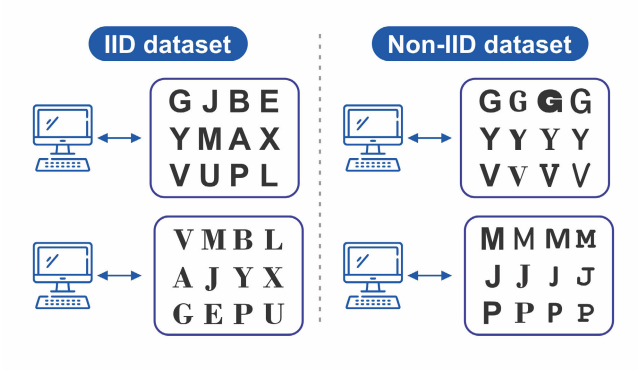


FIGURE 1. The iid and non-iid data.

Moreover, non-iid data violate at least one of the conditions that define iid data; each observation or sample must be independent of the others, and all samples originate from the same probabilistic distribution.

More specifically, non-iid data involve correlated distributions, where samples may exhibit mutual statistical dependence, and probabilistic distributions may vary among data. Additionally, non-iid datasets may present different subsets of data that follow different distributions.

B. CENTRALIZED LEARNING AND FEDERATED LEARNING

In centralized Machine Learning (ML), data are collected, stored, and processed centrally on a dedicated server or in a centralized location. This architecture promotes efficiency in statistical modeling and pattern detection, thereby enabling the deployment of ML algorithms that require large amounts of data to achieve generalization and produce reliable outcomes.

However, centralized models present challenges, primarily related to data security and the collection and centralization of large volumes of data. Therefore, this exposes sensitive information to risks, which contributes to leakage or cyber-attacks [32]. Centralization also leads to issues related to latency, where data from different sources is centralized on a server, consuming a significant amount of communication, especially with geographically distributed data [33].

Unlike the centralized approach, FL follows a decentralized strategy for training ML models. Data from different sources contribute to training multiple devices or nodes in a network (clients), such as smartphones, tablets, IoT sensors, and other edge-computing devices. Each device uses its data to train an AI model and then sends model parameter updates to a central server that aggregates the updates from the parameters of the neural network. In this approach, private data remains on edge devices, never being shared directly, thereby respecting ethical and legal perspectives in sensitive data contexts. Furthermore, it reduces massive data transfers and the risk of large-scale data leaks [34].

Figure 2 compares ML architectures. In Figure 2 (1), the data from multiple devices are centralized and stored for model training. In contrast, Figure 2 (2) depicts decentralized training, where the data remains on the devices, and the model trains locally. A global aggregation algorithm combines updates from the neural networks of each device.

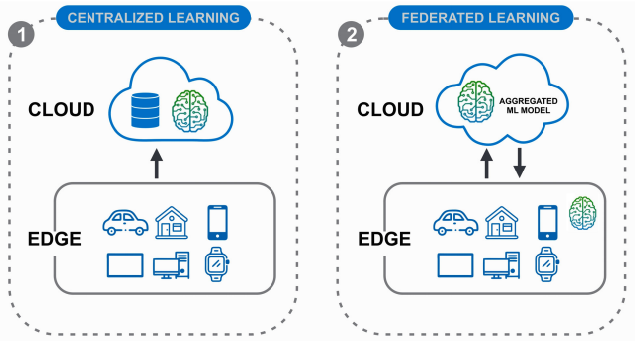


FIGURE 2. Centralized vs decentralized learning FL.

C. ENTROPY

Entropy is a central concept in information theory. It plays a fundamental role in understanding efficient communication and information transmission and providing a quantitative means of measuring uncertainty. Consider, for example, a simple system composed of coin flips. In such a system, both faces have an equal probability of occurrence. The entropy, denoted by $H(X)$, where X is a random variable that represents the result of each coin flip, can be calculated using the entropy formula. This calculation enables us to quantify the uncertainty associated with the information produced by the flips.

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

where:

i. X represents the set of all possible symbol values.

ii. $p(x_i)$ is the probability of occurrence of the i-th symbol value.
iii. The $\sum_{i=1}^{n}$ all possible values.
iv. $\log_2 p(x_i)$ is the logarithm with the base b of probability $p(x_i)$, making the entropy unit in bits.

In this case, there are heads and tails, where the entropy $H(X)$ represents a value of 1; thus, each coin provides one bit of information. However, with such a result, it is impossible to precisely determine the outcome because each outcome remains statistically unpredictable with equal likelihood.

However, if entropy is reduced, each new flip exhibits lower entropy and reduced informational uncertainty as outcomes become increasingly predictable. Conversely, uncertainty and unpredictability reach their maximum at maximum entropy, rendering system outcomes completely random.

Figure 3 presents the concepts discussed concerning entropy, highlighting its applicability as a metric for quantifying the uncertainty or the degree of unpredictability of an information source. It quantifies heterogeneity and randomness in data distributions within datasets, contributing to the selection of features that serve as a criterion for feature relevance.

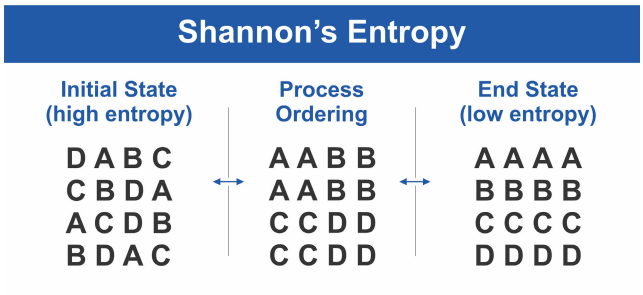


FIGURE 3. The figure illustrates entropy reduction through symbolic states, representing high, intermediate, and low entropy, which correspond to decreasing uncertainty.

Furthermore, entropy serves as a tool for analyzing the presence of noise or missing values in data samples, which, in turn, hinders the identification of patterns and negatively affects the convergence of ML models.

Furthermore, entropy contributes significantly to ML, particularly in decision tree classification algorithms, where it measures the degree of disorder and impurity in the dataset to minimize the uncertainty regarding the classes in each split. The information gain criterion, derived from entropy, is used to select the best data features, aiming for a more accurate classification [35]. About data quality, entropy is also used in ML; noisy or incomplete data tend to increase a system's entropy, making it more difficult to identify clear patterns and reducing the efficiency of AI models [36].

III. RELATED WORK

Recent advances in FL have contributed to measurable improvements in communication efficiency, precision, and convergence optimization, particularly when dealing with the challenges imposed by non-iid and heterogeneous environments. These issues are summarized in Table 1, which presents the FL approaches related to this study, along with their architectural and algorithmic features.

For instance, "FedAVO," a method inspired by natural optimization strategies to improve communication efficiency in FL, reflects an interest in solutions inspired by nature [37]. Likewise, "Fedco," which utilizes grouping optimization to increase communication efficiency, has been suggested for managing and effectively reducing data communication overload [38]. Orlandi et al. [39] presented the FedAvg-BE algorithm, which reduces the runtime in non-iid data in FL by up to 22% for MNIST and 26% for CIFAR-10 using edge entropy evaluation.

These advancements focus on improving communication efficiency, accuracy, and convergence in FL environments using non-iid data. Despite these advances, computational cost, communication, optimization, and data handling challenges in FL architectures remain crucial.

Some algorithms enhance learning efficiency and precision, such as those proposed by Yu et al. [20], which automatically adjust weights to achieve optimal performance. Preconditioned FL was also introduced in this context, proposing a method to precondition learning environments or data to enhance FL performance [40].

A subsequent study introduces "FedWNS," which utilizes node selection based on data distribution through learning by reinforcement, highlighting a node selection strategy to achieve better results [21]. In another approach, Wolfrath et al. considered heterogeneity and focused on selecting grouped clients, accelerating the FL process to address the challenges of client heterogeneity during the learning process.

In the Li et al. [50] approach, clients send their models to the server and share the distribution of their training data, providing the server with additional information and global optimization. However, this last approach violates the sensitive data strategy of the FL.

Other approaches have introduced the FL architecture based on blockchain technology. It leverages a data and model provenance ledger built on intelligent contracts and a fair and weighted data sampling algorithm [48]. Similarly, incentive mechanisms have been developed to increase participation and ensure collaboration more fairly [61].

Yang et al. [51] proposed an aggregating strategy that improves the model's convergence speed in non-iid environments by accounting for server-side characteristics with high variation. Similarly, Dolaat et al. [53], and Xu et al. [45] introduced strategies to enhance precision and personalize global FL models to address non-iid challenges. These studies incorporate techniques for incentive mechanisms, aggregation techniques, and balancing strategies in FL environments.

Ma et al. [12] explored the efficacy problem of AI model training in distributed scenarios, especially the solution to non-iid data in FL, and the relevance of efficient data

TABLE 1. Overview of federated learning studies by system characteristics and optimization techniques.

Author	Date	Properties				Strategies							
		Deep Learning	Machine Learning	Edge Computing	IoT Devices	Algorithm Optimization	Automatic Adjustment	Data/Client Selection	Data Distribution	Image Data Analysis	Data Quality Analysis	Aggregation Method Compatible	Embedded Compatible
Li, Beibei et al. [41]	2020	x				x				x			
Kang, Jiawen et al. [16]	2020	x											
Du, Zhaoyang et al. [42]	2020	x		x	x				x				
Itahara, Sohei, et al. [18]	2021	x	x		x	x		x	x				
Criado, Marcos F. et al. [10]	2022	x	x						x				
Gafni, Tomer et al. [43]	2022	x		x	x	x			x				
Al-Saedi, Ahmed A et al. [38]	2022	x		x	x	x			x	x			
Yu, Xi et al. [20]	2022					x	x		x	x			
Ullah, Shan et al. [44]	2022	x				x			x	x			
Xu, Jian et al. [45]	2022	x				x			x	x			
Wolfrath, Joel et al. [46]	2022	x		x	x	x		x	x	x			
Li, Yang, et al. [47]	2022	x								x	x		
Zhang, Yu, et al. [35]	2023	x	x			x				x			
Condori Bustincio, et al. [25]	2023	x	x		x			x	x				
Orlandi, Fernanda C. et al. [39]	2023	x		x	x	x	x	x	x	x			
Lo, Sin Kit et al. [48]	2023	x		x	x	x		x	x	x			
Hossain, Md Zarif et al. [37]	2023			x	x	x			x	x			
Tao, Zeyi et al. [40]	2023	x		x	x	x			x	x			
Lee, Hyeongok et al. [49]	2023					x			x	x			
Tu, Chengwu et al. [21]	2023	x		x	x	x		x	x	x			
Li, Boyuan et al. [50]	2023	x		x	x	x			x	x			
Yang, Wei-Jong et al. [51]	2023								x	x			
Chen, Huancheng et al. [19]	2023	x		x	x	x			x				
Zheng, Shu et al. [23]	2023					x			x	x			
Huang, Chenxi et al. [52]	2023	x				x			x	x			
Dolaat, Khalid Mahmoud Mohammad et al. [53]	2023		x	x	x	x			x	x			
Qiao, Yu et al. [54]	2023			x	x	x			x	x			
Sabah, Fahad et al. [55]	2023	x	x	x				x	x				
Li, Zexi et al. [22]	2023	x	x			x		x	x	x			
Wu, Chenrui et al. [56]	2023				x		x		x				
Sun, Qiheng et al. [24]	2023					x		x	x	x			
Iyer, Venkataraman Natarajan et al. [57]	2024		x						x	x			
Milan Ilić et al. [58]	2024	x	x	x		x							
Yan, Litao, et al. [59]	2024		x	x	x	x					x		
Hamidi, Shayan Mohajer, et al. [60]	2024	x	x		x	x	x	x	x				
Our Model	2024	x	x	x	x		x	x	x	x	x	x	x

selection methods. Parallel to this, other authors approach methodologies to improve convergence and address data heterogeneity through local model training adjustments and ''hyper-knowledge'' sharing [19], [44].

Researchers have introduced mechanisms to adjust gradients, optimize learning in environments with data heterogeneity, and optimize global structures, approaching critical challenges in the search for an efficient FL [23], [52]. These studies demonstrate the impact of data heterogeneity and how adjustments in local models, particularly at the edge, are currently considered a primary research objective in the literature.

The FL convergence improvement through regularization was recently proposed by Qiao et al. [54] to address the data heterogeneity between clients directly. Additionally, the FedGroup Framework incorporates a client clustering strategy using the K-means++ algorithm and optimization techniques, including meta-learning, adaptive optimization, and gradient aggregation strategies [55]. Another study by Ilic et al. [58] simultaneously examined several clients updating a global model.

The experiments involved aggregated updates using a method known as federated averaging. Incremental updates, in which the global model undergoes sequential updates, are also considered. This study also includes cyclic updates, where minor updates occur at the end of each epoch, and semi-simultaneous updates, which combine simultaneous and incremental strategies. Moreover, these studies

emphasize the importance of advanced strategies at nodes (clients), where these techniques have a significant impact on the learning processes and updates of global models.

Noise in the data under non-iid scenarios poses an additional challenge for global FL models. In this context, [56] proposed "FEDCNIA," an approach aimed at mitigating the impact of noise on FL clients. Furthermore, the discrimination in data distribution in FL [57] is explored using techniques to handle the non-iid nature of data in the healthcare and finance sectors. Regarding client data heterogeneity, [24] proposed an adaptive mechanism inspired by the Shapley value to promote greater client fairness. Furthermore, client consistency is a critical factor that has been explored in aggregation models. [22].

Thus, the FEDLAW algorithm demonstrates how data heterogeneity, the number of local epochs, and client variability influence the global model [62]. However, noise and data heterogeneity remain critical challenges in global aggregation techniques. The proposed model aims to mitigate quality loss and address data discrimination, ensuring fairness among clients and maintaining quality during training.

Finally, several studies in the medical field have explored the application of FL, emphasizing different aspects and challenges. For example, Antunes et al. [63] identified research questions regarding adoption and data aggregation mechanisms in Electronic Health Records (EHR). In medical imaging, researchers applied FL to address privacy concerns related to brain tumors [53]. Furthermore, technical challenges of FL, such as non-iid data, were discussed in [55] using heart rate data.

Additionally, [11] addresses issues related to complications in data transfer in the healthcare field and other medical data and applications. Furthermore, several studies have highlighted the applications of FL, emphasizing its importance in data privacy, sensitivity, and global aggregation models. These issues in the medical field address the technical challenges related to the heterogeneity of health data and the area of medical imaging.

A. PROBLEM OVERVIEW

This section explores the main topics of the problems addressed in this work and the state-of-the-art literature, focusing on challenges related to data quality in FL, particularly in non-iid scenarios.

The exponential growth of IoT devices and the massive volume of generated data pose significant challenges for practical analysis and the extraction of meaningful insights while addressing critical issues such as privacy protection, statistical heterogeneity, optimization and performance, and communication efficiency, all while preserving privacy and accuracy [64], [65].

These are especially critical scenarios where devices face limitations in battery, communication costs, latency, and synchronization. In the context of FL, data exhibits complex characteristics, such as non-iid properties, necessitating efficient algorithms and architectures that can securely process large data volumes, respect privacy, and enhance performance.

Thus, the central problem addressed in this work is data quality, which involves mitigating unnecessary processing by eliminating redundant information that does not add value to the neural network or significantly contribute to model accuracy.

The goal is to address device-related issues and reduce energy costs. This approach allows AI systems to train more rapidly while consuming fewer computational resources.

This work addresses these challenges by proposing an algorithm to enhance data quality in the FL context, with a focus on identifying information that efficiently contributes to the neural network and reducing the computational cost of training.

B. DISCUSSION OF PROBLEMS

The FL model inherently exhibits high latency, which leads to delayed convergence, infrequent model updates, synchronization issues, and an impact on accuracy, as well as increased energy consumption. Furthermore, one of the current limitations of the federated environment is the selection of nodes with intermittent participation or high churn rates by new nodes that incrementally introduce information.

In this context, data selection procedures often overlook input quality, instead focusing on the processing capacity and availability of the nodes (clients). The data that has a negligible impact on the model optimization results has consequences at the edge processing level. As a result, such data is underutilized and fails to produce a significant update in the local model weights, leading to unnecessary computation at the edge, excessive communication requirements, and increased energy consumption.

Since none of these new node sets are specifically validated, they allow for input data of varying quality, which can potentially increase latency and make it challenging to achieve high accuracy and low loss.

By contrast, the proposed model (e.g., Entropy-Based Selection (EnBaSe)) prioritizes both the quality of the nodes and their processing capacity. Thus, the EnBaSe entropy algorithm excludes information that does not significantly contribute to the model or has a limited contribution, selecting data that exhibits the highest information content based on entropy metrics.

Proper validation of the data's quality can increase its homogeneity. For example, suppose the average entropy of a data set is reduced from 4.8 to 4.6215, representing a decrease of 0.1785. This reduction means less uncertainty and unpredictability in the system, messages, or processed data.

When reduced, entropy, which measures global uncertainty, implies greater predictability and uniformity. Therefore, this results in images and data with a high standard of consistency, enhancing model regularity and eliminating inconsistent samples from the system. Removing these

sample inconsistencies accelerates the convergence of the neural network, enabling the model to be trained more quickly and efficiently. Consequently, this reduces energy consumption and processing, optimizing the system's performance.

For clarity, the MNIST dataset, comprised of 60,000 images in the iid scenario, has a total of 4.8 bits of entropy. Reducing entropy by removing 30,000 images results in an increase in accuracy from 98.95% to 99.27%, as shown in Table 9. This increase enhances the system's predictability and reduces the computational cost of processing the 30,000 images, which contributed negligibly to the model's predictive performance.

For the non-iid scenario, following the same pattern of data removal, this results in a loss of accuracy from 81.73% to 81.26%, as shown in Table 12. Consequently, the removed images add zero or negligible value (0.47%) to the neural network's accuracy, wasting processing time and energy.

IV. PROPOSED MODEL

This study tests the hypothesis and investigates the feasibility of applying information theory to quantify and analyze the quality of information in a dataset, as well as the uncertainty or surprise associated with its data distribution. Additionally, this evaluation focuses on improving input data quality and minimizing noise to reduce the computation time and energy costs.

In line with this perspective, the current state-of-the-art research explores the application of Entropy to quantify the degree of uncertainty and assess the redundancy present in information. These studies aim to comprehensively analyze information systems to measure the informational gain achieved through data processing.

In this context, Entropy serves as a metric to identify informative subsets of data that effectively contribute new information to a system. Analyzing redundancy enables a more detailed evaluation of distortions, data quality, and information reliability.

This method aims to reduce statistical uncertainty and provide a rigorous framework for quantifying information with low bias, thereby becoming a metric of information gain [47], [66]. Based on these principles, Entropy is approached as a strategy in deep neural networks to address the heterogeneity of data and clients [39], [60].

Furthermore, researchers argue that Entropy is a suitable metric to assess the degree of disorder in a system (e.g., dataset) [67]. As a measure of disorder, capturing this fundamental characteristic of the system is considered a suitable approach. Therefore, Entropy enables the quantification of inter-client data heterogeneity between clients' data, allowing for adaptive adjustments to achieve better convergence of the global model.

Additionally, Entropy is used to identify subsets of relevant and representative data. By serving as a metric to assess the relevance or diversity of data, it ensures that clients have meaningful and relevant information [25]. Additionally, some studies argue that Entropy reduces communication overload, as only relevant clients send updates, decreasing the required communication [18], [39], [59].

A. HYPOTHESIS

The hypothesis is that when Entropy applies to the field of Computer Vision (CV), each pixel set encodes color or intensity values, enabling the calculation of intensity values of a specific color or intensity. Therefore, the probability of each color or intensity value and each unique pixel occurrence can be calculated based on the frequency with which each specific color or intensity appears in the image.

Thus, high Entropy represents a great diversity of pixels, indicating a high complexity in texture, significant variation, and little predictability of the information. Conversely, low Entropy indicates greater image homogeneity, that is, better uniformity in identifying regions with little or no relevant information, facilitating the segmentation of elements in a scene.

This study further hypothesizes that the available data adds little value to the model and introduces noise into the training process. The following analogy illustrates this hypothesis: Initially, an unstructured image dataset presents high Entropy and great uncertainty. By organizing and separating these sets of images, this process divides the images into segments with low Entropy, considered statistically less informative or noisy.

In contrast, the other part exhibits high Entropy and continuous unpredictability. Consequently, data with high Entropy tends to be viewed as low-quality or noisy and is thus excluded from the training process to enhance model performance.

Thus, Shannon's entropy formula is employed as an estimation method for CV, to measure the frequency-based dispersion of pixel intensity values, aiming to quantify the visual complexity of an image. This approach disregards spatial structure and enables the identification of both simple and complex samples, as well as the recognition of stochastic variance, by assessing the statistical homogeneity and heterogeneity of pixel distributions.

B. REFERENCE MODEL AND OPERATION SCHEME

Figure 5 illustrates the operational stages of the proposed algorithm. Initially, Entropy is computed for each 2D image class, represented by matrices, forming key-value pairs, where the key is the image number, and the value is the corresponding Entropy. Subsequently, the process orders these entropy values sequentially within each class and partitions them using the median of the entropy distribution as a threshold, thereby selecting images from the class. This step applies to all classes in the dataset. The final step forwards the selected data to the neural network for training.

Figure 4 illustrates the integration of the proposed model with iid and non-iid data. Specifically, Figures 4 (**1**) and (**2**) demonstrate the scenario where, in (**1**), data from centralized devices are used for training an AI model, and in (**2**), global

parameters are sent by the devices for aggregation into a global model. Figure 4 (**3**) shows an intermediate layer that facilitates processing, aggregation, orchestration, device management, and security. Finally, Figure 4 (**4**) represents the dataset from different IoT devices that have been centralized or will be used in decentralized learning.

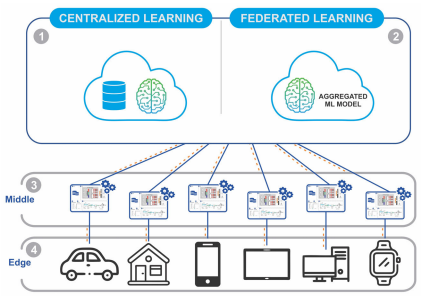


FIGURE 4. EnBaSe model applied to iid and non-iid scenarios.

C. ALGORITHM DETAILS

As established, according to state-of-the-art studies in Information Theory, Entropy is a mathematical and statistical tool used to measure the degree of disorder and information gain in a system. As a result, decreasing the Entropy within an information system increases the predictability of outcomes. In this context, the proposed algorithm extracts informative components by selecting the most informative samples, which are those with the highest information gain, within each subset.

Building on this concept, creating a data subset with reduced redundancy and noise is possible, which ensures a higher quality of the data subset. Consequently, neural networks can be trained more efficiently and at lower computational costs.

The Algorithm 1 is designed to be implemented in embedded systems, whether centralized servers or IoT devices. Specifically, among these devices is the most representative dataset from each data subset, using Entropy to identify the most informative and representative data.

As illustrated in Figure 5, the proposed model interacts with both the centralized and decentralized environments, as shown in Figure 4. Thus, the EnBaSe algorithm applies an entropy-driven selection method across various subsets, selecting the lower half of the entropy values for each class. This technique creates a more homogeneous sample within each subset and ensures a balanced representation in both centralized and decentralized contexts, thereby reducing computational, energy, and time costs.

The algorithm is embedded and receives training sets and labels, represented by Initialization, which occurs by creating two sets for data storage: $\mathcal{X}$selected and $\mathcal{Y}$selected. The algorithm iterates over each class from 0 to K-1, calculating the Entropy and measuring the degree of disorder for each image in each class. Finally, it retains samples with entropy below the class median. It returns the selected data in

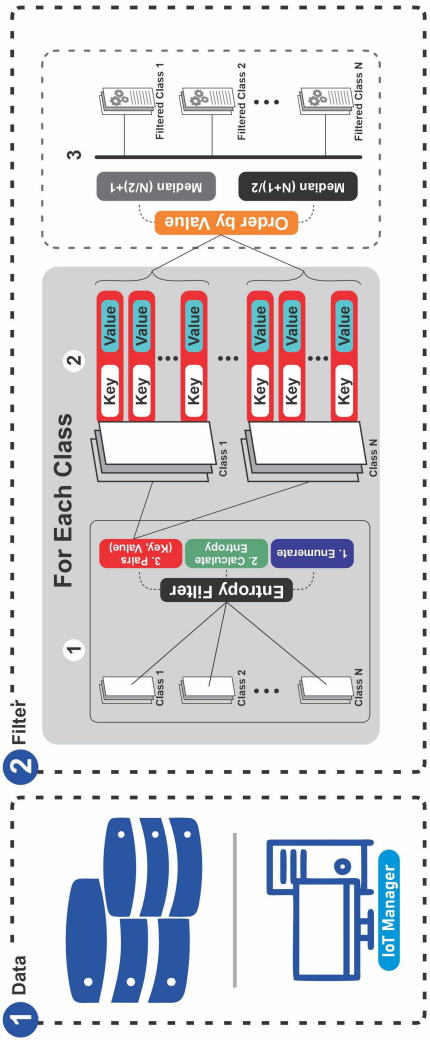


FIGURE 5. EnBaSe Algorithm: Entropy-based selection.

$\mathcal{X}$selected and $\mathcal{Y}$selected, which serve as input for training. Algorithm 1 is as follows:

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

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**
    $\mathcal{M}_{\text{Entropy}} \leftarrow$ (key, ComputeEntropy(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

xii. **H**: Entropy calculated for the image.
xiii. $p(image)$: Calculations are performed using the image represented as a one-dimensional (1D) array.

Algorithm 1 operates through several stages, detailed as follows:

i. **Initialization**: Two empty sets ($\mathcal{X}_{\text{selected}}$ and $\mathcal{Y}_{\text{selected}}$), are created to store the selected data and their corresponding classes.
ii. **Iteration over classes**: The algorithm iterates through each class present in the training set ($\mathcal{X}_{\text{train}}$ and $\mathcal{Y}_{\text{train}}$), identifying the indices associated with each class.
iii. **Entropy computation**: For each element in the current class, the ComputeEntropy function calculates the Entropy of the sample based on the probability distribution of its attributes. The results appear as (index, ComputeEntropy) pairs in $\mathcal{M}_{\text{Entropy}}$, which is also referred to as the entropy map.
iv. **Sorting and median-based selection**: The $\mathcal{M}_{\text{Entropy}}$ pairs are sorted by Entropy, and the median entropy value is calculated. Only elements with entropy values less than or equal to the median are selected, ensuring the inclusion of the most representative and least redundant data.
v. **Updating the selected subsets**: The selected indices are used to copy the corresponding data from $\mathcal{X}_{\text{train}}$ and $\mathcal{Y}_{\text{train}}$ to the subsets $\mathcal{X}_{\text{selected}}$ and $\mathcal{Y}_{\text{selected}}$.
vi. **Final output**: At the end of the Iteration over all classes, the subsets $\mathcal{X}_{\text{selected}}$ and $\mathcal{Y}_{\text{selected}}$ contain the most informative and homogeneous data, optimized for model training.

In summary, the algorithm proposed in this study, Algorithm 1, utilizes entropy as a metric to identify relevant data subsets for each class, thereby reducing redundancy and noise while enhancing the quality of the selected data. The choice of lower Entropy is grounded in Information Theory, which asserts that systems with lower Entropy exhibit greater predictability. In addition to its data selection strategy, the EnBaSe is designed to operate independently, ensuring broad applicability across different FL methods.

This design choice means that EnBaSe operates independently of an embedded design. The approach remains decoupled from any specific FL algorithm. As a result, our algorithm is general-purpose and compatible with any global aggregation method.

This design provides practical advantages in scenarios where computational resources are limited. Thus, the data subsets are structured homogeneously to optimize the training process. For instance, in medical classification systems with computational power constraints, Algorithm 1 prioritizes more informative and less redundant features, reducing the computational effort required by the neural network. This approach enables rapid and highly accurate responses in real-time IoT systems.

D. MATHEMATICAL FORMULATION OF ENBASE

The model described at the beginning of Subsection IV-B, based on Hypothesis IV-A, is grounded in the principles of Information Theory to select data with greater information gain II-C. In particular, it applies Shannon entropy to quantify the degree of disorder in each image, enabling the detection of low-entropy samples that are most representative of the training distribution. The following formulations define the mathematical base of pseudocode in the Subsection IV-C.

Let the amostral space I represent a matrix of pixel values from an image, and $p(x_i)$ be its probability distribution of pixel values x_i in the image. Then, the Shannon entropy $H(I)$ is defined as:

$$H(I) = -\sum_{i=1}^{d} p(x_i) \log_2 p(x_i) \tag{2}$$

Entropy $H(I)$ quantifies the degree of uncertainty in new data: the higher the uncertainty, the more associated information. It is calculated using the $\log_2$ and measured in bits. Thus, $p(x_i)$ represents the frequency of occurrence of the pixel value x_i in bits, and a low $H(I)$ value indicates a low degree of uncertainty in the image. Therefore, a low entropy value implies high predictability, which benefits neural network training when specializing in a specific subset of data. This approach improves training efficiency and convergence behavior with fewer input data.

Thus, given a class $c \in \{1, \ldots, K\}$ with a sample set $\mathcal{I}_c = \{I_{c_1}, I_{c_2}, \ldots, I_{c_n}\}$, the sorted set of image entropies is defined as:

$$\mathcal{H}_c = \left\{ H(I_{c_j}) \,\middle|\, j = 1, \ldots, n \right\} \tag{3}$$

The selected subset $\mathcal{S}_c$ from the class c is defined based on the median Entropy from $H(I_{c_j})$ set, and consists of the image collections whose Entropy is less than or equal to $\mathcal{H}_c$:

$$\mathcal{S}_c = \{I_{c_j} \in \mathcal{I}_c \mid H(I_{c_j}) \leq \mathcal{H}_c\} \tag{4}$$

The data selected for training the neural network is represented by the union of selected subsets for all classes of $\mathcal{S}_c$.

$$\mathcal{S} = \bigcup_{c=1}^{K} \mathcal{S}_c \tag{5}$$

EnBaSe selects samples based on Entropy that is lower than or equal to the median value of $\mathcal{H}_c$ for each class c, thereby constructing a more informative subset $\mathcal{S}_c$. The final training set $\mathcal{S}$ is obtained as $\mathcal{S} = \bigcup_{c=1}^{K} \mathcal{S}_c$, combining all subsets across the K classes. The higher predictability and lower noise of the data reduce the vanishing gradient problem, improving the model's convergence and generalization performance with the data. As a result, the selected subset ($\mathcal{S}$) indirectly reduces computational costs and accelerates the convergence of the neural network models.

V. EXPERIMENTAL EVALUATION

This section provides a comprehensive analysis of the methodological approaches adopted during the execution of this experiment, encompassing the steps taken to ensure the reproducibility of the experiment by other researchers aiming to replicate it and to support a thorough and critical evaluation of the results obtained. Source code is available on GitHub[1] for reproducibility.

A. DATASET DESCRIPTION

The MNIST, Fashion MNIST, CIFAR-10, and CIFAR-100 datasets are commonly adopted for the training and evaluation of ML and CV. The scope of the experiment encompassed the same datasets as the iid and non-iid scenarios. The characteristics of the selected datasets are detailed as follows:

i. **MNIST:** Images in a grayscale of handwritten digits (0-9), divided into training and test sets. Researchers use these datasets to train models for recognizing and classifying these digits.
ii. **Fashion MNIST:** An alternative to MNIST contains images of fashion articles in categories (0-9), such as shirts and pants, in grayscale. These datasets are adopted to evaluate model robustness due to their increased challenge for accurate classification.
iii. **CIFAR-10:** This dataset presents colored images in different classes (0-9), including cars and animals, with training and test sets utilized for benchmarking in image recognition.
iv. **CIFAR-100:** Analogous to CIFAR-10, but with classes (0-99) providing more granularity, including categories such as people and various natural elements, this dataset introduces increased classification complexity due to the increased number of classes.

The experiments progressed from MNIST, Fashion-MNIST, and CIFAR-10 to CIFAR-100 to validate the performance of FL methods in distributed and heterogeneous CV environments using the EnBase Algorithm at the edge, given that modern edge devices handle high-dimensional visual data with significant class variation and complexity. As an example, the CIFAR-100 dataset provides a realistic benchmark for a standard dataset, enabling the reproducibility of our experiments.

In contrast, **validating our results by researchers is challenging without standard datasets**, due to the impossibility of reproducibility from the heterogeneous input data sources. **To address this limitation**, the datasets were selected for the experiment **based on the state-of-the-art review in Table 14**, which presents the main continuous and discrete datasets used in the literature, as well as studies characterized by unimodal input modalities related to the edge computing and CV challenges addressed in this experiment. Additionally, this table outlines FL algorithms, aggregation approaches, application scenarios, metrics used, advantages, disadvantages, and the most frequently used datasets in the CV area applied to experiments in FL.

Recognizing that, compared to the real world, the results have inherent limitations, **strategies proposed in the literature were adopted to simulate more realistic scenarios**. In this context, the heterogeneity and distributional properties of the data are considered, as presented in Subsection V-C and following [68], [69].

Therefore, we conclude that, **in this methodology**, the experiment strategy **relies on commonly used state-of-the-art datasets** (e.g., for FL in *edge* applications in the CV area), as well as on the **distributions** of these **datasets identified in the literature**.

B. IID EXPERIMENT CONFIGURATION

To address challenges in image processing and pattern recognition under iid conditions, this study selected four benchmark datasets to validate the proposed method in this experiment. Table 2 lists the MNIST, Fashion-MNIST, CIFAR-10, and CIFAR-100 datasets used.

TABLE 2. Datasets summary.

Dataset	Description	Format	Training/Test
MNIST	Hand-written Digits	28x28	60,000/10,000
Fashion-MNIST	Clothes	28x28	60,000/10,000
CIFAR-10	Assorted	32x32	50,000/10,000
CIFAR-100	100 Classes	32x32	50,000/10,000

We aim to utilize these datasets to address a range of benchmarks and challenges in CV. For MNIST and Fashion-MNIST, 20% of the data was designated for validation, whereas 10%

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

C. NON-IID EXPERIMENT CONFIGURATION

1) DATA DISTRIBUTION

For the creation of the non-iid scenario in federated learning, where the data is not independent and identically distributed, the non-iid scenario was constructed considering feature distribution skew, label distribution skew, and quantity skew [68]. These skew types are representative of the heterogeneous data characteristics [69].

The non-iid indices for the clients (nodes) are used as a quantitative metric of the degree of data distribution among the clients (nodes), focusing on the **feature distribution**, **label distribution skew**, and **quantity skew**. The key considerations are as follows:

i. **Distortion in Feature Distribution:** Feature distortion refers to the imbalance between different quantities of labels across various clients concerning a specific client, which significantly affects performance and training. This feature distortion results in each client (node) having different features that may correspond to the same label, which can contain different information. For instance, the same character can be written in various styles, such as stroke width or inclination variations, resulting in heterogeneous representations of the same label.

ii. **Distortion in Label Distribution:** Label distortion occurs when different clients (nodes) in distinct locations exhibit varied distributions owing to demographic differences. These variations result from demographic and contextual factors that affect the frequency of label occurrence for each client (node).

iii. **Quantity Distortion:** Quantity distortion refers to an imbalance in the number of specific labels within a client, which affects the amount of data available for a single client (node). This imbalance hinders training efficiency and model performance, resulting in an underrepresentation or overrepresentation of specific labels, which affects the model's overall balance.

Initially, nodes were constructed using randomly sampled datasets, based on available data, where sufficient or available data from specific classes is not guaranteed. This random distribution introduces heterogeneity across nodes because different nodes may receive varying amounts of data or data types.

The goal is to ensure that any variation between groups (nodes) results from a random rather than a systematic factor, thereby reflecting the inherent variability in IoT environments, permitting variability in client contributions to reflect the uneven data availability typical of real-world IoT deployments, enhancing the model's ability to generalize across heterogeneous conditions.

2) AGGREGATION ALGORITHM

This study selected aggregation approaches that are well-established in the FL literature. The objective of this selection was to validate the hypothesis that entropy serves as a quantitative metric for assessing data quality in a federated environment, thereby reducing latency, noise interference, computational overhead, and energy consumption in such environments. The experiment implemented the following global aggregation models:

- **FedAvg:** Computes a weighted average of model updates from each client. In this method, weights are typically proportional to the number of training samples held by each client, thus adjusting for any imbalances in the dataset.
- **FedProx:** FedProx modifies the local loss function by incorporating a proximal regularization term. This term penalizes large deviations of local model weights from the global model. Thus, FedProx aims to mitigate the effects of data and device heterogeneity, enhancing training stability and convergence under harmonious learning.

D. DEEP NEURAL NETWORKS

1) EXPERIMENTS UNDER iid CONDITIONS

The analysis was performed on the MNIST and Fashion-MNIST datasets without Transfer Learning (TL) or Data Augmentation (DA) using the Stochastic Gradient Descent (SGD) optimizer. MNIST had 32 batches, and Fashion-MNIST had 128 batches, both for ten epochs. In CIFAR-10, we applied DA and utilized the Adam optimizer with a batch size of 128 for 50 epochs. For CIFAR-100, we combined DA and TL with SGD, trained for 50 epochs in batches of 128, and included a training callback to enhance model convergence monitoring.

The model configurations are listed in Table 3. To ensure a consistent comparison across experiments of the proposed algorithm's performance, the same architectures were used with the complete dataset, allowing for an assessment of its performance and limitations when trained on all the data.

To improve convergence stability and generalization performance of Deep Learning (DL) algorithms in centralized scenarios with homogeneous data, pixel intensities were normalized to the [0,1] range by scaling with a factor of 1/255.

2) EXPERIMENTS UNDER non-iid CONDITIONS

For the MNIST and Fashion-MNIST datasets, Convolutional Neural Network (CNN) were used without applying TL or DA, optimized using SGD. For CIFAR-10 and CIFAR-100, the adapted ResNet-50 model was employed, incorporating DA and optimized with SGD. All models were trained for 50 epochs with batches of 128 examples.

Distinct normalization statistics were applied to each dataset under the FL scenario with heterogeneous data to normalize the MNIST, Fashion-MNIST, CIFAR-10, and CIFAR-100 datasets. For MNIST, the average and standard deviation were 0.1307 and 0.3081, respectively. In the Fashion-MNIST dataset, the average and standard deviation of the parameters are 0.2860 and 0.3530, respectively. For CIFAR-10, the average values were 0.4914, 0.4822,

TABLE 3. Model configurations iid.

MNIST	Fashion	CIFAR-10	CIFAR-100
CONV-1	CONV-1	CONV-1	ResNet50
MP	MP	BN	LAYER-1
CONV-2	DP-1	CONV-2	LAYER-2
DP-1	FC-128	BN	LAYER-3
BN	DP-2	MP	LAYER-4
MP	FC-10	DP-1	GAP
CONV-3	SOFTMAX	CONV-3	FL-512
BN		BN	BN
FC-128		CONV-4	DP-2
FC-10		BN	FL-100
SOFTMAX		MP	SOFTMAX
		DP-2	
		CONV-5	
		BN	
		CONV-6	
		BN	
		DP-3	
		FL-512	
		BN	
		DP-4	
		FL-10	
		SOFTMAX	

Note: BN: batch normalization; MP: max pooling; DP: dropout; FC: fully-connected layer; GAP: global average pooling. Layer 1: three blocks (Conv, BN, ReLU, MP); Layer 2: four blocks (Conv, BN, ReLU, MP); Layer 3: six blocks (Conv, BN, ReLU, MP); Layer 4: three blocks (Conv, BN, ReLU, MP).

and 0.4465, with standard deviations of 0.2023, 0.1994, and 0.2010, for the **Red, Blue, Green (RGB)** channels, respectively. Finally, for CIFAR-100, the averages were 0.5071, 0.4867, and 0.4408, and the standard deviations were 0.2675, 0.2565, and 0.2761, respectively, for the RGB channels. The model configurations are presented in Table 4 according to the previously explained criteria.

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
FL-500	FL-500	LAYER-3	LAYER-3
FL-10	FL-10	LAYER-4	LAYER-4
SOFTMAX	SOFTMAX	GAP	GAP
		FL-512	FL-512
		BN	BN
		DP-1	DP-1
		FL-10	FL-100
		SOFTMAX	SOFTMAX

Note: BN: batch normalization; MP: max pooling; DP: dropout; FC: fully-connected layer; GAP: global average pooling. Layer 1: three blocks (Conv, BN, ReLU, MP); Layer 2: four blocks (Conv, BN, ReLU, MP); Layer 3: six blocks (Conv, BN, ReLU, MP); Layer 4: three blocks (Conv, BN, ReLU, MP).

E. EVALUATION METHODS AND METRICS

This section evaluates the methods used to identify unbiased and high-quality neural networks. Accordingly, this section organizes the evaluation process based on the following metrics:

- **Dataset Utilization:** This study initially used the complete dataset to establish a baseline performance, offering a comprehensive view of the model's capabilities (experiment: All Data).
- **Random Selection Method:** A random selection method was applied to ensure diversity and impartiality during training. This technique is well-established in the literature. It is considered an essential step towards creating an adaptable and unbiased network to prevent overfitting and ensure accurate responses to new data challenges (in the experiment named Random).
- **Model Comparison:** The effectiveness of the trained models was assessed by comparing their performance across three distinct scenarios: using all available data, applying a 50% random selection of data, and employing our proposed algorithm (EnBaSe), which selects half of the dataset, utilizing 50% of the data.

Subsequently, the evaluation employed the following metrics for a comprehensive comparison:

- **Accuracy:** Accuracy measures the proportion of correct predictions among the total samples, calculated by dividing the number of correct predictions by the total number of samples.
- **Recall:** Recall evaluates the proportion of correctly identified true positives, calculated by dividing the number of accurate positive samples by the total number of positive samples plus false negatives.
- **F1-Score:** The F1-Score is the harmonic value of precision and recall, indicating a balance between them. Higher values indicate a better model performance. In this specific case, it will be used as an additional metric in the context of FL owing to the high heterogeneity of the data.
- **Loss:** The loss function measures the error between predictions, with specific functions tailored to each problem (e.g., cross-entropy for classification). The aim is to minimize such losses to improve the model.
- **Learning Curve:** Represents model performance over time, comparing training and validation to identify over- and under-fitting and verifying model convergence.

VI. EVALUATIONS AND RESULTS

A. DATA ANALYSIS

To verify our hypothesis about the role of entropy in the selection of information quality, we analyzed its behavior pattern. We examined how it was influenced by normalization and DA operations. Subsequently, we evaluated the impact of data normalization (scaling to a range of 0 to 255 to match the pixel sizes) on sample selection by comparing the entropy before and after normalization.

The results demonstrated that the entropy remained stable, with an average precision agreement of up to 14 decimal places. These results indicate that, regardless of absolute

value transformations, the probabilistic importance of the data – and, therefore, its entropy – remains invariant.

Consistent with theoretical expectations based on Information Theory, it is noteworthy that entropy, by definition, measures uncertainty and disorder. This invariance is an expected outcome, as relative probabilities, not absolute values, define entropy. Table 5 presents the datasets before and after normalization.

TABLE 5. Entropy comparison (MNIST).

Image Indexes	Before Normalization	After Normalization
27582	5.150483033018236	5.150483033018237
5760	5.242222088792437	5.242222088792438
29284	5.247872784912092	5.247872784912093
4484	5.291615062825755	5.291615062825755
18188	5.378906867665029	5.378906867665029
Image Indexes	Before DA	After DA
57362	1.4863968283704654	1.4863968283704652
37920	1.4966680960341756	1.4966680960341756
21618	1.5087870622820911	1.5087870622820911
29180	2.0933393541022376	2.0933393541022376
3637	2.0933393541022380	2.0933393541022376

Following a similar approach, the experiments applied linear geometric transformations through DA techniques, throughout which the entropy value showed no measurable change. These results confirm that entropy is determined by the intensity levels of the pixels, regardless of their spatial position. The geometric transformations applied in DA, such as rotations and translations, preserve the data structure and modify only the position of the pixels while keeping their intensity values invariant.

These findings suggest that entropy remains unaffected by geometric transformations, thus ensuring its relevance in areas where preserving specific informational properties is crucial.

B. ENTROPY DISTRIBUTION AND SAMPLE SELECTION IN DATASETS

This study also investigated the entropy distribution of the histograms by calculating the entropy value of each image to analyze the entropy distribution in the dataset, as shown in Figure 6; this analysis considers both iid and non-iid scenarios. Furthermore, the study also investigated the heterogeneous and imbalanced data distribution across various clients using statistical tests, such as the Shapiro-Wilk, Kolmogorov-Smirnov, D'Agostino, and Pearson tests, to analyze entropy characteristics.

During the implementation and evaluation of the EnBaSe algorithm, the analysis demonstrated that the entropy values, once computed and represented as histograms, are closely approximated by a Gaussian-like distribution. The same analysis consistently identified this statistical pattern in

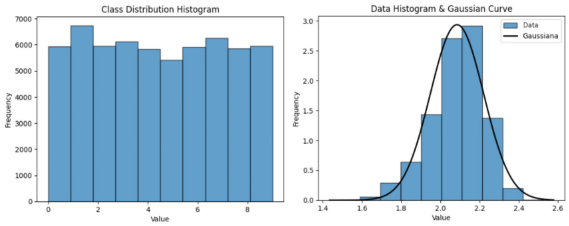


FIGURE 6. Distribution of classes and data distribution after entropy calculation for MNIST.

the MNIST, Fashion-MNIST, CIFAR-10, and CIFAR-100 datasets.

Table 6 presents the results obtained from the mean results in five independent trials for distribution analysis, where we emulated a distributed learning environment with 400 nodes receiving partitioned data in a distributed manner. In addition, we developed a consensus algorithm, as detailed in the table. This algorithm employs a voting mechanism to classify the distribution as usual, provided that at least two-thirds of the remaining statistical tests confirm these characteristics.

TABLE 6. Statistical test results for different dataset scenarios.

Dataset	Consensus	Shapiro-Wilk	Kolmogorov-Smirnov	D'Agostino & Pearson
MNIST	344	344	399	342
Fashion-MNIST	181	181	391	223
CIFAR-10	37	37	318	71
CIFAR-100	7	7	203	17

i. **MNIST (iid & non-iid):** The class distribution in MNIST is homogeneous and balanced in the iid environment. In contrast, in non-iid settings, after entropy selection, it is observed that data from a large part of the clients exhibit a near-normal distribution. Entropy plays a significant role in bringing the distribution closer to a Gaussian curve.
ii. **Fashion-MNIST (iid and non-iid):** In Fashion-MNIST, after selection, the Gaussian distribution presents itself less uniformly. In a considerable number of clients, the data follow a normal distribution.
iii. **CIFAR-10 (iid & non-iid):** In the CIFAR-10 set, a small proportion of the clients displayed normally distributed data after selection, indicating a less robust statistical distribution.
iv. **CIFAR-100 (iid & non-iid):** For CIFAR-100, a minimal number of clients (nodes) exhibited data distributions that approximated a normal distribution. This set was the most challenging in terms of approaching a Gaussian distribution.

The image datasets in the state-of-the-art literature cover a range of difficulty and semantic complexity, presenting both straightforward and more challenging labels for training AI models. From information theory, we know that data can possess high or low entropy and, consequently, we observe a distribution pattern that approaches a Gaussian distribution.

Therefore, these findings support the hypothesis that, for images, there is a dataset of complex images with high variability and low predictability. Similarly, some data exhibit low variability and high predictability, which is indicative of low entropy. As such, the EnBaSe model was constructed using an empirical and experimental approach based on information theory and state-of-the-art studies on entropy, focusing on low entropy and data with entropy values near the median of the observed distribution.

Finally, the same observation supports the selection of a more robust metric, such as the median. Although these results approximate a normal distribution, it cannot be concluded that all datasets strictly adhere to the same pattern. The asymmetric distributions identified in this experiment suggest that the mean may exhibit skewness, with the mean shifting toward the distribution tails, potentially leading to underestimation or overestimation.

C. ENTROPY COMPUTATIONAL COST

We conducted a total of 120 experiments under the iid condition experiments (10 simulations for All Data, 10 for EnBaSe, and 10 for Random). In the non-iid scenario, we performed 240 experiments: 10 simulations each for the complete set, EnBaSe, and Random in FedAvg, and an additional set of 10 simulations per configuration for the full set, EnBaSe, and Random in FedProx.

The entropy calculation times for the execution of the image-selection EnBaSe process are listed in Table 7, along with the average times for iid and non-iid environments, representing the computational time required to evaluate entropy throughout the entire dataset and to select representative samples.

TABLE 7. Average time for entropy computation.

Dataset	Distribution	Average (s)
MNIST	iid	≈1.76
Fashion MNIST	iid	≈1.83
CIFAR-10	iid	≈3.64
CIFAR-100	iid	≈3.67
MNIST	non-iid	≈2.77
Fashion MNIST	non-iid	≈3.21
CIFAR-10	non-iid	≈5.57
CIFAR-100	non-iid	≈5.49

Note: Average client (node) time in non-iid configuration; Centered mean time of dataset in iid configuration.

Additionally, Google Colaboratory (GC) was employed for simulations in a Cloud Computing (CC) environment. The T4 architecture was used, featuring high-speed memory, 12 Gigabytes (GB) of Random Access Memory (RAM), 15 GB Graphics Processing Units (GPU), and a 201.2 GB disk. The models were trained in the iid scenario in a centralized environment using Scikit-learn, reflecting the sequential nature of the operations without advanced parallelization. In contrast, TensorFlow was used for the non-iid scenario.

D. EVALUATION IN THE IID SCENARIO

This section discusses the experimental results obtained under the iid scenario. The iid scenario enables the evaluation of the hypothesis that entropy influences the training process as a measure of data quality. The results include analyses of the MNIST, Fashion-MNIST, CIFAR-10, and CIFAR-100 datasets across various configurations, focusing on the data distribution, accuracy, recall, and error rate.

The table 8 presents the datasets used in the experiment for the iid scenario within a centralized architecture. These datasets, widely recognized in the literature, are fundamental for advancing ML as they enable the evaluation of deep neural network performance under diverse challenges and varying levels of complexity.

The MNIST dataset serves as an essential starting point for model development due to its simplicity and widespread adoption in introductory studies. Conversely, the Fashion-MNIST dataset increases the complexity compared to MNIST, encompassing more diverse and challenging visual data. In contrast, the CIFAR-10 and CIFAR-100 datasets represent significant challenges as they involve image classification in more varied and complex scenarios, necessitating models with greater generalization capabilities.

TABLE 8. Datasets in centralized architecture.

Dataset	Description	Classes	Distribution	Architectures
MNIST	Handwritten digits	10	iid	Centralized
Fashion-MNIST	Clothing items	10	iid	Centralized
CIFAR-10	Various objects	10	iid	Centralized
CIFAR-100	100 categories	100	iid	Centralized

A total of 120 experiments were distributed equally among the All Data, Random, and Entropy categories, as shown in Figure 7. In the MNIST dataset experiments, the focus was on using entropy to guide data selection for this research. The experimental setup involved 30 experiments: 10 using the entire dataset, 10 with random selection, and 10 employing the proposed EnBaSe algorithm.

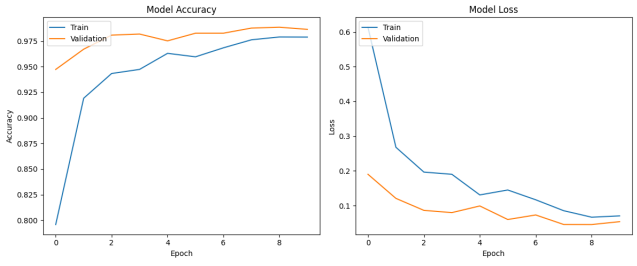


FIGURE 7. EnBaSe MNIST training.

Figure 7 shows that the training and validation curves of EnBaSe are close, and when the validation curve surpasses the training curve, it indicates better generalization. A higher

validation curve indicates potential benefits in regularization, thereby improving its generalization to new data.

The increase in the loss curve and reduction in the learning curve suggests that the model exhibits signs of overfitting the training data, indicating the need to adjust the learning rate to prevent deeper layers from learning less helpful patterns.

In the study involving the Fashion-MNIST dataset shown in Figure 8, we applied EnBaSe and presented a learning curve. We found a consistently higher validation curve than the training curve for the Fashion-MNIST dataset. Therefore, this suggests that the model effectively generalizes to unseen data, highlighting potential areas for improvement during the training process.

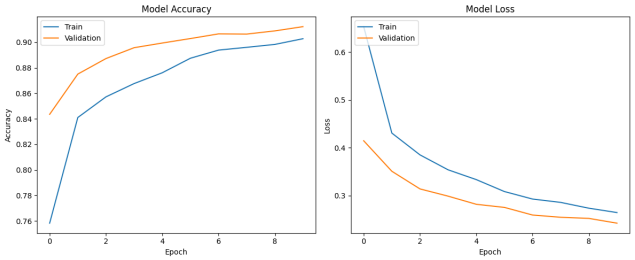


FIGURE 8. EnBaSe Fashion-MNIST training.

Finally, we observed a positive and steady evolution in the learning curve, accompanied by a continuous decrease in loss for both the training and testing data. Consequently, this indicates that with each epoch, the model enhances its capacity to minimize the loss function and demonstrates increasing accuracy.

In Figure 9, the overlay of the training and validation curves, with minimal variance at intersection points, generally indicates positive performance. As a result, the model consistently performed well on both the training and validation datasets. Their proximity suggests that the model is generalized efficiently, effectively transferring the knowledge acquired during training to the validation data.

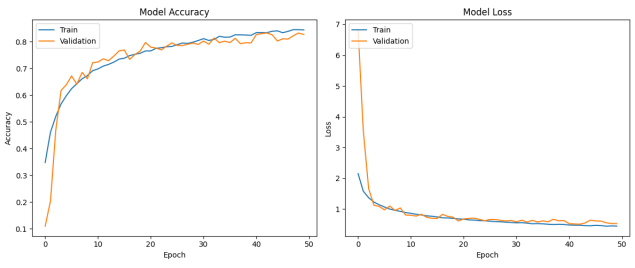


FIGURE 9. EnBaSe CIFAR-10 training.

This scenario indicates a balance between bias and variance. A low bias reveals that the model can understand the complexity inherent in the data. Simultaneously, a low variance suggests that the model does not overfit the training data, allowing for good performance on the new data.

The learning curve, with a constant and high learning rate, and the loss curve, with reduced values, suggest that the model may have approached its maximum potential within the constraints of its architecture and training settings.

As illustrated in Figure 10, the validation curve initially starts above the training curve, likely because of the composition of the training and validation sets, which enhances the learning effectiveness of the model. As the training progressed, the curves converged, indicating the model's ability to optimize its learning while maintaining generalization capability.

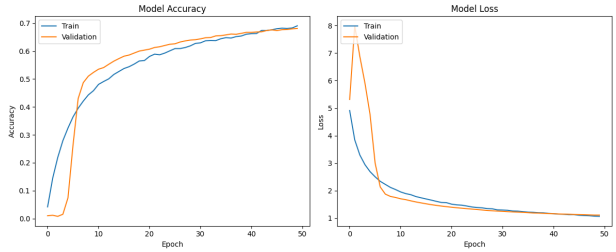


FIGURE 10. EnBaSe CIFAR-100 training.

The decline in the training and validation losses over time confirms that the model learns effectively and avoids overfitting. For CIFAR-100, a well-known benchmark in image recognition, the EnBaSe algorithm demonstrated strong performance in handling the dataset's complexity while preserving the model's generalization capability. For this reason, this highlights the effectiveness of EnBaSe in achieving efficient learning without introducing a bias that compromises the model's performance.

Table 9 presents the EnBaSe algorithm, which aims to enhance data quality by prioritizing consistency and precision, retaining only high-quality data, and minimizing computational cost. This algorithm was compared with a random selection method, which tends to minimize selection bias and make the model more robust and less dependent on specific features.

In doing so, observing the general scenario of complete datasets and entropy behavior about a robust technique is possible. For a fair comparison, half of the set was used as a random dataset.

By doing so, it is possible to observe the general scenario of complete datasets and entropy behavior regarding a robust technique. For a fair comparison, half of the set was used as a random dataset.

Furthermore, it presents the average results of the experiments with All Data, EnBaSe, and Random, which resulted in a minimal reduction in the accuracy for MNIST. For Fashion-MNIST, we observed the same pattern of accuracy. In CIFAR-10, the same accuracy pattern was observed.

Finally, for CIFAR-100, a benchmark in the CV field with 100 classes, a minimal and acceptable loss in processing cost savings was observed. Additionally, a reduction in overall

TABLE 9. Average results of the iid experiment.

Dataset	Type	Training			Validation			Test	Time (s)
		Accuracy (%)	Recall (%)	Loss	Accuracy (%)	Recall (%)	Loss	Accuracy (%)	
Mnist	All Data	≈ 99.53	≈ 99.52	≈ 0.038	≈ 98.95	≈ 98.94	≈ 0.049	≈ 99.19	≈ 73
Mnist	**EnBaSe**	≈ 99.28	≈ 99.27	≈ 0.070	≈ 98.64	≈ 98.61	≈ 0.050	≈ 97.61	≈ 32
Mnist	Random	≈ 99.49	≈ 99.49	≈ 0.055	≈ 99.49	≈ 98.77	≈ 0.058	≈ 98.93	≈ 36
Fashion	All Data	≈ 92.12	≈ 92.11	≈ 0.279	≈ 90.78	≈ 90.79	≈ 0.253	≈ 89.73	≈ 17
Fashion	**EnBaSe**	≈ 92.41	≈ 92.41	≈ 0.262	≈ 91.06	≈ 91.06	≈ 0.246	≈ 79.62	≈ 10
Fashion	Random	≈ 91.29	≈ 91.29	≈ 0.308	≈ 88.98	≈ 88.99	≈ 0.303	≈ 88.50	≈ 10
CIFAR-10	All Data	≈ 90.93	≈ 90.93	≈ 0.395	≈ 86.43	≈ 86.44	≈ 0.403	≈ 85.90	≈ 1,019
CIFAR-10	**EnBaSe**	≈ 89.22	≈ 89.21	≈ 0.441	≈ 82.05	≈ 82.03	≈ 0.558	≈ 78	≈ 501
CIFAR-10	Random	≈ 90.12	≈ 90.10	≈ 0.450	≈ 82.23	≈ 82.23	≈ 0.554	≈ 81.78	≈ 505
CIFAR-100	All Data	≈ 79.33	≈ 79.33	≈ 0.948	≈ 72.32	≈ 72.31	≈ 0.947	≈ 71.92	≈ 16,393
CIFAR-100	**EnBaSe**	≈ 77.38	≈ 77.37	≈ 1.114	≈ 67.68	≈ 67.69	≈ 1.134	≈ 65.04	≈ 10,398
CIFAR-100	Random	≈ 77.26	≈ 77.26	≈ 1.150	≈ 67.41	≈ 67.41	≈ 1.129	≈ 66.66	≈ 12,775

Note: The MNIST and Fashion models were trained for ten epochs, whereas the CIFAR-10 and CIFAR-100 models were trained for 50 epochs.

computational cost was evident for MNIST, Fashion-MNIST, and CIFAR-10 without a significant compromise in accuracy.

As shown in Table 9, the EnBaSe algorithm demonstrated robust and consistent results when selecting half of the datasets, preserving the accuracy with minimal possible loss of quality. Comparing the results obtained with the accuracy reported in other state-of-the-art studies, it is evident that the algorithm is robust, highly scalable, lightweight, and can be seamlessly integrated into embedded systems.

Furthermore, the algorithm exhibits high adaptability and can address various challenges in the field of CV using different architectures. Its efficiency is also evidenced by the reduction in computational costs and the acceleration of the convergence time of the AI model, enabling faster responses to events and dynamic environments, where time and computational cost are critical factors. For example, onboard health systems in the AI model can take a significant amount of time to acclimate to a patient's patterns.

Thus, we conclude that in the scenario with centralized data and iid, the EnBaSe algorithm is a computationally efficient solution that optimizes the selection of samples from the dataset with application in the field of CV. Its integration into embedded systems allows it to be applied dynamically as an AI tool in different centralized learning systems.

Table 10 presents a comparison with available data from other studies that utilized similar neural network architectures, including the training and validation sets provided by the authors in recent studies, which represent the state-of-the-art.

The experiments in this section demonstrate that the EnBaSe method may slightly underperform compared to training with the entire dataset, resulting in a minor reduction in accuracy. This difference becomes more pronounced in complex benchmarks, such as CIFAR-100 (Table 9).

Furthermore, the experiments demonstrate that, although computationally efficient, reducing the cost and convergence time by 50%–the EnBaSe method exhibits an average accuracy loss of 3%. On CIFAR-100, this reduction is even more significant, achieving an accuracy of 67.68%, compared to 72.32% for the "All Data" method during validation.

TABLE 10. Performance comparison (iid) with different works.

Dataset	Architecture	Author	Training	Validation
Mnist	CNN-2	[70]	97.07	-
Mnist	CNN	[71]	98.54	97.76
Mnist	CNN-1	[70]	99.21	-
Mnist	**CNN**	**EnBaSe**	99.28	98.64
Fashion	**CNN**	**EnBaSe**	92.41	91.06
Fashion	MCNN-14	[72]	93.08	-
Fashion	CNN-1	[73]	95.22	88.95
Fashion	CNN-2	[73]	98.01	93.11
CIFAR-10	CNN-2	[74]	85.90	-
CIFAR-10	CNN-1	[74]	87.57	-
CIFAR-10	**CNN**	**EnBaSe**	89.22	82.05
CIFAR-10	CNN	[71]	98.91	97.71
CIFAR-100	FC-CNN-Lab	[75]	42.26	-
CIFAR-100	CNN-1	[76]	63.50	-
CIFAR-100	CNN-2	[76]	68.60	-
CIFAR-100	CNN	**EnBaSe**	77.38	67.68

Note: The values of accuracy provided by the authors for training and validation.

The experiments on CIFAR-100, with only 500 images per class, challenged the EnBaSe method, which selects 250 high-quality images per class. Despite this sampling strategy, the limitation in sample size contributes to suboptimal results in this more complex scenario.

Finally, in Table 10, in the scenario with centralized and distributed data in a iid manner, EnBaSe performance aligns with findings from related studies available in the literature. The model achieves the highest accuracy on the MNIST dataset, with 99.28% in training and 98.64% in validation.

In the case of Fashion-MNIST, EnBaSe achieves an accuracy of 92.41%, compared to one of the best models, which reaches 98.91%. EnBaSe obtains 91.01% for validation, while the best model reaches 93.11%.

For CIFAR-10, EnBaSe achieved an accuracy of 89.22%, compared to the top-performing model, which achieved an accuracy of 98.91%. EnBaSe achieved 82.05% in validation, compared to 97.71% for the best model.

In the case of CIFAR-100, EnBaSe presented an accuracy of 77.38%, surpassing the second-best model, which achieved 68.60%. In this scenario, it is observed that most authors did

not provide validation samples, which limits more detailed comparisons.

E. EVALUATION IN A NON-IID SCENARIO

Table 11 presents the datasets considered for evaluation for the FL architecture. The selected datasets have varying levels of complexity and encompass a broad range of computational challenges in the field of ML. For instance, the MNIST dataset serves as a baseline for state-of-the-art experiments. In contrast, Fashion-MNIST offers greater complexity and represents a more complex classification task than MNIST. Finally, the CIFAR-10 and CIFAR-100 datasets present high-complexity classification problems for the state-of-the-art models.

In the MNIST, Fashion-MNIST, CIFAR-10, and CIFAR-100 datasets, the distribution skew discussed in Section V-C was applied. In this scenario, each node (e.g., device) involved in the training process within the FL architecture receives data allocations randomly.

This randomized allocation yields a non-uniform distribution, implying that class representation across nodes is unbalanced and not guaranteed to be equally represented or assigned to each node. This approach is intended to ensure that the differences observed between nodes arise from intrinsic randomness rather than systematic bias. This approach aims to replicate real-world data heterogeneity in IoT scenarios, allowing each node to contribute unevenly and thereby simulate realistic conditions.

TABLE 11. Datasets in FL architecture.

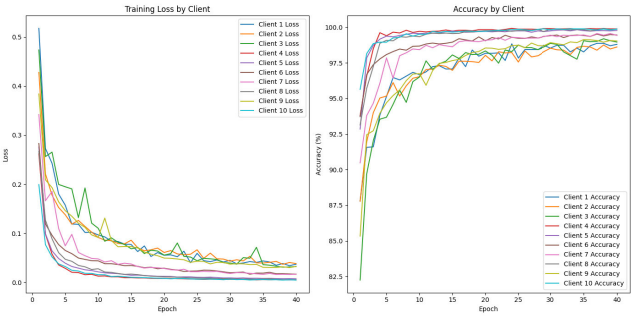
Dataset	Description	Classes	Distribution	Architectures
MNIST	Handwritten digits	10	non-iid	FL
Fashion-MNIST	Clothing items	10	non-iid	FL
CIFAR-10	Various objects	10	non-iid	FL
CIFAR-100	100 categories	100	non-iid	FL

A total of 240 experiments were conducted to identify the model's behavior in a non-iid environment, focusing on optimizing data quality and reducing computational costs.

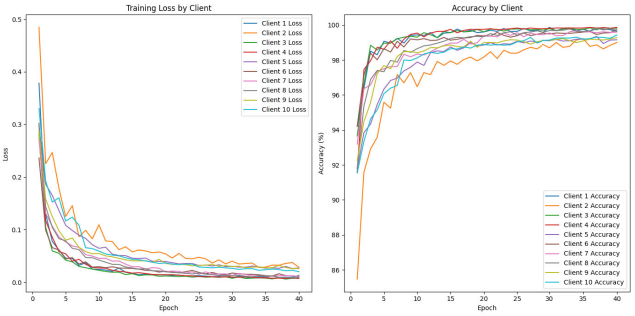
Figures 11 (a) and 11 (b) show the results of FL applied to the MNIST dataset, using the FedAvg and FedProx algorithms, respectively. Similarly, Figures 12 (a) and 12 (b) show the same algorithms applied to the Fashion-MNIST dataset. In addition, the results for the CIFAR-10 set are illustrated in Figures 13 (a) and 13 (b), while those for CIFAR-100 are represented in Figures 14 (a) and (b).

Table 12 presents the results obtained from 240 experiments, providing the average precision, recall, F1-score, accuracy, loss, and training time. These results provide insight into how entropy influences data quality and reduces computational and energy costs.

Half of the dataset was randomly selected using the Random method to ensure a fair comparison and follow the same methodology. As a result, this allowed for comparing training with a complete dataset and training using an entropy-based selection. All models in the table followed

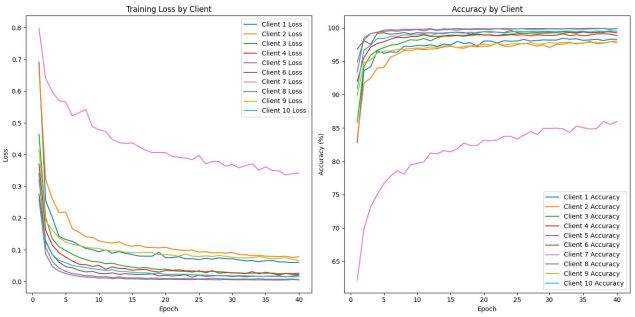


(a) FedAvg convergence.

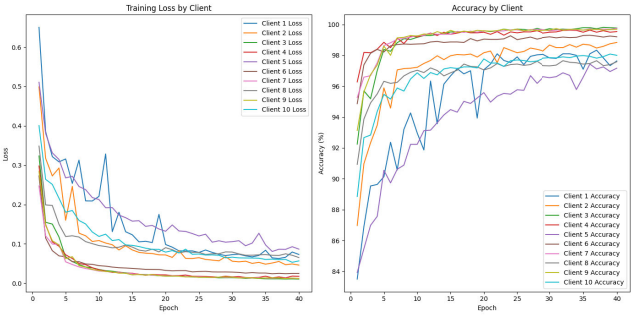


(b) FedProx convergence.

FIGURE 11. MNIST: EnBaSe (FedAvg & FedProx).



(a) FedAvg convergence.



(b) FedProx convergence.

FIGURE 12. Fashion-MNIST's: EnBaSe (FedAvg & FedProx).

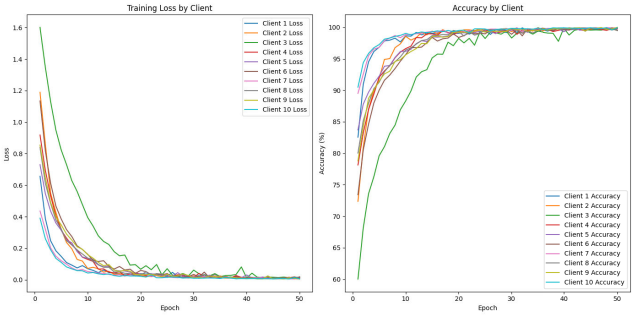
the same training pattern using 50 epochs with ten available nodes participating.

Consequently, it is possible to underscore the efficacy of the quality-based selection strategy using EnBaSe, which

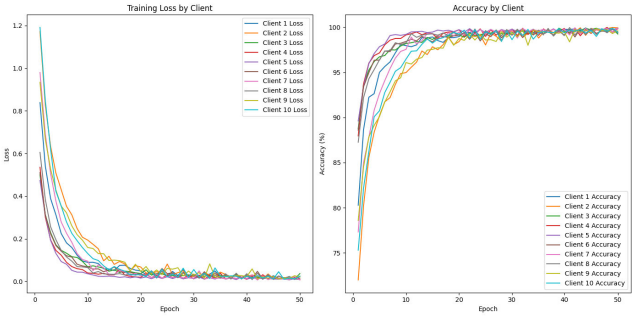
TABLE 12. Average experiment results with FedAvg and FedProx.

Dataset	Algorithm	Model Selection	Precision	Recall	F1-Score	Accuracy (%)	Loss	Time (s)
Mnist	FedAvg	All Data	≈74.30%	≈74.40%	≈71.90%	≈85.70%	≈0.253	≈1,096
Mnist	**FedAvg**	**EnBaSe**	≈66.10%	≈66.50%	≈62.60%	≈78.91%	≈0.193	≈586
Mnist	FedAvg	Random	≈42.50%	≈40.20%	≈34.60%	≈56.80%	≈0.248	≈555
Mnist	FedProx	All Data	≈69.26%	≈71.40%	≈67.48%	≈81.73%	≈0.179	≈1,180
Mnist	**FedProx**	**EnBaSe**	≈69.28%	≈68.93%	≈66.01%	≈81.26%	≈0.264	≈629
Mnist	FedProx	Random	≈42.55%	≈42.97%	≈37.81%	≈59.13%	≈0.264	≈606
Fashion	FedAvg	All Data	≈56.80%	≈52.70%	≈47.50%	≈62.21%	≈1.753	≈1,084
Fashion	**FedAvg**	**EnBaSe**	≈53.80%	≈49.70%	≈45.40%	≈58.90%	≈1.572	≈556
Fashion	FedAvg	Random	≈27.80%	≈25.50%	≈20.70%	≈39.90%	≈3.505	≈574
Fashion	FedProx	All Data	≈60.30%	≈57.10%	≈52.30%	≈66.60%	≈1.390	≈1,125
Fashion	**FedProx**	**EnBaSe**	≈55.70%	≈52.50%	≈47.60%	≈60.80%	≈1.709	≈601
Fashion	FedProx	Random	≈26.50%	≈24.80%	≈18.80%	≈35.90%	≈3.856	≈632
CIFAR-10	FedAvg	All Data	≈47.80%	≈41.50%	≈34.40%	≈43.12%	≈1.749	≈14,958
CIFAR-10	**FedAvg**	**EnBaSe**	≈43.90%	≈38.40%	≈32.70%	≈40.32%	≈1.889	≈8,259
CIFAR-10	FedAvg	Random	≈8.96%	≈9.97%	≈2.72%	≈10.01%	≈4.274	≈8,205
CIFAR-10	FedProx	All Data	≈46.12%	≈38.75%	≈32.33%	≈39.53%	≈2.009	≈15,662
CIFAR-10	**FedProx**	**EnBaSe**	≈43.69%	≈37.39%	≈32.12%	≈40.42%	≈1.821	≈8,413
CIFAR-10	FedProx	Random	≈8.79%	≈9.99%	≈3.06%	≈10.09%	≈4.776	≈8,394
CIFAR-100	FedAvg	All Data	≈47.84%	≈41.18%	≈34.85%	≈43.24%	≈1.715	≈15,214
CIFAR-100	**FedAvg**	**EnBaSe**	≈44.70%	≈39.75%	≈33.98%	≈43.57%	≈1.744	≈8,131
CIFAR-100	FedAvg	Random	≈8.31%	≈9.96%	≈2.27%	≈10.00%	≈5.092	≈8,274
CIFAR-100	FedProx	All Data	≈55.41%	≈55.84%	≈51.94%	≈63.08%	≈1.834	≈18,020
CIFAR-100	**FedProx**	**EnBaSe**	≈48.01%	≈46.41%	≈42.99%	≈53.54%	≈2.198	≈8,202
CIFAR-100	FedProx	Random	≈27.70%	≈35.71%	≈28.29%	≈41.20%	≈3.345	≈8,418

Note: The datasets were trained for 50 epochs with ten available clients (nodes).

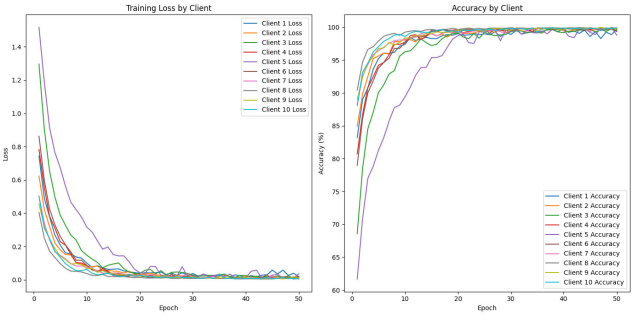


(a) FedAvg convergence.

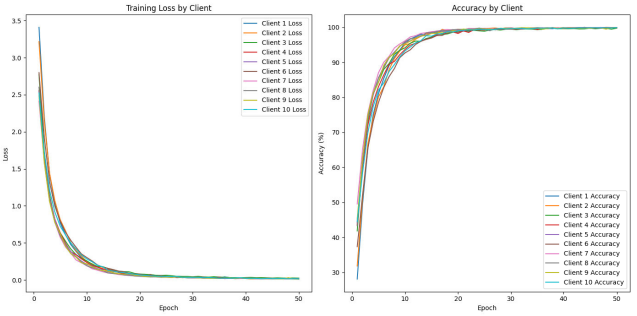


(b) FedProx convergence.

FIGURE 13. CIFAR-10: EnBaSe (FedAvg & FedProx).



(a) FedAvg convergence.



(b) FedProx convergence.

FIGURE 14. CIFAR-100: EnBaSe (FedAvg & FedProx).

demonstrates a significant reduction in training time compared to the model using the entire dataset. This observation was corroborated through the analysis of the training times of all the EnBaSe and Random datasets.

Additionally, a marginal decrease was observed in the overall accuracy of the EnBaSe model compared to the model trained on the entire dataset. Finally, in the global aggregation method, FedProx EnBaSe proved particularly effective over time, significantly reducing computation time while preserving accuracy within acceptable bounds.

The results presented in Table 12 compare EnBaSe and All Data using the same neural network architecture and hyperparameters. Although EnBaSe reduces the computational cost by approximately 50%, and the performance metrics are similar in magnitude, a slight degradation in precision and accuracy was observed across nearly all models.

This slight performance drop is evident in most datasets, where EnBaSe exhibits minor reductions in metrics such as accuracy, precision, recall, and F1-score compared to the entire dataset (e.g., All Data) as detailed in Table 12.

Moreover, Table 12 also indicates that, in some cases, the loss values are slightly higher than those observed with All Data, suggesting that EnBaSe has not yet achieved complete convergence. This observation highlights the need for further optimization of hyperparameters and the neural network architecture to enhance performance.

F. BENCHMARK: MULTIPLE CLIENTS AND HIGH LOAD

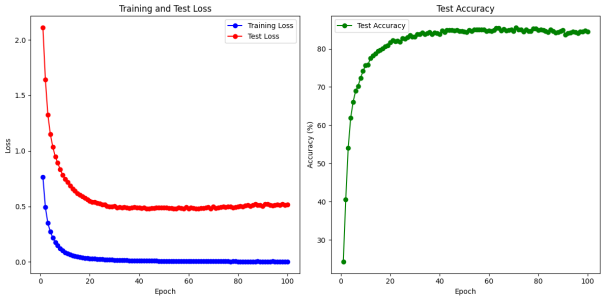
This section re-evaluates the experiment conducted under the non-iid scenario using the same criteria for the evaluation metrics specified in Section V-C. The neural networks used were the same as those introduced in Section V-D. The detailed experimental configurations are provided in Section V-B. To give more insight into the core hypothesis presented in the initial hypothesis of this work regarding the quality of local data at the edge EnBaSe, a benchmark experiment was conducted using FedProx, a model developed to address challenges in more realistic scenarios, particularly those with skewed and challenging distributions.

This test will be re-evaluated to gain a better understanding of how an edge-embedded algorithm, improving quality, interacts with a more sophisticated global aggregation technique. In other words, this study addresses both components of the FL process: the edge and the global neural network.

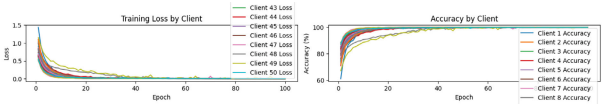
The same settings were maintained for the neural network's architecture and parameters to test the model's efficacy under challenging conditions. We utilized the following hardware: 83 GB of RAM, 40 GB GPU, and 201 GB storage. The number of clients and epochs was increased to 50 and 100, respectively. The analysis evaluated the algorithm's capacity using a large workload and data diversity.

This adjustment aims to replicate an advanced computational system for rigorous model analysis under high demand. In doing so, it is observed that with many clients (devices with datasets), the EnBaSe algorithm achieves smooth convergence.

The results presented in Figures 15 and 16 for the **CIFAR-10** and **CIFAR-100** benchmarks, respectively, show the performance of the **FedProx** model using **EnBaSe** over 100 epochs with 50 clients (nodes). For CIFAR-10, the model

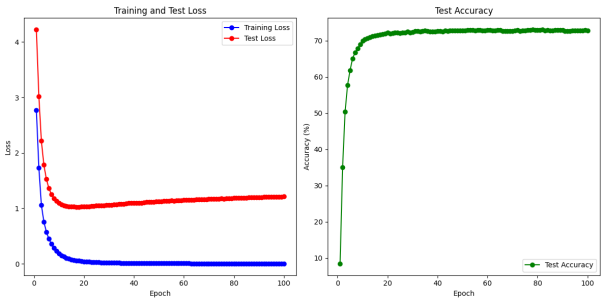


(a) Global model convergence over 100 epochs for the CIFAR-10.

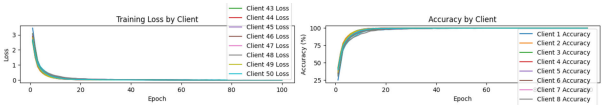


(b) The FedProx algorithm clients converged over 100 epochs for 50 clients at CIFAR-10

FIGURE 15. Global model and clients (nodes) convergence for 100 epochs and 50 clients at CIFAR-10 for the FedProx algorithm.



(a) Global model convergence over 100 epochs for CIFAR-100.



(b) FedProx algorithm clients convergence over 100 epochs for 50 clients at

FIGURE 16. Global model and clients (nodes) convergence for 100 epochs and 50 clients at CIFAR-100 for the FedProx algorithm.

achieved a precision of 82.20%, recall of 81.51%, F1-Score of 81.32%, and accuracy of 84.46%, with a loss value of 0.515, and a training time of 47,011.90 seconds. For CIFAR-100, the precision was 71.71%, recall 70.83%, F1-Score 70.69%, and accuracy 72.84%, with a loss value of 1.216 and a training time of 46,983.84 seconds.

Furthermore, the model reached the maximum neural network architecture maximum accuracy level much earlier than the total number of training epochs. The EnBaSe model resulted in a considerable reduction in processing time, approximately halving the computational cost. Therefore, this indicates that energy consumption and computational

expenses would likely increase if the experiment were conducted again without the proposed EnBaSe algorithm.

Table 13 presents the benchmark test results of EnBaSe, along with a comparison to other recent studies that use state-of-the-art methods to evaluate the effects of the EnBaSe algorithm on accuracy. The most representative results were compiled from the experiments reported in this article.

TABLE 13. Performance of models on CIFAR-10 and CIFAR-100 datasets in the non-iid scenario.

Dataset	Architecture	Author	Model	Acc (%)
CIFAR-10	CNN	[77]	Scaffold-Vanilla	26.65
	CNN	[77]	FedProx-Vanilla	27.42
	CNN	[77]	FedAvg-Vanilla	58.57
	CNN	[78]	AdaFedAdam	72.77
	CNN	[23]	FedCOME	75.88
	CNN	**Our Model**	**FedProx (EnBaSe)**	**84.46**
	CNN	[45]	FedPer++	85.09
	CNN	[44]	FedAvg (Adapted)	90.80
CIFAR-100	CNN	[77]	Scaffold-Vanilla	4.49
	CNN	[77]	FedProx-Vanilla	10.39
	CNN	[19]	FedHKD	29.88
	CNN	[79]	FedCOME	37.66
	CNN	[60]	Fed-IT	39.29
	CNN	[77]	FedAvg-Vanilla	40.36
	CNN	[80]	FedProx(FedFed)	70.02
	CNN	**Our Model**	**FedProx (EnBaSe)**	**72.84**

This study compares the results of the proposed algorithm, obtained in Section VI-E (Table 12), where FL was applied, with the benchmark results presented in this section (Table 13).

This comparison considered the increase in the number of connected devices from 10 to 50, as well as the increase in the number of training epochs from 50 to 100, representing a higher workload and an additional challenge for the algorithm (EnBaSe). The analysis focuses on the two most challenging datasets, CIFAR-10 and CIFAR-100, which offer a reliable basis for comparative evaluation.

It was observed that, in the case of CIFAR-10 and CIFAR-100, the precision values for FedProx EnBaSe improved from 40.42% and 53.54% to 84.46% and 72.84%, respectively. These results indicate that the algorithm exhibits improved performance as more devices are connected under increasingly complex non-iid conditions.

Finally, Table 13 presents a comparative analysis of the results obtained in comparison to the state-of-the-art. The most challenging datasets and the results obtained by different methods were considered. In the context of the CIFAR-10 dataset, in a non-iid scenario, the EnBaSe model achieved an accuracy of 84.46%, a performance comparable to that of FedPer++ (85.09%) and superior to that of FedCOME (75.88%).

In the case of the CIFAR-100 dataset, EnBaSe presented an accuracy of 72.84%, which yields a higher accuracy than comparable methods, such as FedCOME (37.66%) and Fed-IT (39.29%). These findings indicate the effectiveness of EnBaSe in optimizing data selection, reducing computational costs while maintaining comparable levels of accuracy of the models.

VII. DISCUSSION

The hypothesis of this study posits that entropy can be used to measure data quality, considering that it quantifies the uncertainty or unpredictability in each node. High entropy values are associated with data that is highly unpredictable or varies within a node. As a result, each dataset in a node provides significantly different information, reducing the predictability of samples within the node based on previous information.

Accordingly, the hypothesis was empirically evaluated using the low entropy in each node, which was measured as a separation criterion to avoid asymmetry. For clarification purposes, this study analyzes hypotheses regarding data quality in nodes and the reduction of computational costs in the field of CV in collaboration with a laboratory involved in a project under contractual confidentiality, targeting real-time monitoring systems devices that complement data related to biosignals, involving resource-constrained embedded systems. In this context, the study, related to the field of CV, conducted initial experiments using a CNN due to its lower computational complexity and cost efficiency.

A systematic literature search was conducted using keywords in conjunction with the names of the MNIST, Fashion-MNIST, CIFAR-10, and CIFAR-100 datasets, including time benchmark, training benchmark, training time comparison, training time performance, optimization training time, throughput training time, throughput training performance, and including multiple logical combinations of these terms. Despite this effort, the literature lacks comprehensive analyses of simulations that vary the implemented hardware, training times, or the number of epochs. Therefore, we compared the latest metrics from the most recent state-of-the-art models with those in this study.

One of the complex challenges identified in the literature during the development of this study was the need for more standardization of the experiments presented. Many studies report incomplete experimental details, often omitting essential metrics such as validation, F1-score, recall, loss values, or the model's training time. Finally, a limitation of the present study was the funding for the experiments, which necessitated reducing the number of epochs to 50 and the number of nodes to 10 due to the extensive volume of experiments conducted, as stated in Subsection VI-E.

Recent state-of-the-art approaches in FL architecture and centralized architectures predominantly focus on enhancing data homogeneity and addressing the heterogeneity and challenges associated with non-iid distributions.

However, such approaches often do not adequately address the limitations imposed regarding the computational capacity of devices and the resources required for neural networks to achieve a high generalization capability.

This issue is critical for fostering more significant equity in the integrability and applicability of systems designed for architectures with low computational and energy capacities, which is the primary focus of this study.

Thus, the EnBaSe algorithm demonstrates efficiency in optimizing the computational resources allocated to the neural network, such as the bandwidth used for global model updates in FL or data transfer in a centralized architecture, yielding a reduction of approximately 50% in communication.

This optimization also reduces the time required for training the neural network and, correspondingly, decreases the energy consumption, as models utilizing the EnBaSe algorithm complete training approximately 50% more rapidly.

To clarify the relationship between energy and execution time, energy is directly related to the Consumption of Power from devices and Execution time based on the following formula:

$$E = P(Whatts) \times Execution\_time(s)$$

where E represents the energy consumed, P is the device's power in Watts, and *Execution_time* in seconds is the duration of use. Thus, a decrease in execution time directly impacts battery savings for IoT devices in the same percentage ratio as the reduction in execution time. This is an essential benefit of the EnBaSe algorithm strategy.

Moreover, this advancement is particularly relevant for systems with energy and computational constraints, such as resource-constrained smart devices, including drones or AI-powered medical equipment with low processing capabilities. The proposed approach applies to diverse real-world computational systems.

As a result, the neural network applied to the dataset, referred to as "All Data," encompasses the entirety of the available data, whereas EnBaSe acts as a structured data selection framework that effectively represents the representative abstraction of the information landscape.

Finally, the analyses conducted in this study demonstrated the feasibility of using Shannon's entropy formula as a method for estimating frequency and quantifying visual complexity, identifying complex and direct samples by recognizing the stochastic variance of the pixel distribution, thus selecting the sample with the best data quality and, at the same time, reducing computational and energy processing costs.

VIII. CONCLUSION

This study underscores the importance of assessing the quality of input data. In this context, entropy experiments demonstrated that it retains probabilistic relevance even after linear transformations. Moreover, because entropy preserves consistent probabilistic values for the data and remains unaffected by linear geometric transformations, entropy can be effectively utilized for data selection before applying DA techniques and normalization, thereby optimizing resource utilization. This approach is particularly applicable to ML methods, which are crucial in the context of the IoT, where the processing capacity of devices is often limited.

The primary method used in this study was the implementation of the EnBaSe algorithm, which selects data based on its entropy to improve data quality in both iid and non-iid scenarios. The algorithm is designed to reduce computational costs while maintaining performance within operational thresholds. It plays a critical role in FL for IoT devices, where computational resources are limited, and non-iid mitigation is essential.

Additionally, the consistency of the results demonstrates that this approach has significant applications in real-world systems, where controlling data volume, data quality, and computational resources is critical. Examples include CV, image processing, traffic monitoring, automated inspection, digital health, diagnostics, and more.

The main contributions of this research are as follows:

- A detailed study on the behavior of entropy in images and its distribution in CV;
- Analysis of the impact of linear transformations and normalization on data entropy;
- Reducing the computational cost of IoT edge devices;
- The presentation of detailed metrics such as accuracy, F1-score, recall, loss values, and model training time;
- Structuring performance metrics for reproducibility and benchmarking for future experiments;
- The comparison of accuracy in iid and non-iid scenarios with other experiments;
- A comprehensive literature review on data quality and FL; and
- The development of the EnBaSe algorithm, which efficiently selects high-quality data based on entropy analysis. This method reduces unnecessary computational processing, optimizes model convergence, and improves data selection for FL and centralized learning scenarios, particularly in resource-constrained environments such as IoT.

Currently, global aggregation models for FL are being developed to comprehensively address the challenges inherent in weight aggregation, including data distribution, customer selection, heterogeneity, and temporal and spatial dependencies, especially in scenarios characterized by non-iid samples. However, these experiments demonstrate the feasibility of using algorithms in edge devices to autonomously address data distribution challenges, offering a simpler alternative to existing methods.

Finally, one of the most important contributions of this study is an embedded algorithm that adaptively operates on edge IoT devices based on their distribution and is compatible with various aggregation models in FL.

In conclusion, the algorithm selects subsets of each class with lower entropy, aiming to enhance the system's predictive capacity, as discussed in Subsection II-C and Section IV. Thus, the inherent heterogeneity of each device is reduced according to its dataset, handling data asymmetry (e.g., extreme data, outliers) according to the distribution of each class, as specified in Algorithm IV-C. In this

process, the resulting sample distribution approximates a Gaussian, and the equivalent part with lower entropy is selected, as demonstrated in Subsection VI-B. Additionally, subsequent investigations will be carried out to analyze the statistical properties of extreme-value data when analyzed using various analytical techniques.

This section presents the main conclusions of this study. At the same time, Section IX explores the future directions for developing this research, discussing the key areas that warrant further investigation.

IX. FUTURE WORKS AND DIRECTIONS

Subsequent research efforts will focus on extending the algorithm EnBaSe to additional FL use cases and data modalities, as well as integrating it with other optimization techniques. In summary, future research should expand the algorithm's applicability to different data types and FL scenarios.

Furthermore, future research will address the convergence of global models aimed to preventing excessive training and minimizing the associated computing costs and energy consumption.

In future work, we intend to expand the experiments to create more significant variability in non-iid and iid scenarios, for example, by utilizing the Dirichlet distribution. Additionally, we aim to increase comparisons by employing other methods, such as FedDyn, FedDF, Scaffold, and FedLAW. Thus, evaluating EnBaSe using aggregation methods that adopt different strategies allows us to identify the contexts in which performance remains more robust and where potential improvements are possible. This provides a deeper understanding of its behavior in diverse scenarios and solidifies its strengths and limitations for future advancements. Additionally, we will assess the individual impact of EnBaSe on the energy consumption of IoT devices.

This simulated experiment indicated a strong trend towards the feasibility of adoption in the current AI algorithms. Since the experiments with EnBaSe demonstrated its initial feasibility, future work will conduct experiments under real-world operational conditions. Furthermore, we will conduct FL experiments on real-world IoT devices to capture fine-grained adjustments in our algorithm.

Finally, for future work, we intend to revisit the state-of-the-art and explore other application domains of artificial intelligence for FL algorithms. Additionally, we aim to explore new metrics that complement or extend existing ones used in the state-of-the-art.

APPENDIX A
STATE-OF-THE-ART APPROACHES AND METHODS

This section provides a summary of the state of the art, based on a systematic review of the literature. The table outlines the main approaches adopted and the scenarios in which the models have been applied, as well as the scenarios addressed in the respective studies by the authors.

Additionally, the main datasets and evaluation metrics were emphasized, as well as the metrics adopted, and the advantages and disadvantages pointed out by the authors in their respective works were systematically identified. In cases where this information was not explicitly mentioned, potential impacts were inferred based on available data when explicit mentions were absent, using the available evidence.

As we can see from the state-of-the-art analysis in Table 14, common approaches that employ customer aggregation, selection, sampling, or measurement techniques for customer contribution are commonly used. This strategy represents a predominant approach to addressing device heterogeneity. Many of these methods are adaptive, seeking to extract characteristics to train neural networks and address the challenges inherent to heterogeneous data distributions.

In contrast, a second approach, which has gained increasing attention in recent studies, is based on Information Theory. This research direction focuses on quantifying information gain and system homogeneity based on entropy, as discussed by [18], [25], [35], [39], [47], [59], and [60].

The main application scenarios in the literature predominantly involve the IoT or FL context, i.e., environments characterized by multiple connected devices and distributed data. While some authors focus their studies on scalability issues in multi-client systems, communication efficiency, or addressing known limitations in existing approaches, others concentrate on specific application areas, such as medical or noisy data.

The primary datasets used are those widely known in the literature, such as MNIST, Fashion-MNIST, CIFAR-10, and CIFAR-100. These datasets represent different levels of difficulty and complexity in problems related to deep neural networks, with CIFAR-100 being widely regarded as a high-complexity benchmark in the field.

However, alternative datasets are also employed for domain-specific analyses, targeted at specific applications, or designed for particular challenges. For example, in [47], a dataset of crop pest images was utilized, whereas in [35], protein-related datasets were analyzed.

The primary metrics adopted often include accuracy, communication cost, efficiency, precision, recall, F1-Score, and Mean Absolute Error (MAE). In addition, some authors use metrics such as communication overhead per training round, TPR (True Positive Rate), TNR (True Negative Rate), entropy index, and Standard Deviation.

The primary focus and advantages discussed in the papers relate to developing more adaptive systems capable of addressing convergence stability, model scalability, and efficiency, as well as the difficulties associated with training neural networks in non-iid scenarios. These aspects are frequently identified as core challenges faced by FL in the literature.

Despite these advances, several approaches continue to face limitations in achieving in achieving global model convergence. These difficulties include loss of accuracy at the end of training, instability at critical moments of convergence,

TABLE 14. State-of-the-art approaches, metrics, and scenarios.

Author	Date	Approach Used	Application Scenario	Datasets Used	Metrics Adopted	Advantages	Disadvantages
Itahara, Sohei, et al. [18]	2021	Aggregation with entropy reduction	IoT, FL and non-iid	MNIST, Fashion-MNIST, IMDb, Reuters	Accuracy, Communication Cost	Robustness against attacks and noise	Loss of Accuracy
Criado, Marcos F, et al. [10]	2022	Continual Learning	IoT, FL and non-iid	MNIST, SVHN, USPS, Office-31, Bing-Caltech256, COREL5000	Accuracy	Adapts to data distribution changes	Labeled data assumptions
Al-Saedi, et al. [38]	2022	Round-wise Clustering	Communication Efficiency	MNIST, Fashion MNIST, Fashion-MNIST, CIFAR-10	Accuracy	Reduces Communication, dynamic partitioning	Complex dynamic clusters
Yu, Xi, et al. [20]	2022	Dynamic Regularization	Multiple Clients	MNIST, Fashion-MNIST	Accuracy, Precision, Recall, AUC	Adaptive learning, resource-efficient	Instability near convergence
Ullah, Shan, et al. [44]	2022	Local Parameter Optimization	Image Classification	CIFAR-10	Accuracy	Improves efficiency and performance	Dependency on FedAVG
Xu, Jian, et al. [45]	2022	Custom Classifier	Heterogeneous Data	Fashion-MNIST, CIFAR-10	Accuracy	Handles data variations, reduces Communication	Risk of overfitting with insufficient local data
Li, Yang, et al. [47]	2022	Disturbed entropy	Agricultural pest recognition	Agricultural pest images	Accuracy	Reduces redundancy in datasets	Restricted to multi-class classification
Wolfrath, Joel, et al. [46]	2022	Client Clustering	Mobile, IoT Devices	Fashion-MNIST, CIFAR-10	Time-to-accuracy, model accuracy.	Faster convergence, efficient training	Dependence on stable distributions
Tu, Chengwu et al. [21]	2023	Node selection	IoT, FL and non-iid	MNIST, CIFAR-10	Accuracy, Communication Rounds	Fewer rounds, Global Accuracy	Poorly representative data
Li, Boyuan et al. [50]	2023	Knowledge-based dynamics	IoT, FL and non-iid	MNIST, Fashion-MNIST, CIFAR-10	Convergence, Accuracy	Better convergence and Accuracy	Complexity with diverse data
Yang, Wei-Jong et al. [51]	2023	Dynamic weights	IoT, FL and non-iid	MNIST, Fashion-MNIST, CIFAR-10	Global Accuracy, Communication Rounds	Less communication, dynamic adaptation	Additional processing for weights
Chen, Huancheng et al. [19]	2023	Feature extraction	IoT, FL and non-iid	CIFAR-10, CIFAR-100, SVHN	Local Accuracy, global Accuracy	Robustness for heterogeneous data	Dependency on hyper-knowledge
Zheng, Shu et al. [23]	2023	Client sampling	IoT, FL and non-iid	MNIST, Fashion-MNIST, CIFAR-10, CIFAR-100	Accuracy	Reflects global distribution	Quadratic computational complexity
Huang, Chenxi et al. [52]	2023	Global memory vectors	Heterogeneous devices	CIFAR-100	Weighted Accuracy, rounds-Accuracy	Reduces variance	Initial setup computations
Orlandi, Fernanda C. et al. [39]	2023	Entropy for Non-IID Data Mitigation	IoT Devices	MNIST, CIFAR-10	Accuracy, execution time	Mitigates impacts of non-IID data, lower power consumption	slight reduction in accuracy
Qiao, Yu et al. [54]	2023	Prototype regularization	Image classification	MNIST, Fashion-MNIST	Average Accuracy, Communication Efficiency	Fast convergence in non-iid scenarios	Prototype computation per round
Wu, Chenrui et al. [56]	2023	Prototypes and dynamic pseudo-labeling	Noisy data, imbalanced classes	CIFAR-10, CIFAR-100, Clothing1M	Accuracy, Precision, Recall	Stabilizes accurate pseudo-labeling	Computational cost
Sun, Qiheng et al. [24]	2023	Client contributions	Malicious clients, healthcare	CIFAR-10, Fashion-MNIST, Fed-ISIC2019	Accuracy, Convergence Rate	Robustness against poisoning attacks	Intensive computations, high computational cost
Milan Ilić et al. [58]	2023	Model updates	Medicine, IoT, FL and non-iid	Fashion-MNIST, LEAF, Adult Income, Body Signal of Smoking	F1-Score, Mean Absolute Error (MAE)	Flexible across multiple domains/tasks	Competing strategies with inferior performance
Condori Bustincio, et al. [25]	2023	Adaptive selection by entropy	Heterogeneity, communication overhead	CIFAR-10	Accuracy, Communication Cost	Reduces communication overhead	Limited generalization on datasets
Zhang, Yu, et al. [35]	2023	Differential evolution by entropy	Protein structure prediction	25PDB, FC699, D1189, D640	Accuracy, TPR, TNR, F1-Score	Robustness in feature selection	High complexity, requires fine-tuning
Hamidi, Shayan Mohajer, et al. [60]	2024	Entropy in loss function	Medical diagnosis	CIFAR-10, CIFAR-100, TinyImageNet	Accuracy, Standard Deviation	Better Accuracy on unbalanced datasets	Increased complexity
Yan, Litao, et al. [59]	2024	Entropy production model	Wireless communication	Physical and mathematical simulations	Entropy rate, Communication Cost	Optimized allocation, parallel processing	Complexity in modeling

and disadvantages associated with adding complexity to architectures. Furthermore, this generally results in higher computational costs and problems related to increased modeling complexity.

A. DISCUSSION

The analysis indicates that some articles address class balancing or heterogeneity, while relatively few studies address fairness and bias mitigation in FL or bias mitigation. In addition, most studies concentrate on communication efficiency and data heterogeneity, addressing FL by discussing the importance of privacy and the relevance of IoT devices.

Although some authors have explored strategies to reduce communication costs and improve convergence, energy efficiency, and computational performance gains, as well as the generalization of neural networks, a lack of standardized evaluation metrics is evident in current FL literature experiments. Standardized metrics that consider the distribution of results and computational efficiency are particularly relevant in scenarios with IoT applications, where energy efficiency is as crucial as accuracy, particularly for resource-constrained edge devices.

Most studies mention FL training in IoT systems (e.g., smart cities, medical applications, industrial applications,

etc.). However, these topics remain underexplored, particularly in scenarios where prediction needs to occur in real-time, such as continuous data flows. In addition to the adaptive models often used, it would be interesting to consider approaches based on online learning in the context of FL. These systems would need to adapt dynamically and make inferences with low latency, a requirement that is critical in innovative city environments.

Another recurring observation in the literature is that most domains of interest focus on image classification and analysis problems. However, other essential domains, such as natural language processing, audio and video systems, and biosignal analysis, require further investigation.

In summary, the findings reveal that many aspects analyzed in the current state of the art address core concerns, such as privacy preservation and data security, as well as the use of advanced technologies for data analysis. These studies contribute to a comprehensive understanding of the current limitations in the literature, enabling the identification of open challenges and future research directions that remain unaddressed.

REFERENCES

[1] J. Verbraeken, M. Wolting, J. Katzy, J. Kloppenburg, T. Verbelen, and J. S. Rellermeyer, "A survey on distributed machine learning," *ACM Comput. Surv.*, vol. 53, no. 2, pp. 1–33, Mar. 2021, doi: 10.1145/3377454.

[2] L. Caruccio, D. Desiato, G. Polese, and G. Tortora, "GDPR compliant information confidentiality preservation in big data processing," *IEEE Access*, vol. 8, pp. 205034–205050, 2020, doi: 10.1109/ACCESS.2020.3036916.

[3] J. Shehu Yalli, M. Hilmi Hasan, and A. Abubakar Badawi, "Internet of Things (IoT): Origins, embedded technologies, smart applications, and its growth in the last decade," *IEEE Access*, vol. 12, pp. 91357–91382, 2024, doi: 10.1109/ACCESS.2024.3418995.

[4] W. A. Al-Nbhany, A. T. Zahary, and A. A. Al-Shargabi, "Blockchain-IoT healthcare applications and trends: A review," *IEEE Access*, vol. 12, pp. 4178–4212, 2024, doi: 10.1109/ACCESS.2023.3349187.

[5] J. B. Minani, F. Sabir, N. Moha, and Y.-G. Guéhéneuc, "A systematic review of IoT systems testing: Objectives, approaches, tools, and challenges," *IEEE Trans. Softw. Eng.*, vol. 50, no. 4, pp. 785–815, Apr. 2024, doi: 10.1109/TSE.2024.3363611.

[6] S. Chintala, "Iot and ai synergy: Remote patient monitoring for improved healthcare," in *Proc. 4th Int. Conf. Innov. Practices Technol. Manage. (ICIPTM)*, Feb. 2024, pp. 1–6, doi: 10.1109/iciptm59628.2024.10563530.

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

[8] H. Sun, S. Li, F. R. Yu, Q. Qi, J. Wang, and J. Liao, "Toward communication-efficient federated learning in the Internet of Things with edge computing," *IEEE Internet Things J.*, vol. 7, no. 11, pp. 11053–11067, Nov. 2020, doi: 10.1109/JIOT.2020.2994596.

[9] L. U. Khan, W. Saad, Z. Han, E. Hossain, and C. S. Hong, "Federated learning for Internet of Things: Recent advances, taxonomy, and open challenges," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1759–1799, 3rd Quart., 2021, doi: 10.1109/COMST.2021.3090430.

[10] M. F. Criado, F. E. Casado, R. Iglesias, C. V. Regueiro, and S. Barro, "Non-IID data and continual learning processes in federated learning: A long road ahead," *Inf. Fusion*, vol. 88, pp. 263–280, Dec. 2022, doi: 10.1016/j.inffus.2022.07.024.

[11] 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, Apr. 2021, doi: 10.1109/JIOT.2020.3030072.

[12] 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 Gener. Comput. Syst.*, vol. 135, pp. 244–258, Oct. 2022, doi: 10.1016/j.future.2022.05.003.

[13] P. R. R. De Souza, K. J. Matteussi, A. D. S. Veith, B. F. Zanchetta, V. R. Q. Leithardt, A. L. Murciego, E. P. De Freitas, J. C. S. Dos Anjos, and C. F. R. Geyer, "Boosting big data streaming applications in clouds with BurstFlow," *IEEE Access*, vol. 8, pp. 219124–219136, 2020, doi: 10.1109/ACCESS.2020.3042739.

[14] 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 Syst. Appl.*, vol. 250, Sep. 2024, Art. no. 123947, doi: 10.1016/j.eswa.2024.123947.

[15] D. Rosendo, A. Costan, P. Valduriez, and G. Antoniu, "Distributed intelligence on the edge-to-cloud continuum: A systematic literature review," *J. Parallel Distrib. Comput.*, vol. 166, pp. 71–94, Aug. 2022, doi: 10.1016/j.jpdc.2022.04.004.

[16] J. Kang, Z. Xiong, D. Niyato, Y. Zou, Y. Zhang, and M. Guizani, "Reliable federated learning for mobile networks," *IEEE Wireless Commun.*, vol. 27, no. 2, pp. 72–80, Apr. 2020, doi: 10.1109/MWC.001.1900119.

[17] A. Imteaj, U. Thakker, S. Wang, J. Li, and M. H. Amini, "A survey on federated learning for resource-constrained IoT devices," *IEEE Internet Things J.*, vol. 9, no. 1, pp. 1–24, Jan. 2022, doi: 10.1109/JIOT.2021.3095077.

[18] 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 Trans. Mobile Comput.*, vol. 22, no. 1, pp. 191–205, Jan. 2023, doi: 10.1109/TMC.2021.3070013.

[19] H. Chen, Johnny, Wang, and H. Vikalo, "The best of both worlds: Accurate global and personalized models through federated learning with data-free hyper-knowledge distillation," 2023, *arXiv:2301.08968*.

[20] X. Yu, L. Li, X. He, S. Chen, and L. Jiang, "Federated learning optimization algorithm for automatic weight optimal," *Comput. Intell. Neurosci.*, vol. 2022, pp. 1–19, Nov. 2022, doi: 10.1155/2022/8342638.

[21] C. Tu, S. Zhao, and H. Deng, "FedWNS: Data distribution-wise node selection in federated learning via reinforcement learning," in *Proc. 26th Int. Conf. Comput. Supported Cooperat. Work Design (CSCWD)*, May 2023, pp. 600–605, doi: 10.1109/CSCWD57460.2023.10152675.

[22] Z. Li, T. Lin, X. Shang, and C. Wu, "Revisiting weighted aggregation in federated learning with neural networks," in *Proc. Int. Conf. Mach. Learn.*, Jan. 2023, pp. 19767–19788. [Online]. Available: https://proceedings.mlr.press/v202/li23s.html

[23] S. Zheng, T. Ye, X. Li, and M. Gao, "Federated learning via consensus mechanism on heterogeneous data: A new perspective on convergence," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2024, pp. 7595–7599, doi: 10.1109/ICASSP48485.2024.10446892.

[24] 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 *Proc. 29th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, Aug. 2023, pp. 2096–2108, doi: 10.1145/3580305.3599500.

[25] R. W. Condori Bustincio, A. M. de Souza, J. B. D. Da Costa, and L. Bittencourt, "EntropicFL: Efficient federated learning via data entropy and model divergence," in *Proc. IEEE/ACM 16th Int. Conf. Utility Cloud Comput.*, Dec. 2023, pp. 1–6, doi: 10.1145/3603166.3632611.

[26] Z. Zhang, Y. Zhang, D. Guo, L. Yao, and Z. Li, "SecFedNIDS: Robust defense for poisoning attack against federated learning-based network intrusion detection system," *Future Gener. Comput. Syst.*, vol. 134, pp. 154–169, Sep. 2022.

[27] N. Rodríguez-Barroso, E. Martínez-Cámara, M. V. Luzón, and F. Herrera, "Backdoor attacks-resilient aggregation based on robust filtering of outliers in federated learning for image classification," *Knowledge-Based Syst.*, vol. 245, Jun. 2022, Art. no. 108588.

[28] A. Al-Dulaimy, M. Jansen, B. Johansson, A. Trivedi, A. Iosup, M. Ashjaei, A. Galletta, D. Kimovski, R. Prodan, K. Tserpes, G. Kousiouris, C. Giannakos, I. Brandic, N. Ali, A. B. Bondi, and A. V. Papadopoulos, "The computing continuum: From IoT to the cloud," *Internet Things*, vol. 27, Jul. 2024, Art. no. 101272.

[29] A. Ullah, T. Kiss, J. Kovács, F. Tusa, J. Deslauriers, H. Dagdeviren, R. Arjun, and H. Hamzeh, "Orchestration in the cloud-to-things compute continuum: Taxonomy, survey and future directions," *J. Cloud Comput.*, vol. 12, no. 1, pp. 1–29, Sep. 2023.

[30] A. Margara, G. Cugola, N. Felicioni, and S. Cilloni, "A model and survey of distributed data-intensive systems," *ACM Comput. Surv.*, vol. 56, no. 1, pp. 1–69, Jan. 2024, doi: 10.1145/3604801.

[31] Z. Lu, H. Pan, Y. Dai, X. Si, and Y. Zhang, "Federated learning with non-IID data: A survey," *IEEE Internet Things J.*, vol. 11, no. 11, pp. 19188–19209, Jun. 2024, doi: 10.1109/JIOT.2024.3376548.

[32] J. Coutinho-Almeida, R. J. Cruz-Correia, and P. P. Rodrigues, "Evaluating distributed-learning on real-world obstetrics data: Comparing distributed, centralized and local models," *Sci. Rep.*, vol. 14, no. 1, p. 11128, May 2024, doi: 10.1038/s41598-024-61371-1.

[33] B. Rao, J. Zhang, D. Wu, C. Zhu, X. Sun, and B. Chen, "Privacy inference attack and defense in centralized and federated learning: A comprehensive survey," *IEEE Trans. Artif. Intell.*, vol. 6, no. 2, pp. 333–353, Feb. 2025, doi: 10.1109/TAI.2024.3363670.

[34] L. Yuan, Z. Wang, L. Sun, P. S. Yu, and C. G. Brinton, "Decentralized federated learning: A survey and perspective," *IEEE Internet Things J.*, vol. 11, no. 21, pp. 34617–34638, Nov. 2024, doi: 10.1109/JIOT.2024.3407584.

[35] 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," *Appl. Soft Comput.*, vol. 136, Mar. 2023, Art. no. 110064, doi: 10.1016/j.asoc.2023.110064.

[36] S. Zhuge, Z. Zhou, W. Zhou, J. Wu, M. Deng, and M. Dai, "Noisy image segmentation utilizing entropy-adaptive fractional differential-driven active contours," *Multimedia Tools Appl.*, vol. 83, pp. 1–26, Sep. 2024, doi: 10.1007/s11042-024-20058-5.

[37] M. Zarif Hossain and A. Imteaj, "FedAVO: Improving communication efficiency in federated learning with African vultures optimizer," 2023, *arXiv:2305.01154*.

[38] A. A. Al-Saedi, V. Boeva, and E. Casalicchio, "FedCO: Communication-efficient federated learning via clustering optimization," *Future Internet*, vol. 14, no. 12, p. 377, Dec. 2022, doi: 10.3390/fi14120377.

[39] F. C. Orlandi, J. C. S. Dos Anjos, V. R. Q. Leithardt, J. F. De Paz Santana, 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, 2023, doi: 10.1109/ACCESS.2023.3298704.

[40] Z. Tao, J. Wu, and Q. Li, "Preconditioned federated learning," 2023, *arXiv:2309.11378*.

[41] B. Li, Y. Wu, J. Song, R. Lu, T. Li, and L. Zhao, "DeepFed: Federated deep learning for intrusion detection in industrial cyber–physical systems," *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5615–5624, Aug. 2021, doi: 10.1109/TII.2020.3023430.

[42] Z. Du, C. Wu, T. Yoshinaga, K.-L.-A. Yau, Y. Ji, and J. Li, "Federated learning for vehicular Internet of Things: Recent advances and open issues," *IEEE Open J. Comput. Soc.*, vol. 1, pp. 45–61, 2020, doi: 10.1109/OJCS.2020.2992630.

[43] 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, May 2022, doi: 10.1109/MSP.2021.3125282.

[44] S. Ullah and D.-H. Kim, "Federated learning convergence on IID features via optimized local model parameters," in *Proc. IEEE Int. Conf. Big Data Smart Comput. (BigComp)*, Jan. 2022, pp. 92–95, doi: 10.1109/BigComp54360.2022.00028.

[45] J. Xu, Y. Yan, and S.-L. Huang, "FedPer++: Toward improved personalized federated learning on heterogeneous and imbalanced data," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2022, pp. 1–8, doi: 10.1109/IJCNN55064.2022.9892585.

[46] J. Wolfrath, N. Sreekumar, D. Kumar, Y. Wang, and A. Chandra, "HACCS: Heterogeneity-aware clustered client selection for accelerated federated learning," in *Proc. IEEE Int. Parallel Distrib. Process. Symp. (IPDPS)*, May 2022, pp. 985–995, doi: 10.1109/IPDPS53621.2022.00100.

[47] Y. Li, X. Chao, and S. Ercisli, "Disturbed-entropy: A simple data quality assessment approach," *ICT Exp.*, vol. 8, no. 3, pp. 309–312, Sep. 2022, doi: 10.1016/j.icte.2022.01.006.

[48] S. K. Lo, Y. Liu, Q. Lu, C. Wang, X. Xu, H.-Y. Paik, and L. Zhu, "Toward trustworthy AI: Blockchain-based architecture design for accountability and fairness of federated learning systems," *IEEE Internet Things J.*, vol. 10, no. 4, pp. 3276–3284, Feb. 2023, doi: 10.1109/JIOT.2022.3144450.

[49] H. Lee, "Towards convergence in federated learning via non-IID analysis in a distributed solar energy grid," *Electronics*, vol. 12, no. 7, p. 1580, Mar. 2023, doi: 10.3390/electronics12071580.

[50] B. Li, S. Chen, and K. Yu, "FeDDkw – federated learning with dynamic Kullback–Leibler-divergence weight," *ACM Trans. Asian Low-Resource Lang. Inf. Process.*, vol. 22, pp. 1–19, Apr. 2023, doi: 10.1145/3594779.

[51] W.-J. Yang and P.-C. Chung, "Significant weighted aggregation method for federated learning in non-iid environment," in *Proc. 6th Int. Symp. Comput., Consum. Control (IS3C)*, Jun. 2023, pp. 330–333, doi: 10.1109/is3c57901.2023.00095.

[52] C. Huang, L. Xie, Y. Yang, W. Wang, B. Lin, and D. Cai, "Neural collapse inspired federated learning with non-iid data," 2023, *arXiv:2303.16066*.

[53] K. M. M. Dolaat, A. Erbad, and M. Ibrar, "Enhancing global model accuracy: Federated learning for imbalanced medical image datasets," in *Proc. Int. Symp. Netw., Comput. Commun. (ISNCC)*, Oct. 2023, pp. 1–4, doi: 10.1109/ISNCC58260.2023.10323682.

[54] Y. Qiao, H. Q. Le, and C. Seon Hong, "Boosting federated learning convergence with prototype regularization," 2023, *arXiv:2307.10575*.

[55] F. Sabah, Y. Chen, Z. Yang, M. Azam, N. Ahmad, and R. Sarwar, "Model optimization techniques in personalized federated learning: A survey," *Expert Syst. Appl.*, vol. 243, Jun. 2024, Art. no. 122874, doi: 10.1016/j.eswa.2023.122874.

[56] C. Wu, Z. Li, F. Wang, and C. Wu, "Learning cautiously in federated learning with noisy and heterogeneous clients," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2023, pp. 660–665, doi: 10.1109/ICME55011.2023.00119.

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

[58] M. Ilić, M. Ivanović, V. Kurbalija, and A. Valachis, "Towards optimal learning: Investigating the impact of different model updating strategies in federated learning," *Expert Syst. Appl.*, vol. 249, Sep. 2024, Art. no. 123553, doi: 10.1016/j.eswa.2024.123553.

[59] L. Yan and X. Ge, "Entropy production-based energy efficiency optimization for wireless communication systems," *IEEE Open J. Commun. Soc.*, vol. 5, pp. 6482–6494, 2024, doi: 10.1109/OJCOMS.2024.3476454.

[60] 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 *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2024, pp. 1848–1853, doi: 10.1109/ISIT57864.2024.10619204.

[61] Z. Li, H. Zhu, D. Zhong, C. Li, B. Wang, and Y. Yuan, "A novel framework for distributed and collaborative federated learning based on blockchain and smart contracts," in *Proc. IEEE 3rd Int. Conf. Digit. Twins Parallel Intell. (DTPI)*, Nov. 2023, pp. 1–4, doi: 10.1109/DTPI59677.2023.10365414.

[62] J. Lu, H. Zhang, P. Zhou, X. Wang, C. Wang, and D. O. Wu, "FedLaw: Value-aware federated learning with individual fairness and coalition stability," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 9, no. 1, pp. 1049–1062, Feb. 2025, doi: 10.1109/TETCI.2024.3446458.

[63] R. S. Antunes, C. André da Costa, A. Küderle, I. A. Yari, and B. Eskofier, "Federated learning for healthcare: Systematic review and architecture proposal," *ACM Trans. Intell. Syst. Technol.*, vol. 13, no. 4, pp. 1–23, Aug. 2022, doi: 10.1145/3501813.

[64] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," *Knowl.-Based Syst.*, vol. 216, Mar. 2021, Art. no. 106775, doi: 10.1016/j.knosys.2021.106775.

[65] J. Wen, Z. Zhang, Y. Lan, Z. Cui, J. Cai, and W. Zhang, "A survey on federated learning: Challenges and applications," *Int. J. Mach. Learn. Cybern.*, vol. 14, no. 2, pp. 513–535, Feb. 2023, doi: 10.1007/s13042-022-01647-y.

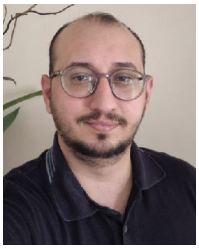
[66] H. Azami, S. Sanei, and T. K. Rajji, "Ensemble entropy: A low bias approach for data analysis," *Knowledge-Based Syst.*, vol. 256, Nov. 2022, Art. no. 109876, doi: 10.1016/j.knosys.2022.109876.

[67] S. Zheng, W. Yuan, X. Wang, and L. Duan, "Adaptive federated learning via new entropy approach," *IEEE Trans. Mobile Comput.*, vol. 23, no. 12, pp. 11920–11936, Dec. 2024, doi: 10.1109/TMC.2024.3402080.

[68] 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 *Proc. IEEE/ACM Symp. Edge Comput. (SEC)*, 2021, pp. 68–79.

[69] S. X. Lee and G. J. McLachlan, "An overview of skew distributions in model-based clustering," *J. Multivariate Anal.*, vol. 188, Mar. 2022, Art. no. 104853, doi: 10.1016/j.jmva.2021.104853.

[70] M. Sakthimohan, E. Rani, G. Naveneethakrishnan, M. Abinaya, M. Karthigadevi, and P. V. Siddhartha, "Digit recognition of MNIST handwritten using convolutional neural networks (CNN)," in *Proc. Int. Conf. Intell. Syst. Commun., IoT Secur. (ICISCoIS)*, Feb. 2023, pp. 328–332, doi: 10.1109/ICISCoIS56541.2023.10100602.
[71] A. Tomar and H. Patidar, "Optimizing CNN model performance for MNIST and CIFAR classification using rectified sigmoid and ReS activation functions," in *Proc. 7th Int. Conf. Comput., Commun., Control Autom. (ICCUBEA)*, Aug. 2023, pp. 1–6, doi: 10.1109/iccubea58933.2023.10392280.
[72] S. Saed, B. Teimourpour, K. Kalashi, and M. A. Soltanshahi, "An efficient multiple convolutional neural network model (MCNN-14) for fashion image classification," in *Proc. 10th Int. Conf. Web Res. (ICWR)*, Apr. 2024, pp. 13–21, doi: 10.1109/icwr61162.2024.10533341.
[73] E. Xhaferra, E. Cina, and L. Toti, "Classification of standard FASHION MNIST dataset using deep learning based CNN algorithms," in *Proc. Int. Symp. Multidisciplinary Stud. Innov. Technol. (ISMSIT)*, Oct. 2022, pp. 494–498.
[74] S. Aslam and A. B. Nassif, "Deep learning based CIFAR-10 classification," in *Proc. Adv. Sci. Eng. Technol. Int. Conferences (ASET)*, Feb. 2023, pp. 01–04, doi: 10.1109/aset56582.2023.10180767.
[75] A. Sikdar, S. Udupa, and S. Sundaram, "Fully complex-valued deep learning model for visual perception," in *Proc. ICASSP - IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2023, pp. 1–5, doi: 10.1109/ICASSP49357.2023.10095290.
[76] D. Li, Y.-C. Hsu, R. Sumikawa, A. Kosuge, M. Hamada, and T. Kuroda, "A 0.13 mJ/prediction CIFAR-100 raster-scan-based wired-logic processor using non-linear neural network," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2023, pp. 1–5, doi: 10.1109/ISCAS46773.2023.10181427.
[77] M. Morafah, M. Reisser, B. Lin, and C. Louizos, "Stable diffusion-based data augmentation for federated learning with non-IID data," 2024, *arXiv:2405.07925*.
[78] L. Ju, T. Zhang, S. Toor, and A. Hellander, "Accelerating fair federated learning: Adaptive federated Adam," *IEEE Trans. Mach. Learn. Commun. Netw.*, vol. 2, no. 1, pp. 1017–1032, Jan. 2024, doi: 10.1109/TMLCN.2024.3423648.
[79] Z. Yang, Y. Zhang, Y. Zheng, X. Tian, H. Peng, T. Liu, and B. Han, "FedFed: Feature distillation against data heterogeneity in federated learning," in *Proc. Adv. Neural Inf. Process. Syst.*, Jan. 2023, pp. 1–12. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2023/hash/bdcdf38389d7fcefc73c4c3720217155-Abstract-Conference.html

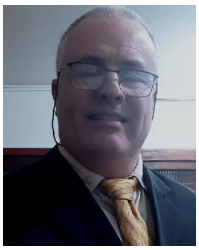


ERNESTO GURGEL VALENTE NETO received the bachelor's degree in computer science and IT management and the master's degree in data science from Centro Universitário Farias Brito (FB UNI), in 2021 and 2022, resepectively, and the master's degree in teleinformatics engineering from the Federal University of Ceará (UFC), where he is currently pursuing the Ph.D. degree. He is a Researcher with the Center of Reference in Artificial Intelligence (CRIA), focusing on artificial intelligence, data science, and health in collaboration with the Processing for Data Analysis and Learning Systems Laboratory (SPIRAL).

SOLON ALVES PEIXOTO JR. received the bachelor's and master's degrees in computer science from the Federal Institute of Education, Science, and Technology of Ceará (IFCE), in 2016 and 2018, respectively, and the Ph.D. degree in teleinformatics engineering from the Federal University of Ceará (UFC), in 2023. He is currently pursuing the Ph.D. degree in electrical engineering. He has been a Professor with the Federal University of Ceará, Campi Itapajé, Brazil. Additionally, he is a Researcher with the Image Processing and Computational Simulation Laboratory (LAPISCO), IFCE, Fortaleza, with a focus on machine learning and pattern recognition in cloud environments, and embedded systems related to the Internet of Things (IoT).

VALDERI REIS QUIETINHO LEITHARDT (Senior Member, IEEE) received the Ph.D. degree in computer science from INF-UFRGS, Brazil, in 2015. He is currently a Professor with the Instituto Universitario de Lisboa (ISCTE-IUL) and a Researcher integrated with the ISTAR-Information Sciences, Technologies, and Architecture Research Centre (ISTA), and the Research Group Software Systems Engineering. He is also a collaborating Researcher with the following research groups: COPELABS, Universidade Lusofona de Lisboa, Portugal; the Laboratory of Embedded and Distributed Systems, University of Vale do Itajai (UNIVALI), Brazil; Federal University of Ceará-UFC, Brazil; and the Expert Systems and Applications Laboratory, University of Salamanca, Spain. His research interests include distributed systems with a focus on data privacy, communication, and programming protocols, involving scenarios and applications for the Internet of Things, smart cities, big data, and cloud computing.

JUAN FRANCISCO DE PAZ SANTANA received the degree in technical engineering in systems computer sciences, in 2003, and the Engineering degree in computer sciences, the degree in statistics, and the Ph.D. degree in computer science from the University of Salamanca, Spain, in 2005, 2007, and 2010, respectively. He is currently a Full Professor with the University of Salamanca, where he is also a Researcher with the Expert Systems and Applications Laboratory (ESALab). He has been the co-author of published articles in several journals and a presenter at workshops and symposia.



JULIO C. S. DOS ANJOS received the bachelor's degree in electrical engineering from PUC/RS, in 1991, and the master's, Ph.D., and Postdoctoral degrees in computer science from UFRGS, in 2012, 2017, and 2021, respectively. He has been a Professor with the Federal University of Ceará, Campi Itapajé, Brazil, and the Graduate Program in Teleinformatics Engineering (PPGETI/UFC), since 2022. His research interests include distributed systems, hybrid infrastructures, collaborative learning, intelligent autonomous systems, and big data analytics with deep learning.

Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) - ROR identifier: 00x0ma614