Blekinge Institute of Technology
Doctoral Dissertation Series No. 2024:04
ISSN 1653-2090
ISBN 978-91-7295-478-6

Resource-Aware and Personalized Federated Learning via Clustering Analysis

Ahmed Abbas Mohsin Al-Saedi

DOCTORAL DISSERTATION
for the degree of Doctor of Philosophy at Blekinge Institute of Technology to be publicly defended on May 17th, 2024, at 10:00 in room J1630, Campus Gräsvik

Supervisors
Prof. Veselka Boeva, Blekinge Institute of Technology
Prof. Emiliano Casalicchio, Sapienza University of Rome and Blekinge Institute of Technology

Faculty Opponent
Prof. György Dán, KTH Royal Institute of Technology, Sweden
Grading Committee
Prof. Volker Markl, Technical University of Berlin, Germany
Prof. Paul Davidsson, Mälmo University, Sweden
Assoc. Prof. Eva Cernadas Garsia, University of Santiago de Compostela, Spain

Abstract

Today's advancement in Artificial Intelligence (AI) enables training Machine Learning (ML) models on the daily-produced data by connected edge devices. To make the most of the data stored on the device, conventional ML approaches require gathering all individual data sets and transferring them to a central location to train a common model. However, centralizing data incurs significant costs related to communication, network resource utilization, high volume of traffic, and privacy issues. To address the aforementioned challenges, Federated Learning (FL) is employed as a novel approach to train a shared model on decentralized edge devices while preserving privacy. Despite the significant potential of FL, it still requires considerable resources such as time, computational power, energy, and bandwidth availability. More importantly, the computational capabilities of the training devices may vary over time. Furthermore, the devices involved in the training process of FL may have distinct training datasets that differ in terms of their size, quality and distribution. As a result of this, the convergence of the FL models may become unstable and slow. These differences can influence the FL process and ultimately lead to sub-optimal model performance within a heterogeneous federated network.

In this thesis, we have tackled a number of the aforementioned challenges. Initially, a resource-aware FL algorithm is proposed that utilizes cluster analysis to address the problem of communication overhead. This issue poses a major bottleneck in FL, particularly for complex models, large-scale applications, and frequent updates. The subsequent step in this thesis involved extending the previous study to include wireless networks (WNs). In WNs, achieving energy-efficient transmission is a significant challenge due to their limited resources. This has motivated us to continue with a comprehensive overview and classification of the latest advancements in context-aware edge-based AI models, with a specific emphasis on sensor networks. The review has also investigated the associated challenges and motivations for adopting AI techniques, along with an evaluation of current areas of research that need further investigation. To optimize the aggregation of the FL model and alleviate communication expenses, the resource-aware FL algorithm is extended with cluster optimization approach. Furthermore, to reduce the detrimental effect caused by data heterogeneity between edge devices on FL, a new study of group-personalized FL models is conducted. Finally, resource-aware techniques to evaluate a client's contribution by assessing its behavior during training are proposed.

The proposed FL algorithms are assessed on a range of real-world datasets. The extensive experiments have demonstrated their effectiveness and robustness. They improve communication efficiency, resource utilization, model convergence speed, and aggregation efficiency in comparison with similar state-of-the-art methods.

Keywords: Federated Learning, Clustering Analysis, Eccentricity Analysis, Non-IID Data, Model Personalization

Blekinge Institute of Technology
Doctoral Dissertation Series No. 2024:04

Resource-Aware and Personalized Federated Learning via Clustering Analysis

Ahmed Abbas Mohsin Al-Saedi

Doctoral Dissertation in Computer Science

Department of Computer Science
Blekinge Institute of Technology
SWEDEN

Blekinge Institute of Technology
Department of Computer Science

Blekinge Institute of Technology Doctoral Dissertation Series No. 2024:04
ISBN 978-91-7295-478-6
ISSN 1653-2090
urn:nbn:se:bth-?????

Printed in Sweden by Media-Tryck, Lund University, Lund 2024

Dedication

To Zaynab, Mohamed and Maryam. I love you all more than words could ever say.

“If we knew what it was we were doing, it would not be called research, would it?”

Albert Einstein

Acknowledgements

First and foremost, I would like to express my utmost gratitude to *Prof. Veselka Boeva*, for her constant guidance, insightful suggestions, and kind support over the last four years. I will always consider myself very privileged to have been one of her students. Also, I extend my sincere gratitude to *Prof. Emiliano Casalicchio*, whose indispensable guidance and valuable feedback were continuous throughout this endeavor.

My sincere appreciation goes to my parents and brothers. I owe immense gratitude to my father. I really wish my father were still alive to share this with us. Extremely grateful to my mother, whose prayers are always with me. I am also so indebted to my father-in-law for his support and encouragement, who has always been there for me.

This would never have been possible without my beloved wife, *Zaynab*, who put her successful academic career and dreams on hold to help me achieve mine. Your unlimited support and unwavering patience have been a constant motivation and strength source through the challenges of my PhD studies. My son *Mohamed* and my daughter *Maryam*, you were refreshing me in all my ways. Words would never express how grateful I am to have you all.

What remains to express is my heartfelt gratitude to all staff in BTH and colleagues in the computer science department for the friendly environment and permanent willingness to help at any time during the entire PhD journey.

March 2024, Karlskrona, Sweden
Ahmed A. Al-Saedi

List of Papers

This thesis is a compilation of the six papers found below. The formatting of the included papers has been changed to conform to a common style, no other changes have been performed.

Paper I

Ahmed A. Al-Saedi, Veselka Boeva and Emiliano Casalicchio. ”Reducing Communication Overhead of Federated Learning through Clustering Analysis”. 2021 IEEE Symposium on Computers and Communications (ISCC), Athens, Greece, 2021, pp. 1-7. DOI: 10.1109/ISCC53001.2021.9631391

Paper II

Ahmed A. Al-Saedi, Emiliano Casalicchio and Veselka Boeva. ”An Energy-Aware Multi-Criteria Federated Learning Model for Edge Computing”. 2021 8th International Conference on Future Internet of Things and Cloud (FiCloud), Rome, Italy, 2021, pp. 134-143. DOI: 10.1109/FiCloud49777.2021.00027

Paper III

Ahmed A. Al-Saedi, Veselka Boeva, Emiliano Casalicchio and Peter Exner. ”Context-Aware Edge-Based AI Models for Wireless Sensor Networks—An Overview”. In: Emerging Sensor Communication Network-Based AI/ML Driven Intelligent IoT, Sensors 2022, 22(15). ISSN: 1424-8220. DOI: 10.3390/s22155544

Paper IV

Ahmed A. Al-Saedi, Veselka Boeva and Emiliano Casalicchio. ”FedCO: Communication-Efficient Federated Learning via Clustering Optimization”. In: Edge-Cloud Computing and Federated-Split Learning in the Internet of Things, Future Internet 2022,

14(12). ISSN: 1999-5903. DOI: 10.3390/fi14120377. The paper is an extension of Paper I.

Paper V

Ahmed A. Al-Saedi and Veselka Boeva. ”Group-Personalized Federated Learning for Human Activity Recognition Through Cluster Eccentricity Analysis”. In Engineering Applications of Neural Networks, June, 2023, pp. 522- 536, Springer, León, Spain.

Paper VI

Ahmed A. Al-Saedi, Veselka Boeva and Emiliano Casalicchio. “Contribution Prediction in Federated Learning via Client Behavior Evaluation”, submitted for journal publication (under review).

Other research contributions that are related to this thesis but are not included:

Paper VII

Boeva, Veselka, Emiliano Casalicchio, Shahrooz Abghari, Ahmed A. Al-Saedi, Vishnu Manasa Devagiri, Andrej Petef, Peter Exner, Anders Isberg and Mirza Jasarevic. ”Distributed and Adaptive Edge-based AI Models for Sensor Networks (DAISeN)”. Position Papers of the 17th Conference on Computer Science and Intelligence Systems, Annals of Computer Science and Information Systems 31 (2022): 71-78. DOI: 10.15439/2022F267

Paper VIII

Emiliano Casalicchio, Simone Esposito and Ahmed A. Al-Saedi. ”FLWB: a Workbench Platform for Performance Evaluation of Federated Learning Algorithms”. 2023 IEEE International Workshop on Technologies for Defense and Security (TechDefense), Rome, Italy, 2023, pp. 401-405. DOI: 10.1109/TechDefense59795.2023.10380832

Funding

- The author is supported by the Iraq Ministry of Higher Education and Scientific Research PhD Scholarship.
- Part of the research work presented in this thesis was partially funded by
  - ”Distributed and Adaptive Edge-based AI Models for Sensor Networks”, Sony Research Award Program 2020 Project.
  - ”Human-centered Intelligent Realities (HINTS)”, a project funded by the Swedish Knowledge Foundation (grant: 20220068).

Author's contribution to the papers

Paper I

Co-defined the research problem. Designed and executed the experimental evaluation. Conducted the main part of the analysis of the results. Wrote and edited most of the paper.

Paper II

Defined the research problem. Designed and implemented the proposed algorithm. Designed and executed the experimental evaluation. Conducted the main part of the analysis of the results. Wrote and edited most of the paper.

Paper III

Co-defined the research problem. Performed the literature search and the main part of the data analysis and synthesis. Wrote and edited the majority of the paper.

Paper IV

Defined the research problem. Designed and implemented the proposed algorithm. Designed and executed the experimental evaluation. Conducted the main part of the analysis of the results. Wrote and edited most of the paper.

Paper V

Defined the research problem. Designed and implemented the proposed algorithm. Designed and executed the experimental evaluation. Conducted the main part of the analysis of the results. Wrote and edited most of the paper.

Paper VI

Defined the research problem. Designed and implemented the proposed algorithm. Designed and executed the experimental evaluation. Conducted the main part of the analysis of the results. Wrote and edited most of the paper.

Abbreviations

AI	Artificial Intelligence.
CFL	Clustered Federated Learning.
CMFL	Communication-Mitigated Federated Learning.
CNN	Convolutional Neural Network.
DL	Deep Learning.
ED	Euclidean Distance.
FedAvg	Federated Averaging.
FL	Federated Learning.
HAR	Human Activity Recognition.
IID	Independently and Identically Distributed.
IoT	Internet of Things.
KD	Knowledge Distillation.
MCL	Markov Clustering.
ML	Machine Learning.
MTL	Multi-Task Learning.
NN	Neural Network.
Non-IID	Non-Independently and Identically Distributed.
PFL	Personalized Federated Learning.
RL	Reinforcement learning.
SGD	Stochastic Gradient Descent.
SI	Silhouette Index.
SNs	Sensor Networks.
TEDA	Typicality and Eccentricity Data Analytics.
WNs	Wireless Networks.

Table of Contents

1 Introduction

Today, developments in Artificial Intelligence (AI)-based approaches such as Machine Learning (ML) and Deep Learning (DL) techniques, in particular, have resulted in remarkable advances of the Internet of Things (IoT) applications. Much of this success is mainly due to the presence of large-scale training infrastructures and vast amounts of training data [1]. The widely used approach involves gathering of data in a central location, processing it in a centralized manner, and then creating a unified model based on the processed data. However, in practical terms, sharing large amounts of IoT data in a central location is expensive and raises privacy issues.

In addition to preserving privacy, the concept of learning on the edge, which involves moving computing to the location where data was initially captured and stored, is becoming increasingly attractive due to its energy efficiency and considerations for climate change [2]. Although the idea of transferring computation to distributed edge devices has been presented for a long time, its application was mainly limited to basic tasks such as querying in sensor networks [3] and fog computing [4]. However, with AI chipsets and available computing resources on edge devices, the training of AI models on these devices has gradually shifted from the central server to edge devices.

In light of this context, Google presented the concept of Federated Learning (FL) [5] as an emerging ML paradigm for decentralized data to address such issues, allowing multiple edge devices to collaboratively train a central AI model without direct access to their private data. Learning occurs locally on the devices, orchestrated by a central server. In this paradigm, instead of sharing raw data, model updates are exchanged. Collaboration among these devices can lead to better generalization, offering advantages in terms of privacy and distributed computation [6]. This is especially advantageous in sectors such as healthcare care, where the utmost importance is placed on data privacy. McMahan et al. [5] introduced the Federated Averaging (FedAvg) algorithm to implement this concept, originally intended for a cross-device scenario often observed in mobile phones. In this context, there exists a large pool of devices that possess limited computational capabilities. These devices are frequently characterized by their unreliability, both in terms of availability due to communication limitations and costs, as well as reliability. The ML model was effectively used to improve the predictive capabilities of Google Board (GBoard), which offers word suggestions to users while typing [7].

The field of FL offers a promising approach, as it addresses the issues of centralized learning while upholding data privacy when applied in the real world [8]. However, despite its potential, this approach still confronts considerable challenges in the domain. These include the elevated communication cost required for data transferred between the central server and client devices, the energy consumption required for client devices, and the diversity of potentially large amounts of data involved in such a process. Some of these challenges are addressed in our papers; in the following section, we will discuss these challenges in more detail.

1.1 Federated Learning Challenges

FL continues to gain attention, researchers are actively working to address the associated challenges and improve system performance. These challenges, owing to the various nature of the federated setting from the traditional problems, introduce an interesting research paradigm to the research community. However, despite FL having several advantages, such as preservation of privacy, collaborative learning, and decentralization. The more benefits it offers, the more challenges it presents that need to be paid attention to. In this section, we discuss specific challenges that can be explored further to improve the performance of the system.

- **Resource limitations**: FL process requires iterative transfer of data (e.g. model parameters, weights, etc.) between a central server and edge devices. For example, when it comes to Neural Network (NN) [9], these models contain a large number of parameters, reaching millions, and require frequent updates to reach the desired convergence. Consequently, the requirement for communication bandwidth is exceptionally ultra-high. On the other hand, in such environments, participating devices are typically small in size and have a constrained nature in terms of connectivity and computing resources [10, 11]. In addition, the number of participant devices can range from hundreds to millions. Thus, FL requires substantial communication resources and energy overhead before reaching the desired accuracy in such a scenario [12, 13]. To reap the advantages of FL, these challenges must be addressed to allow for the wider adoption of FL systems [14]. These challenges have been discussed in **Papers I, II, IV,** and **VI**. Different FL models are suggested to address the issue of communication overhead, minimize energy usage, and evaluate the individual contribution of each client participating in FL. Each paper focuses on these aspects independently.

- **Expensive Communication**: As stated previously, the models for which participating devices train locally and exchange with the server may be quite substantial. For instance, VGG-16, a NN used for image recognition, contains 138

million parameters [15] and requires 526 megabytes of storage when encoded with 32 bits. In addition, the number of devices involved in the FL systems could range from hundreds of thousands to millions, Figure 1.1 illustrates the run-time cost of FedAvg, with the blue and green blocks denoting the local computation time and communication delay, respectively. To demonstrate the impact of different numbers of clients W_t in round (t) on the FedAvg algorithm for a fixed number of training rounds (e.g., $T = 3$), we generate a graph of two global averaging steps. Specifically, we set $W_t = 3$ for the first step and $W_t = 5$ for the second step. The run-time cost of each global averaging step in FedAvg can be observed in Figure 1.1, which consists mainly of two components: the time taken to calculate the local model and the delay in communication with the central server. The computation time per global averaging step in synchronous FedAvg is determined by the slowest client device, while the communication delay is influenced by the shared bandwidth among all client devices. Therefore, although numerous client devices can help accelerate model training, both the computation time and transmission time will increase significantly when W_t is large. Likewise, a decreased communication period T helps in the convergence of the model, but the cost of communication delay will increase compared to the computation time. Hence, the optimization

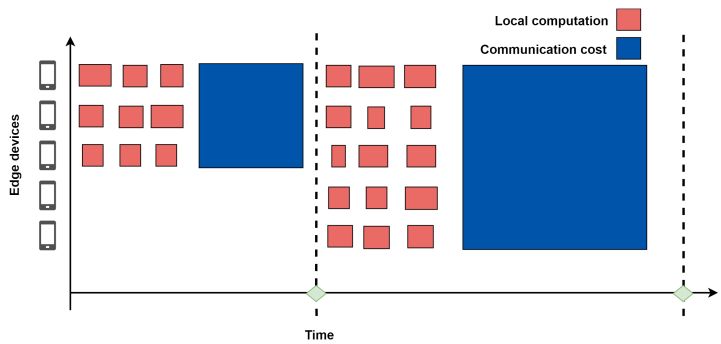


Figure 1.1: An example of the runtime expense of FedAvg algorithm, where the red and blue blocks indicate the time taken for local computation and transmission time, respectively.

of the FedAvg algorithm involves a complex trade-off between the number of client devices and the transmission time. Furthermore, FL systems are frequently characterized by high dynamic, due to the participation of new client devices, as well as the continuous generation of data by existing devices with limited network bandwidth. These limitations pose a considerable communication cost challenge in FL. It directly impacts the effectiveness, scalability, and overall performance of the FL process, making it a crucial area of concern.

In recent years, several techniques have been used to improve communication efficiency in such a context. For instance, one potential way is data compression, like quantization and sparsification methods, which are used to directly

decrease part of the data size. More details about compression methods will be introduced later in Section 3.1.2. However, these methods often face heavy performance when a high compression ratio is required. Furthermore, compression of global model updates could also degrade the model's ability to handle the diversity of decentralized data [16].

Another method commonly used in communication-efficient FL is to minimize the number of model updates that are transferred to the server [12]. Since the exchanging of large model updates requires significant communication resources, it is vital to reduce the volume of data that has to be collected by the server [12, 13, 17]. Our proposed FL models also fall into this category, which has been explored in **Papers I** and **IV** focused on decreasing the number of transferred data.

- **Data heterogeneity**: In ML settings, the data is assumed to be independently drawn from the same joint distribution. This is known as the data is Independently and Identically Distributed (IID). However, when we move to FL, we quickly face violations of this assumption. To be more precise, FL involves training a model on a global scale using multiple distributed devices that might collect unique dataset and possess different class distributions. These distributions, known as Non-Independently and Identically Distributed (Non-IID), reflect real-world applications [18, 19]. Non-IID data represents one of the key challenges in FL [20, 21]. Non-IID data implies that the datasets may vary in size and distribution, which makes it hard to fit all local datasets with one global model. Moreover, the presence of Non-IID data may lead to client drift [22]. Such a phenomenon can considerably undermine the performance of FL [23, 24]. The impact of client drift on IID and Non-IID data is shown in Figure 1.2. In the FedAvg approach, the server updates gradually converge towards the average of client optima. In the case of IID data, the average of client updates is close to the global optimum $\mathcal{M}$, as it is equidistant from both the local optima (e.g, two clients) $\mathcal{M}^1$ and $\mathcal{M}^2$. So, the direction of the average model is also similar to that of the global model. However, in the case of Non-IID data, the global optimum $\mathcal{M}$ is far from the true local optima. In this example, $\mathcal{M}$ is closer to $\mathcal{M}^2$. Therefore, the global model deviates from its true global optimum direction for Non-IID data. In other words, the averaged model (global model) $\mathcal{M}_{t+1}$ in round $(t+1)$ will be far from the true global optimum. Furthermore, the divergence of the model may increase with successive communication rounds (t). Recent works [23–26] have shown that such data heterogeneity in FL approaches can significantly degrade the performance of the global model, which could eliminate the main motivation to participate in FL training. While tradtional FL approaches pursue a global optima of all client devices, the concept of Personalized Federated Learning (PFL) seeks to learn personalized models for each task or specific device [27,

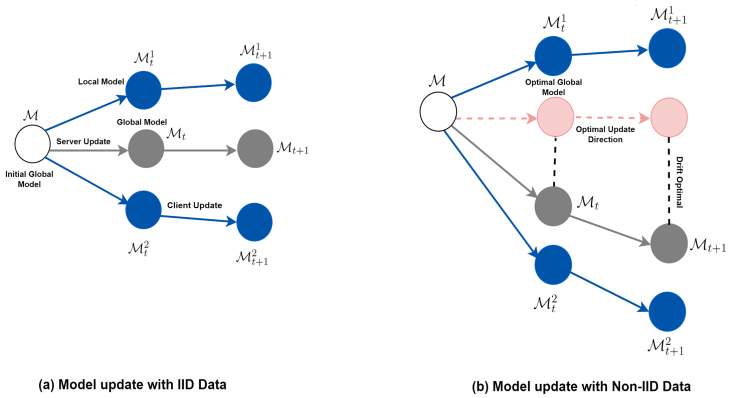


Figure 1.2: Visualization of client drift in FedAvg algorithm for two clients with two local epochs on different structured data. (a) IID data setting. (b) Non-IID data setting.

28] that fit the diverse local dataset. The challenge of data heterogeneity is addressed in **Paper V**.

- **Client selection**: In the FL training process, the naive FedAvg algorithm employs a strategy of randomly choosing clients in each training epoch, without taking into account the available resources of each client. As a result, the performance of the federated model is negatively affected due to the slow convergence rate, decreased training efficiency, and the failure to fully utilize local updates from heterogeneous clients [29]. Meanwhile, the computing capacity and communication resources of the participant devices in FL vary, and may be reluctant to take part in FL training. More importantly, the performance of the federated model are significantly influenced by the client selection scheme involved [30, 31]. Therefore, it is essential to have an effective mechanism to select participants in the FL systems. This challenge has been discussed in **Papers I, II**, and **IV**.

- **Personalization**: The vanilla FL approach creates a single global shared model by taking the average of all local models of client devices [32]. It assumes that all client devices have the same learning task. Furthermore, certain client devices may show poor performance, while others perform well, inevitably eliminating client personalization and decreasing the ability to represent client characteristics [33]. However, in real-world applications, different devices involved may encounter distinct ML problems and possess diverse data distributions that need to tailor specialized/ personalized approaches to meet their specific requirements. For example, vanilla FL does not personalize the model for each client device that is necessary for each user on platforms such as YouTube and Netflix. On the other hand, participant devices may have different models with completely different architectures. These models can include Convo-

lutional Neural Network (CNN) with 5 layers, CNN with 10 layers, ResNet, Random Forest, etc., as a result of diverse computational resources or distinct requirements [34]. Therefore, while there are benefits to using PFL for personal learning, it did not fully take advantage of the potential for collaborative learning. This limitation is problematic for two main reasons: First, most of the client devices have a limited amount of data. Furthermore, despite differences among client devices or tasks, it is reasonable to assume that there is some similarity among devices. The challenge of personalization has been studied in **Paper V**.

- **Security and privacy**: Traditional FL addresses the data security issues that arise from the need to centralize client datasets on a central server for model training. Although traditional FL ensures significant data privacy compared to centralized learning. Recent research indicates that FL applications are still vulnerable to numerous attacks. These attacks can have adverse effects on various aspects, such as the precision of the learned model, the confidentiality, integrity, and availability of the data used [35, 36]. In FL scenarios, the privacy of the input data is ensured by transferring only trained parameters instead of the original raw data. However, model updates in training can accurately infer valuable information [37]. Hence, even though there have been enhancements in comparison to centralized methods, ensuring data security and privacy in FL remains crucial issues that need to be addressed. It is important to note that the privacy aspect will remain beyond the scope of this thesis.

1.2 Thesis Scope and Objectives

This thesis aims *to develop new personalized and resource-aware FL models by using clustering analysis*. These new solutions aim to improve the resource efficiency and robustness of personalized FL when dealing with heterogeneous and dynamic data.

The thesis goal is achieved by addressing the following objectives:

1. *To develop FL resource-aware solutions based on clustering analysis*.
2. *To develop groups' personalized FL models to address data heterogeneity*.
3. *To investigate the recent advances in context awareness for sensor networks using AI*.

1.3 Research Questions

This thesis integrates three distinct directions: resource-conscious FL solutions, PFL solutions, and the exploration of edge-based AI for Sensor Networks (SNs). In this

context, six different studies have been carried out as a component of this PhD thesis. Given the scope and objectives of this thesis. As a result, the answers to the following questions are sought:

RQ 1: *How we can develop FL models that reduce resource consumption without sacrificing the model performance?*
Motivation: In practical federated systems, the participating devices often have limitations in terms of resources such as cache memory, storage, network bandwidth, and processing capabilities. Reducing resources is crucial to achieving cost-effective training in FL. In particular, when dealing with a diverse fleet of client devices that differ in data quality, computational capacity, and battery lifetime levels. Several studies [38–40] have been conducted to minimize the resources of client devices in FL. Hence, it is prudent to develop effective methods in a FL system that considers the constraints of the device resource.
Papers: This research question is explored in **Paper I** through the use of clustering analysis. In this method, the local updates made by representatives of a cluster are identified, and only these updates are uploaded to the central server. This helps to reduce the expenses associated with network communication in Human Activity Recognition (HAR) datasets. **Paper II** tackles the question by incorporating clustering analysis into the Wireless Networks (WNs) environment as well. In this study, various factors such as energy consumption, bandwidth usage, and accuracy are taken into account when choosing a representative from a cluster to communicate with a central server. This study examined how these factors affect FL performance. In **Paper IV**, we extended the work presented in **Paper I** by improving the optimization of model aggregation and minimizing communication overhead. This was achieved by implementing clustering optimization to select representatives. Additionally, the split optimization technique is utilized to update and enhance the overall clustering solution.

RQ 2: *How the clients' behaviour can be efficiently evaluated during the FL process?*
Motivation: Numerous applications illustrate that FL is an effective solution for making collaborative decisions while maintaining data privacy. However, FL still faces the challenge of data quality. Because federated learning fuses models trained by data from various client devices. The data quality from some clients could be low. Low-quality data has two negative impacts. First, they disrupt the training process and cause excessive computational costs [41, 42]. Second, they have the potential to adversely affect the models of other devices involved that possess high-quality data [43, 44]. These concerns might cause certain federated learning participants to withdraw from federated learning training. To address the issue of variable data quality, FL requires a robust method for evaluating data quality.
Papers: The majority of current solutions require significant resources and are typ-

ically executed as an extra evaluation step. This results in a high computational burden for data owners with large datasets. **Paper VI** answers this research question by using the FL models proposed in **Papers I** and **V**. Although the main focus of **Papers I** and **V** were different, we noticed that these models could be used to assess the contribution of the client. However, we did not have enough time to conduct a thorough investigation. Based on these two **Papers I** and **V**, they reversely demonstrate that evaluating the behavior of clients can be used to measure the contribution of clients in **Paper VI**.

RQ 3: *How we can personalize FL models to achieve robust model performance?*
Motivation: In terms of the present research advancements FL, data heterogeneity represents a significant challenge worthy of attention. In FedAvg (the first FL method), all participants jointly train a joint model [35, 45]. However, a single global model may struggle to fit all participants, thereby restricting its generalizability. Furthermore, given the wide range of application scenarios, each client device may need to build distinct local models that are customized to the specific characteristics and data, which cannot be satisfied in the existing FL setting. Although PFL greatly benefits personalized learning, it does not take advantage of collaborative learning between devices. This limitation raises an issue due to the possibility of a small amount of data in each client device and not leveraging the similarity between devices in terms of tasks or data. Therefore, its existing limitation requires the development of a new personalized FL model.
Papers: The aim is achieved in **Paper V**, which presents a clustering-based approach for group-personalized FL in the context of HAR applications. The FL model presented in this study aims to address the issue of heterogeneity of data and achieve a balance between the global model and local models.

RQ 4: *What AI-based solutions are underrepresented in the recent state-of-the-art of context-aware edge intelligence systems?*
Motivation: Context-aware systems must have a refined understanding of the environment surrounding them and be able to make appropriate moves to adapt to different contexts. From this perspective, applying AI techniques to context-aware systems effectively enables such systems to process complex behaviors and adapt to rapidly changing situations in real time. This research question aims to identify AI-based solutions that are not adequately represented in context-aware systems, particularly within SNs. In addition, it aims to determine any research gaps in current state-of-the-art AI-based solutions.
Papers: This goal is achieved in **Paper III**, where a comprehensive investigation was carried out on the application of AI, ML, and DL methods in the latest developments in context awareness in WNs. In this study, an investigation of the existing literature was conducted to provide an overview of various domains, highlight the main obstacles within each field, outline the reasons behind the research, and iden-

tify any gaps in current studies.

The visualization of the included studies and their connection wih the aim, objectives, and research questions of the thesis is provided in Figure 1.3.

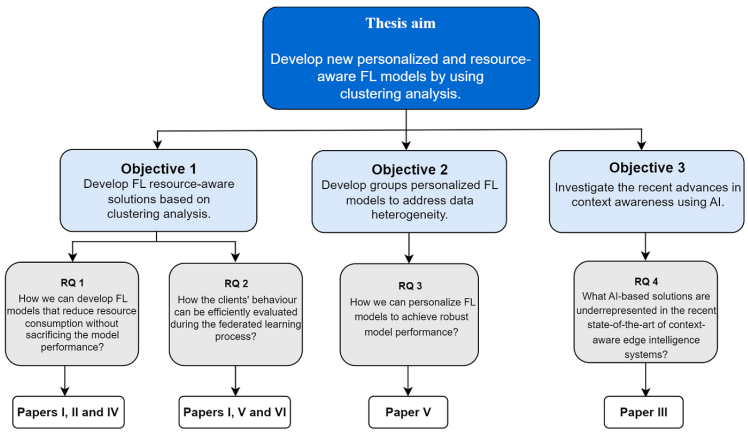


Figure 1.3: A visualization of the relations among the thesis aim, objectives, research questions, and included studies in this thesis.

1.4 Thesis Outline

The remaining sections of this thesis are organized as follows:

Chapter 2 - Background: Here, we elaborate on the relevant background information that serves as the foundation for our thesis.

Chapter 3 - Related Work: This chapter discusses the relevant studies in previous research.

Chapter 4 - Research Methodology: The datasets and baselines used in our studies are presented in this chapter. Additionally, we provide details of Evaluation Measures. We proceed with the examination of the research methodology used in carrying out our studies. Lastly, we address potential threats to research validity.

Chapter 5 - Results and Discussion: This chapter discusses the results of the thesis. Each research question is addressed with the corresponding papers that are part of this thesis.

Chapter 6 - Conclusion of this thesis: Here, we present a conclusion of the thesis and explore future research directions.

2 Background

2.1 Machine Learning

With the advancement in IoT devices, rapid connectivity, AI, and ML have led to the generation of large amounts of data, commonly known as big data [46]. ML techniques are employed to manage large sets of data. However, in traditional ML

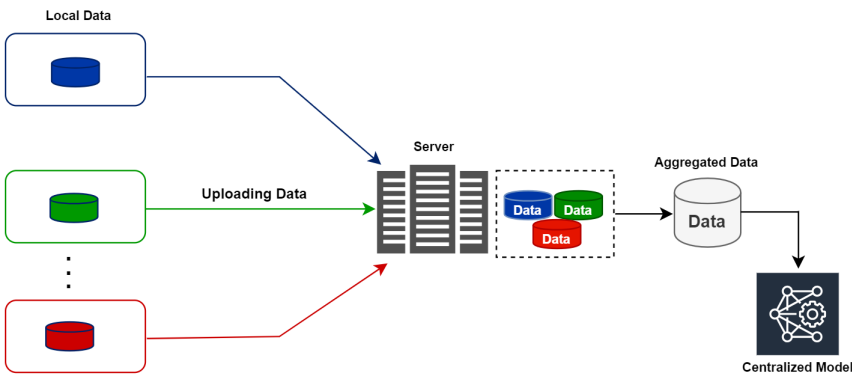


Figure 2.1: General framework for a centralized training approach.

settings, most of these ML techniques are centralized techniques, meaning that data from multiple devices is aggregated into a central server. This central server is used to train a joint model that can then be shipped and applied to all devices that will be used for inference [47]. Figure 2.1 clearly shows that data must be collected from different edge devices to train a joint model. Despite its impressive success, several issues must be highlighted. In fact, the violation of data privacy is high when sensitive data is transferred to a central server. Furthermore, the upload of large chunks of data can also create a huge load on the centralized network and put a huge processing load on a single service provider during joint model training [48].

2.2 Federated Learning

Distributed learning algorithms are designed to address computational challenges that arise when dealing with complex algorithms on large-scale datasets. In con-

trast to centralized machine learning, distributed machine learning algorithms offer improved effectiveness and scalability. In the scenario of distributed learning, the training of a model occurs on multiple devices rather than being centralized in a single location, using a dataset. During training in a distributed algorithm, participants independently train their models and send updates to the central server, where they are averaged [49].

The concept of FL was first proposed by Google in 2017 [5, 50] as a type of distributed machine learning approach that allows training a joint model by cooperating with edge devices without revealing training data under centralized server supervision. The main idea of FL is to cooperatively train ML models among numerous independent edge devices (e.g. mobile phones, wearables, computers, sensors, and IoT devices) under the constraint that training data must remain stored and processed locally in an agreed setting. Instead, the training of the shared model is performed by the edge devices on their local datasets. Updated models are then sent to the central server, which performs the aggregation of these trained local models to produce a unified model, in contrast to traditional centralized machine learning methods [51]. The updated model is then returned to the edge devices for another communication period. This process continues until a stopping criterion is met.

FL ensures data privacy and offers greater scalability compared to centralized learning methods, as it does not involve the exchange of raw data between edge devices. In FL, multiple number of edge devices share a global ML model. Each edge device receives a replica of the shared model and improves it through local learning using its private dataset. The edge device then sends just the updated model to the server, where they are aggregated to produce the global model. By utilizing the resources of edge devices, FL introduces a transition from expensive centralized ML training to a distributed approach [52].

- **Federated averaging (FedAvg)**: The FedAvg algorithm, known for its simplicity and effectiveness, is widely used for federated aggregation. FL consists of a central server and a group of client devices W_t, (i.e. $W_t \subset W$). Each client device uses its local dataset $\mathcal{D}_i$, and n_i is the size of the data set $\mathcal{D}_i$ (i.e. $\mid \mathcal{D}_i \mid = n_i$), where $i = 1, 2, \ldots, N$ denotes the index of the client device involved in FL. Figure 2.2 illustrates a flow diagram of the standard FL approach. To provide background for the proposed methods, this section presents the most commonly used aggregation mechanism in the literature, namely the FedAvg algorithm [5]. In addition, it will serve as a reference for the conducted experimental analysis. Generally, FedAvg can be summarized as follows.

  1. Initialization:
     Step 1: The central server and edge device $W_t \subset W$ are initialized, and the server generates an initial global model $\mathcal{M}_0$ based on the small

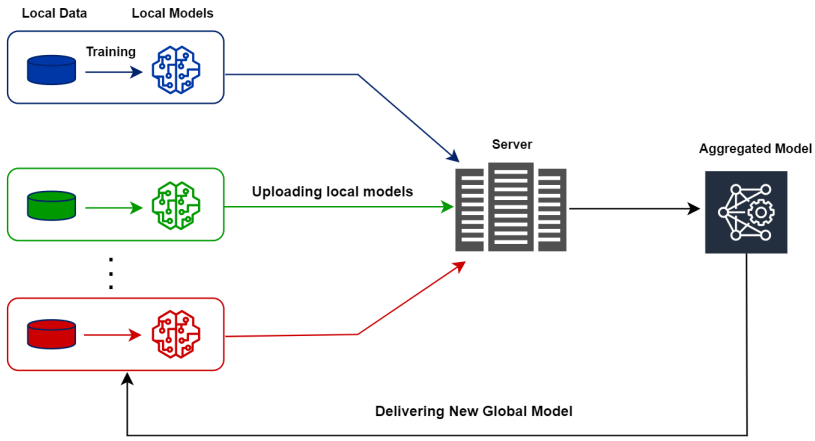


Figure 2.2: General working process of standard FL.

amount of available data. **Step 2**: The central server distributes the global model $\mathcal{M}_0$ to all participating devices $w_i \in W$.

2. Local training:
   Step 3: Each edge device $w_i \in W_t$ performs mini-batch Stochastic Gradient Descent (SGD) with a local training dataset $\mathcal{D}_i$ in parallel and updates the model for a total of E epochs as follows:

$$\mathcal{M}_{t+1}^i = \mathcal{M}_t^i - \eta g(\mathcal{M}_t^i), \tag{2.1}$$

   where t is the index of communication round, η is the learning rate, and $g(\mathcal{M}_t^i)$ refers to the stochastic gradient, which is calculated as follows:

$$g(\mathcal{M}_t^i) = \frac{1}{N_{w_i}} \sum_{\mathcal{D}_i} \nabla \ell(\mathcal{D}_i; \mathcal{M}_t^i). \tag{2.2}$$

3. Global aggregation:
   Step 4: Each edge device $w_i \in W_t$ uploads the updated model $\mathcal{M}_t^i$ (called the local model) to the central server.

   Step 5: Once all local updates have been received, the central server proceeds to initiate the global aggregation process. The updated global model is obtained using the average weighted aggregation method, as specified by the following Eq. 2.3

$$\mathcal{M}_{t+1} = \mathcal{M}_t + \frac{\sum\limits_{w_i \in W_t} p_i \mathcal{M}_{t+1}^i}{\sum\limits_{w_i \in W_t} p_i}, \tag{2.3}$$

where p_i is the relative weight of client w_i. This updated global model is also used as a starting point for the next communication round. However, the model weights are averaged in the traditional FedAvg framework.

Steps 2-5 are repeated until the entire FL process stops after t rounds. The algorithm 1 provides pseudocodes for the FedAvg algorithm, which involves the participation of a set of client devices denoted as $W_t \subseteq W$, as stated in [53]. The structure of the model of both the global model $\mathcal{M}_t$ and all local models $\mathcal{M}_t^i$ is identical, with different values of the model parameters (where t represents the communication round). Under this assumption, direct model aggregation can be implemented as described in line 7 of Algorithm 1, in which each local model uploaded $\mathcal{M}_t^i$ in E local epochs using a learning rate η. Generally, to improve the performance of the FL system, it is often important to select an appropriate setting of four hyperparameters, $W_t \subseteq W$, E, $\mathcal{B}$, and η, based on the specific task that can be determined by metaheuristic search methods [54, 55].

Algorithm 1 FedAvg. W is the total number of client devices; T is the total number of global rounds, E is the total number of local training epochs, $\mathcal{B}$ the local mini-batch size and η is the learning rate.

Input: Initial shared model $\mathcal{M}_0$, set of clients $W_t \subseteq W$, the number of iterations T
Output: The FedAvg procedure global model $\mathcal{M}_t$ for T iterations

1: **procedure** FEDAVG($\mathcal{M}_0, W_t \subseteq W, T$)
2: $t \leftarrow 0$
3: **while** $t \leq T$ **do**
4: $t \leftarrow t + 1$
5: $\forall\, w_i \in W_t$, the server exec SEND($w_i, \mathcal{M}_t$)
6: Each $w_i \in W_t$ exec CLIENTUPDATE($w_i, \mathcal{M}_t$)
7: $\mathcal{M}_{t+1} = \sum_{w_i \in W_t} \frac{n_i}{n} \mathcal{M}_{t+1}^i$ ▷ global update, (2.3)
8: **end while** ▷ Stopping criteria is met
9: **end procedure**
10: **function** CLIENTUPDATE($(w_i, \mathcal{M}_t)$)
11: B = (split $\mathcal{D}_i$ into batches of size $\mathcal{B}$)
12: RECEIVE($w_i, \mathcal{M}_t$)
13: **for** each local epoch i from 1 to E **do**
14: **for** each batch $b \in B$ **do**
15: $\mathcal{M}_{t+1}^i \leftarrow \mathcal{M}_t^i - \eta g_t^i$ ▷ *Local update*, (2.1)
16: **end for**
17: **end for**
18: SEND($i, \mathcal{M}_{t+1}^i$)
19: **end function**

However, Algorithm 1 highlights that the global model $\mathcal{M}$ and local updates $\mathcal{M}^i$ need to be frequently downloaded and uploaded, as indicated in lines 5 and 6. This process consumes a significant amount of communication resources, compared to those typically required for standard centralized learning. Furthermore, FedAvg has established that the more heterogeneous the data, the longer FedAvg takes to converge [22, 23]. Accordingly, a robust and efficient aggregation strategy is crucial to the success of FL.

- **Non-IID Data**: In most of ML and data science scenarios, it is generally assumed that the data are independently sampled from a similar joint distribution. This is known as assuming that the data are IID. To generalize the population from which the data is drawn, it is necessary to consider that each data point is independent of the others and that the population remains unchanged as data points are gathered (identically distributed). In other terms, the data is more uniform [56]. However, in a FL scenario, the devices involved in training are typically IoT devices that produce data distribution that is unstructured and highly random with each other. More specifically, client devices may have different label distributions, and some labels may be more available on some devices than others. This phenomenon is referred to as Non-IID, which can lead to significant model divergence [57, 58]. In the context of supervised learning on a specific device i, let's consider a data sample (x, y), where x represents the features and y denotes the labels, follows a local data distribution $P_i(x, y)$. Non-IID refers to the situation where P_i varies from one device to another. Although McMahan et al. [5] argue that FedAvg can handle Non-IID data to some extent, numerous studies have suggested that a deterioration in FL accuracy is almost inevitable when dealing with Non-IID [24]. Different types of Non-IID Partitions have been introduced based on features x, labels y, and other more complex FL scenarios. The attribute skew, the label skew, and the temporal skew represent the main Non-IID Data categories.

  The *label skew* is investigated in our thesis studies; specifically, different degrees of *label distribution skew* are studied in our papers, where the label distributions $P_i(y)$ on the clients are different. We control the skewness by controlling the fraction of data that is Non-IID. For example, 0% non-iid would mean that the data labels are uniformly/evenly distributed between clients. 30% of the data is Non-IID with 2 clients and 2 labels would imply that one client has at least 30% of one label, while the rest is evenly distributed.

2.3 Clustering Analysis

Clustering plays a crucial role in the research and application of data mining. It is an active research topic that has been applied in various fields, including data science,

and statistics [59–61]. Cluster analysis is a traditional unsupervised classification method that aims to uncover the inherent structural characteristics and patterns of the data and label the data to reveal potential information [62]. Cluster analysis divides datasets into multiple categories, reducing the dissimilarities between data in the same group and increasing those between data in different groups. This section introduces the various components necessary for conducting a cluster analysis approach. In this section, we begin by discussing clustering techniques, specifically focusing on partitioning algorithms. Towards the end of the section, we also introduce similarity measures.

Nowadays, the real world is full of a huge amount of Big Data as a result of the continuous increase in the volume of data every day. Thus, clustering techniques can be employed to uncover interesting patterns within these massive datasets, even with little or no background knowledge [63]. The clustering of client devices is an essential part of our proposed FL models, which involves grouping devices that share similar characteristics. This enables the participating devices to take advantage of collaboration with other devices that exhibit similar learning traits [64, 65]. This is advantageous in contrast to naive FL training, where irrelevant devices contribute to each other, which can potentially harm the performance of their respective datasets. Hierarchical clustering, centroid-based clustering, and density-based clustering are the three most widely used clustering techniques. These algorithms are represented by agglomerative clustering, k-medoids clustering, k-means clustering, DBSCAN, density peak clustering, etc. [66].
The focus of this thesis has been on k-medoids [67] and Markov Clustering (MCL) [68], due to their relevance to FL settings. k-medoids algorithm has a parameter to determine the number of clusters that defines how many clusters should be obtained. In contrast, MCL has a parameter called ”inflation” that indirectly affects the precision of the clustering. Increasing the inflation parameter results in a higher number of clusters.

- **k-medoids Clustering**: k-medoids clustering is a modified version of the k-means-based clustering method that is more robust to noise and outliers. Instead of selecting the mean point as the cluster center, k-medoids clustering chooses an actual point within the cluster to represent it. The k-medoids is the object within a cluster that is located at the center and has the lowest sum of distances to all other points [69]. In this algorithm, we begin by initially selecting k data points and iteratively moving towards the data points in the best cluster. We then examine all possible combinations of data points and assess the clustering quality for each pair of points. If a data point is found with the most enhanced distortion function value, it will replace the current best data point. The newly created optimal data points form the enhanced medoids. This algorithm aims to minimize the dissimilarities between data points and their reference points.

Given a finite set of initial data points $P = \{p_1, p_2, \ldots, p_n\}$, $i = 1, \ldots, n$, we need to split into disjoint clusters k. The clustering of k -medoids selects k medoids $C = \{ob_1, ob_2, \ldots, ob_k\}$ from the set P, to minimize the objective function known as the absolute error function (E) in 2.4:

$$E = \sum_{j=0}^{k} \sum_{p \in c_j} \mid p - ob_j \mid \tag{2.4}$$

Where E represents the sum of the absolute error. The variable p, which belongs to the set P, represents a data point that corresponds to an object in the cluster C_j from the set of clusters C. Additionally, ob_j denotes the representative object of the cluster C_j.

Therefore, medoids (also referred to as client devices in our thesis) are chosen from actual data to serve as cluster representatives. It is important to note that the Euclidean Distance (ED) will be adopted in k-medoids in this thesis.

- **MCL**: It is an efficient graph clustering algorithm. Unlike the k mean and k medoids, this algorithm does not require prior knowledge of the number of clusters. This clustering algorithm is widely used in bioinformatics for clustering protein sequences and co-expression data of genes. Additionally, this algorithm is well suited for distributed computing [68]. The MCL procedure involves two operations performed on stochastic matrices, namely Expand and Inflate. The expansion of matrix M is defined as the result of multiplying M by itself, that is, $M * M$. On the other hand, the inflation operation Inflate (M, r) involves raising each entry in the matrix M to the power of the inflation parameter r (where r is greater than 1, typically set to 2) and then normalizing the columns so that they sum up to 1. This operation is expressed as follows:

$$M_{inf}(i, j) = \frac{M(i, j)^{rM}}{\sum_{k=1}^{n} M(k, j)^{rM}} \tag{2.5}$$

Next, we assign the matrix M_{inf} to M. MCL was used in this thesis to divide client devices with similar empirical probability vectors into similar groups.

In data analysis, similarity measurements are used to discover similarities or dissimilarities between data samples [70, 71], allowing the generation of valuable findings within large datasets. Moreover, these terms are often used in clustering techniques when data instances are split into clusters or to determine the similarity between data points within a given cluster [72]. Centroid-based algorithms represent a notable example of such methods. The selection of a distance metric has a significant impact on the effectiveness of the machine learning classifier. Therefore, how

Table 2.1: Distance measures.

Measure	Equation
Euclidean [73]	$D_{Euc}(p,q) = \sqrt{\sum_{i=0}^{n}(p_i - q_i)^2}$ where p, q is two data points in the Euclidean n-space, q_i, p_i are Euclidean vectors, and n is n-space.
Jaccard [74]	$Sim_{jac}(A,B) = \frac{\lvert A \cap B \rvert}{\lvert A \cup B \rvert}$ Where A and B are two finite sets, and $\lvert A \cap B \rvert$ is the size of the intersection and $\lvert A \cup B \rvert$ size of the union of the sample sets.
Wasserstein [75]	$D_{was}(X,Y) = \min \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} f_{ij}$ Where X and Y are probability distributions, m and n denote the points of X and Y, respectively, d_{ij} denotes the distance from the i point of X to the j point of Y and f_{ij} denotes the number of moves from i to j, $f_{ij} \geq 0, i = 1, \ldots, m, j = 1, \ldots, n$.

distances are calculated between objects is a critical factor in determining the performance of the classifier algorithm. Distance measures are formulated in Table 2.1.

Identifying the appropriate number of compact and well-separated clusters is among the most challenging tasks in cluster analysis. Typically, cluster validity techniques are utilized to assess the quality of clustering solutions, with a specific focus on the compactness and separability of clusters.

2.4 Cluster Validation Measures

The cluster validity measures serve as the evaluation criteria for assessing the quality of the clustering results. To determine the best clustering strategy, the Silhouette Index (SI) [76] is used to evaluate the clustering results. SI assess clustering validity and detect compact and well-separated clusters [77, 78].

The quality of a clustering solution $C = \{C_1, C_2, \ldots, C_k\}$ can be evaluated using SI. Let a_i denote the average distance between the data point i and all other points to its own cluster, and let b_i denote the minimum average distance between the data point i and the points in all to another cluster. The value of $s(i)$ for item i can be calculated using the following formula:

$$s(i) = (b_i - a_i)/\max\{a_i, b_i\}\cdot \tag{2.6}$$

According to its definition, the value of $s(i)$ falls within the range of $[-1, 1]$. If $s(i)$ is close to 1, it indicates that the data point i is assigned to be 'well-clustered' On the other hand, if $s(i)$ is equal to 0 or close to 0, it suggests that the data point i lies between two clusters, making it unclear to which cluster it should belong. In this scenario, the data point can be considered as an 'intermediate case'. Lastly, when $s(i)$ is close to -1, it results in 'misclassification' of the data point i. The SI indicates

which data points belong to their respective clusters and whether they are located closer to one cluster or in between clusters. In other words, it can provide information on the degree of separation between a specific cluster and the others.

The SI can also be computed for each cluster C_j ($j = 1, 2, \ldots, k$) of n_j objects using the following formula:

$$s(C_j) = \frac{1}{n_j} \sum_{i=1}^{n_j} s(i) \cdot \tag{2.7}$$

Furthermore, SI for the entire clustering solution C containing n items is calculated as

$$s(C) = \frac{1}{n} \sum_{i=1}^{n} \frac{(b_i - a_i)}{\max\{a_i, b_i\}} \cdot \tag{2.8}$$

2.5 Typicality and Eccentricity Data Analytics

TEDA, which stands for Typicality and Eccentricity Data Analytics (TEDA), is a statistical approach that utilizes the principles of typicality and eccentricity to categorize similar data observations. Instead of using the conventional concept of clusters, the data is organized into granularities known as data clouds. These data clouds are structures that exist within predefined shapes or boundaries [79]. In [80], novel principles for anomaly detection analysis have been presented, focusing on eccentricity. Building upon these principles, a new algorithm named AutoCloud is proposed in [81]. Eccentricity refers to the degree to which a specific data instance differs from other instances and from its cluster. In this context, the calculation of the eccentricity ξ^j for the data sample i for a cluster of data C_j can be computed as [81]:

$$\xi^j(i) = \frac{1}{n_j} + \frac{(\mu_i^j - \hat{p}_i)^T (\mu_i^j - \hat{p}_i)}{\sigma_i^j}, \tag{2.9}$$

Where n_j represents the size of C_j, $\hat{p}_i$ denotes the empirical probability vector corresponding to the data sample i, μ_i^j is the mean and σ_i^j indicates the variance, assuming that i belongs to C_j. Eq. 2.10 demonstrates the utilization of eccentricity to determine the membership of a data sample in a specific cluster.

In addition, the Chebyshev inequality has been used to apply a threshold to verify whether a data sample remains part of a current cluster [82]. A specific data sample i is considered to be a member of the group C_j if the following condition is met.

$$\xi^j(i) \leq \upsilon_j \text{ and } \upsilon_j = (m^2 + 1)/2n_j, \tag{2.10}$$

Where the parameter m ($m > 0$) is defined by the user and directly affects the evaluation of the cluster, and υ_j is the threshold associated with the cluster C_j.

Although it can be defined using multiple criteria, $m = 3$ is commonly used as a standard value and leads to satisfactory results for different datasets and different configurations [83]. We utilize eccentricity analysis, similar to the approach used in AutoCloud, to maintain the clustering solution of client devices.

3 Related Work

Our focus in this section is on previous work that tackles resources and personalization in the FL setting. For this reason, we categorize the study of previous research into two main categories, namely: resource-aware FL and Personalized FL. Figure 3.1 illustrates our proposed taxonomy of related work and the corresponding subgroups according to the approach used in the solutions.

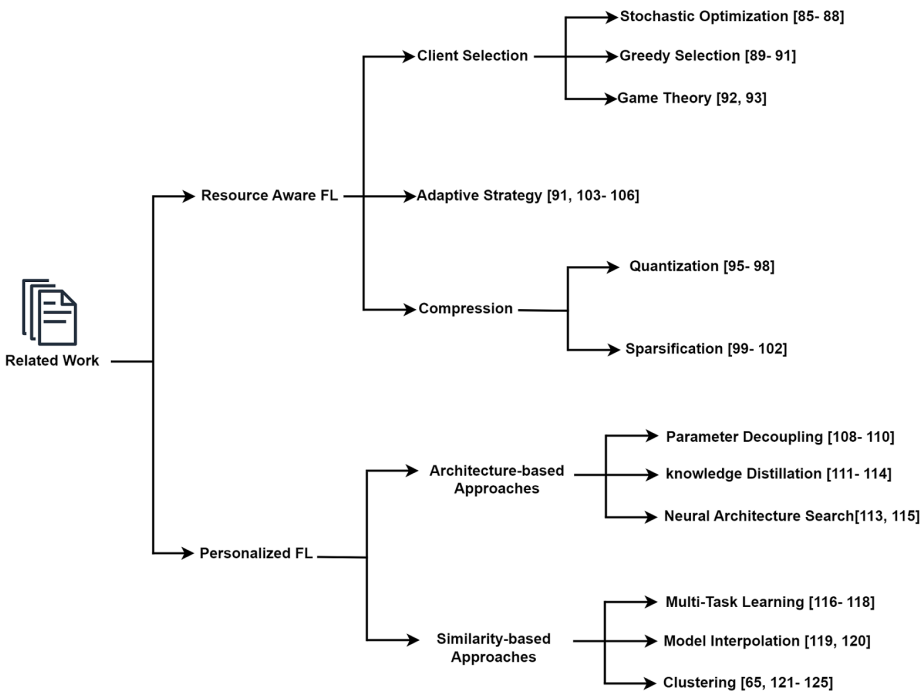


Figure 3.1: Classification of related work in the thesis.

3.1 Resource Aware Federated Learning

In the following, we will examine current research efforts that are closely related to reducing the resource consumption of participants in FL. As transmission models require a significant amount of communication resources, this aspect will be inves-

tigated within existing research aimed at minimizing the communication resources needed by clients. Current research solutions on minimizing communication costs in the field of FL can be classified into the following categories.

3.1.1 Client Selection

During a FL training round, the communication bottleneck is exacerbated by the exchange of model updates between a large number of participating clients. Using a random selection method, such as FedAvg, to choose a subset of clients is a viable approach. However, this randomness can lead to a significant number of missed potentials. In most FL implementations, the clients differ in terms of their design and capability. This diversity also extends to the quality of the communication mediums used. By selecting clients that have the most favorable communication conditions in each round, it is possible to increase the average data rate and consequently reduce communication costs.

- Stochastic Optimization: Chen et al. [84] selected clients that achieved the most optimal probability sampling for clients during each round of communication. In [85], the proposed algorithm chooses a subset of clients with a higher global loss value that have a higher chance of being selected. FLOB framework identifies a subset of clients that achieve the minimum global loss using biased stochastic optimization [86]. AdaFL assigns more importance to clients with a higher probability value, which is determined by the difference between the local updates tensor and the global one [87].

- Greedy Selection: The problem addressed by Balakrishnan et al. in [88] is tackled by FedAvg with Diverse Client Selection (DivFL), which is a greedy approach. FedMCCS [89] takes into account various criteria when selecting clients, including training time, memory size, CPU capacity, and energy consumption during training. To achieve early convergence without increasing communication costs, Wang et al. [90] proposed a method called Communication-Mitigated Federated Learning (CMFL).

- Game Theory: Le et al. [91] defined the problem of FL using the concept of an auction game. The clients act as bidders and the central server acts as the auctioneer. The objective is to select a client with the payoff being the client's selection. The authors in [92] developed a similar auction-based incentive strategy to efficiently choose clients in the FL context.

3.1.2 Compression

Compression techniques play a crucial role in the efficient utilization of edge resources in FL. Specifically, these techniques aim to reduce the size of data and fa-

cilitate the exchange of models between edge devices and the central server, while maintaining the accuracy of the models [93]. Consequently, this reduces the time required to receive the data. In the context of FL, compression is particularly advantageous for handling DL models due to the typically large size of the communicated updates, allowing the execution of FL on edge devices with limited resources. Listed below are a variety of techniques that can be employed to compress models in FL:

- Quantization: Quantization can be utilized to represent models as integers with reduced precision, rather than transmitting high precision floating point numbers [94]. QSGD algorithm in [95] balanced the trade-off between convergence and quantization levels and minimized the communication cost. TernGrad algorithm in [96] involves quantizing floating-point numbers. CosSGD algorithm in [97] utilizes the cosine function to allocate a finer quantization space for values with more significant gradients.

- Sparsification: Sparsification refers to selectively sending partial gradients and reducing communication costs by discarding some gradients with small contributions [98]. The authors proposed using a constant compression rate to choose the sent gradients in [99]. Sparsifying gradients using fixed proportions of positive and negative gradients was introduced in [100]. A LAG sparse communication algorithm adaptively computes a threshold in each round of communication to minimize part of the transmission of gradients [101].

  However, current methods usually perform compression on the device side, resulting in the need for additional processing for both encoding and decoding. In addition, these strategies usually suffer from a significant decrease in performance when the compression ratio needs to be very large [16].

3.1.3 Adaptive Strategy

Several communication strategies can be adapted to reduce communication and resource utilization in the FL context such as those induced by [90]. A framework called communication-mitigating FL framework CMFL was proposed, which enables the transmission of only relevant local updates to the server. This method not only speeds up convergence but also reduces the number of communications needed. In [102], the authors proposed a method that aimed to improve the convergence analysis by advocating the use of an optimal and unbiased sampling technique. The authors in [103] have introduced a method for adaptively aggregating partial models in FL using Reinforcement learning (RL). This strategy aims to optimize the selection of the client devices involved. Wu et al. [104] have introduced a method called FedMed, with adaptive aggregation utilizing the topK strategy to identify the top workers with minimal losses to update the model parameters in each communication round. Similarly, Asad et al. [105] have introduced a filtering method on each local update that

allows the transmission of only the important gradients. Although the methods mentioned above reduce communication costs, they also result in inefficient utilization of computational resources for devices that perform local training without involvement in model aggregation.

Our FL proposed models in **Papers I, II, IV,** and **VI** lie in this category, which introduces a new line of approaches for efficient communication FL that are orthogonal to most of current FL methods.

3.2 Personalized Federated Learning

In this section, we examine PFL methods that focus on developing personalized models. Inspired by the classification of PFL in [106], the strategies were categorized into Architecture-based Approaches and Similarity-based Approaches in the following manner.

3.2.1 Architecture-Based Approaches

Architecture-based PFL approaches for personalized learning aim to achieve customization by tailoring a model for each client. The following sections will explore these techniques to achieve PFL:

- Parameter Decoupling: Parameter decoupling is the process of decoupling the specific parameters of the local private model from the parameters of the global FL model. The Private parameters are trained on the clients' devices without being transmitted to the FL server. This allows the learning of task-specific representations to improve personalization. The authors in [107] introduced an approach that involves splitting model layers into global and personalized parts. These parts are updated separately at different frequencies, with more frequent exchanges of parameters in important layers. Furthermore, in a related work [108], authors presented a layer-wise PFL method. It enhances the aggregation of personalized models by taking into account the significance of individual layers from various clients. Arivazhagan et al. in [109] introduced the concept of "base layers + personalized layers", in which clients keep their personalized layers to learn task-specific representations while sharing the base layers with the server to learn generic features.

- knowledge Distillation: The technique of Knowledge Distillation (KD) allows the transfer of knowledge from one model to another. This procedure involves training the local model to copy the behavior of the more complex model. The authors in [110] introduced a distillation term to the local objective function to train the local model using the output of the global model. In the work of Jeong et al. [111], they utilize the a KD technique to allow individual clients to

determine statistical differences between their local models, leading to better personalization and increased effectiveness within the FL setting. Jin et al., in [112] proposed a method that allows the retrieval of personalized knowledge for new clients by enabling them to distill the knowledge derived from past personalized models into their current local models. The authors proposed a decentralized FL that involves mutual knowledge sharing among local clients in [113].

- Neural Architecture Search: An approach employed to autonomously discover efficient NN structures for a specific task. This method includes exploring through a range of potential network architectures and choosing the optimal design according to a predefined objective, such as improving accuracy or reducing computational resources. The authors introduced a new approach named federated classifier averaging (FedClassAvg) for PFL in [112]. FedClassAvg allows clients with heterogeneous NN architectures to participate in collaborative training without sharing sensitive data, all while maintaining communication efficiency. Wan et al. [114] present a new approach known as Federated Modular Network (FedMN) for PFL. The FedMN technique involves assembling diverse neural architectures by selecting sub-components from a module pool, specifically tailored to the distinct characteristics and requirements of each client.

3.2.2 Similarity-Based Approaches

Similarity-based approaches focus on achieving personalization by representing client relationships. A personalized model is trained for every client and clients with similarities learn similar models.

- Multi-Task Learning: The objective of Multi-Task Learning (MTL) is to develop a model that can jointly perform multiple related tasks. This enhances generalization by utilizing domain-specific knowledge across the various learning tasks. By considering every FL client as a task within MTL, it is possible to capture the relationships between clients based on their heterogeneous local data. The MOCHA algorithm [115] was introduced to extend the distributed MTL to the FL context. It employs a primal-dual formulation to enhance the optimization of the models being learned. MOCHA develops a personalized model for each FL client. [116] introduced the VIRTUAL federated MTL algorithm, which conducts variational inference through a Bayesian method. In the work by Huang et al. [117], a method for pairwise collaboration among FL clients with similar data distributions was introduced.

- Model Interpolation: In the work by Hanzely et al. [118], a novel approach has been introduced that aims to train personalized models using a mixture of

global and local models to balance generalization and personalization. Every federated learning client is trained on a personalized local model. A penalty parameter is used to incentivize local models to be similar to the average model. Adaptive PFL algorithm was introduced by the authors in [119] to find the best combination of global and local models. A mixing parameter was introduced for each client, which is adaptively learned to control the weights of the global and local models. This allows for the optimal level of personalization for each client.

- Clustering: In scenarios where there are inherent partitions between clients or varying data distributions, utilizing a client-server FL architecture to train a joint global model may not be the most efficient approach. Using a multi-model training where a FL model is trained for every homogeneous group of clients is more appropriate. Several recent studies focus on clustering for FL personalization. The fundamental idea behind clustering-based FL is the presence of an inherent grouping of clients determined by their local data distributions. Sattler et al. [120] group FL clients based on the similarity measures of local models to train multiple global models instead of a single model. Clients with similar models are grouped together in the same cluster, and only models within the same cluster are aggregated on the global server. The authors in [64] apply a hierarchical clustering-based FL method to gather clients based on the similarity of their local updates. The approach forms a set of groups, each comprising a group of clients with similar data. Ghosh et al. [121] introduced an iterative Federated Clustering Algorithm (IFCA), which groups clients based on the similarity of data distribution. This approach allows them to collaboratively train a shared model within their group. The authors proposed a FL cluster approach named FedGroup in [122] to achieve personalization by modeling the similarity of different clients. The CFedPer approach proposed in [123] includes a pre-start phase for grouping clients and an in-training phase comprising a base layer and a personalization layer. The authors introduced a personalized FL framework in [124], which identified clients sharing similar data distributions for clustering, followed by conducting co-distillation within the cluster to allow PFL.

  Nevertheless, incorrect clustering may lead to degradation of the system's efficiency in federated learning algorithms. In addition, clustering involves high computational and communication expenses that limit the practical applicability in large-scale settings [64, 120]. The models proposed in **Papers V** and **VI** fall into this type, paving the way for a new line of strategies for group-personalized models.

4 Methodology

This thesis introduces new personalized and resource-aware FL models through clustering analysis. These new solutions aim to improve the efficiency and robustness of personalized FL resources in the face of diverse and evolving data. This chapter introduces the data sets and baselines used in this thesis. Additionally, the evaluation measures are described. The research methodology used in this thesis is then introduced. Finally, the chapter concludes by presenting the validity threats of the conducted studies. First, we give details of the datasets and models applied to the different studies that demonstrate the performance of our proposed FL models on several real-world datasets, federated datasets, and models. Then, we introduce the baseline algorithms that are used to compare with our proposed FL models.

4.1 Datasets

The datasets used to evaluate the FL models in this thesis are outlined in this section. A combination of synthetic data and publicly available real-world data has been used. In particular, we benchmark the proposed FL models on ten different datasets– MHealth [125], PAMAP2 [126], MNIST [127], FashionMNIST [128], CIFAR-10 [129], FEMNIST [130], CelebA [131], REALWORLD [132], HHAR [133] and Synthetic [134] – with respective learning models: (1) logistic regression; and (2) a CNN.

Table 4.1: Summary of the benchmarks used in this thesis.

Task	Model	Dataset	Classes	Papers
HAR	Logistic Regression	MHealth [125]	12	I, II
		PAMAP2 [126]	17	I, II
		REALWORLD [132]	8	V
		HHAR [133]	6	V
Image Classification	CNN	MNIST [127]	10	IV
		FashionMNIST [128]	10	IV
		CIFAR-10 [129]	10	IV
		FEMNIST [130]	62	IV, VI
		CelebA [131]	-	IV, VI
Cluster Identification	Logistic Regression	Synthetic [134]	-	VI

Table 4.1 provides details on the datasets used in our studies, including the model used and the available classes. In **Papers I** and **II**, we used two HAR datasets

containing physical activity monitoring data. The MHealth and PAMAP2 datasets are used to monitor physical activity. Both datasets contain motion sensor data for various physical activities. In **Paper IV**, we used five datasets, namely the MNIST, Fashion MNIST, CIFAR-10, FEMNIST, and CelebA datasets. The MNIST, Fashion-MNIST, and CIFAR-10 are commonly used as benchmark datasets for image classification. Furthermore, we use LEAF datasets [134] that are more realistic than the simulated datasets. **Paper V** introduces two practical datasets available online, REALWORLD and HHAR from the HAR domain. Finally, in **Paper VI**, we used three LEAF datasets. FEMNIST, CelebA, and Synthetic Dataset are used to show the robustness of our proposed FL models.

4.2 Baseline Algorithms

In order to show that our FL proposed models can bring a better training performance and save communication costs, various methods are selected for comparison.

- **FedAvg**: Federated averaging is the first published FL algorithm proposed by McMaham et al. [5]. The approach consists of simply averaging the local updates of the different models communicated by the client devices, as described in section 2.2.

- **FedProx** [135]: FedProx addresses the challenges posed by heterogeneous networks by exploring the limitations of FedAvg algorithm in Non-IID settings. FedProx controls the deviation of local updates from the most recent global model. The devices that participate in the FL process utilize a proximal update technique to ensure that the client model does not deviate from the global model.

- **CMFL** [136]: CMFL improves the efficiency of communication in FL, guaranteeing the achievement of learning convergence. In the FL scenario, CMFL aims to decrease communication overhead by eliminating the need to transmit irrelevant client updates. This approach effectively reduces network usage and minimizes overhead.

- **Clustered Federated Learning (CFL)** [120]: CFL aims to mitigate the detrimental impact of Non-IID data in FL scenarios where the data distribution of individual clients varies. The CFL method divides client populations into clusters that have similar data distributions. This allows for training the same model on each cluster, which alleviates the effects of data heterogeneity on the overall performance of the FL approach.

- **Deletion Approach**: A technique based on deletion diagnostics [137] calculates the contributions of each client in FL utilizing Shapley values. This en-

sures that each party's contributions are correctly appreciated, and motivates high-quality ones to join as early as possible.

- **FL-Cohort**: The proposed algorithm [138] calculates the contribution for each party in FL. Instead of removing a single client at a time, the FL-Cohort removes multiple similar clients from FL training at a time.

Table 4.2 presents an overview of the baseline methods used in our thesis, along with details of the datasets utilized.

Table 4.2: Summary of the Baselines used in this thesis.

Method	Datasets	Papers
FedAvg [5]	MHealth, PAMAP2, MNIST, FashionMNIST, CIFAR-10, FEMNIST, CelebA, REALWORLD, HHAR	I, II, IV, V
FedProx [135]	MNIST, FashionMNIST, CIFAR-10	IV
CMFL [136]	MNIST, FashionMNIST, CIFAR-10	IV
CFL [120]	REALWORLD, HHAR	V
Deletion Approach [137]	Synthetic, FEMNIST, CelebA	VI
FL-Cohort [138]	Synthetic, FEMNIST, CelebA	VI

In **Papers I, II, IV** and **V**, naive FedAvg method was used as a benchmark to compare with our proposed FL models and with other FL methods in terms of communication overhead, accuracy and F-measure. FedProx was used in **Paper IV** to compare the communication cost and accuracy of different FL methods. Similarly, in **Paper IV**, CMFL, a method aimed at mitigating communication overhead in FL, was used to compare performance in terms of communication cost and accuracy with various FL methods. In **Paper V**, CFL as a clustered FL algorithm is employed to compare the performance achieved to the proposed FL model. Finally, the deletion approach and the FL-Cohort, which are used to measure the client contribution, are used to compare the performance achieved with our proposed FL models in **Paper VI**. It is important to note that all our proposed FL models are considered an optimized and efficient version of FedAvg.

4.3 Evaluation Measures

The fundamental aspect of any evaluation involves determining what performance means. However, establishing a clear definition of performance is not straightforward due to the numerous measures proposed to evaluate performance found in the literature [139–141]. When a new algorithm is introduced, it is typical to demonstrate its enhancement over other algorithms in a certain aspect. The fundamental question is whether algorithm A is better than algorithm B, or how probable is that

algorithm A yields better results in contrast to algorithm B. In the context of FL tasks, an improved algorithm is often understood as one that achieves more accurate results in a few iterations and/or reduces resource consumption compared to other cutting-edge FL approaches. Our evaluation focuses on the performance of our proposed FL models against some other baseline methods in terms of communication cost, the model's accuracy, energy consumption, battery life of devices, etc. Table 4.3 lists the measures that were used primarily in the studies of our thesis. In FL, computational tasks are distributed among numerous less powerful devices like smartphones, wearables, autonomous vehicles, and others. Given that communication in FL is more resource intensive than computation, minimizing communication is a highly desirable concern. Therefore, the performance in FL is characterized by the highest accuracy achieved after a given number of communications. This communication involves rounds of communication between a server and its clients to exchange models among them. **Papers I** and **IV** proposed FL models which improve the performance of the FL scenario in terms of reducing communication overhead while achieving better F1 score/ accuracy values. The F-measure was used in **Papers I, II,** and **V** as an evaluation measure to compare the performance of our proposed FL methods with other FL methods, such as FedAvg and CFL methods. Energy consumption and battery lifetime measures are used in paper II as part of the multi-criteria evaluation to calculate the score of sensor nodes in the WNs settings. In **Papers IV** and **VI**, one of the measures used to compare the performance of different FL methods is the accuracy of FL used models. In **Paper VI**, Kendall's tau rank correlation was used as a measure to compare the order of clients, inspired by our studies in **Papers I** and **V**, respectively. Specifically, we compare the FL baseline methods by calculating Kendall's tau correlations between the ranking scores of the CA-FL, GP-FL, Deletion method, and FL-Cohort, respectively.

Table 4.3: Evaluation measures used across studies.

Evaluation measures	Papers
Communication overhead	I, IV
F-measure	I, II, V
Energy Consumption	II
Battery Lifetime	II
Accuracy	IV, VI
Kendall's Tau Rank Correlation	VI

4.4 Research Methodology

Two main research methodologies were employed to obtain scientific results that address the research questions described. Firstly, in **Paper III**, the research method-

ology used is a *literature review*, an approach to gather information to improve the understanding of the topic being studied [142, 143]. Our study focuses on reviewing current academic results related to context awareness using AI models, particularly in the context of SNs. In addition, in **Papers I, II, IV, V** and **VI**, we use a research methodology based on *implementation and experimentation* [144]. In each of the studies presented in this thesis, new FL models are introduced and tested through experimentation. A range of experiments are carried out to verify the effectiveness of the algorithms using diverse datasets and baseline methods. This research methodology involves performing experiments to explore particular research questions. Furthermore, this method assesses algorithms in a controlled experimental setting to measure a specific variable.

In **Paper I**, the proposed algorithm, namely the *Cluster Analysis-based FL (CA-FL)* model, has been compared with a state-of-the-art algorithm (FedAvg [5]) using HAR datasets (MHealth [125] and PAMAP2 [126]), to reduce communication overhead between the central server and participants under the IID and Non-IID data settings. The selection of representatives is based on the evaluation of the performance of the local model of each participant. Various experiments are designed and conducted to demonstrate the algorithm's ability to minimize communication overhead in system performance.

Similarly, in **Paper II**, a proposed *Energy-aware Multi-Criteria Federated Learning (EaMC-FL)* model was compared with FedAvg using the same HAR datasets used in **Paper I**, to select only one representative of a cluster for communication with the server with IID and Non-IID data distributions. In contrast to **Paper I**, the selection of the representatives is based on a multi-criteria evaluation for each sensor node (e.g. the local model performance, consumed energy, and battery lifetime). Several experiments are carried out to show that the EaMC-FL Model can decrease the energy used by the edge nodes by reducing the amount of transmitted data in various use cases.

Paper IV is an extension study of **Paper I** in which the proposed algorithm, entitled *Federated Learning via Clustering Optimization (FedCO)* is evaluated on publicly available datasets (MNIST [127], FashionMNIST [128] and CIFAR 10 [129]) and also on LEAF datasets (FEMNIST [130] and CelebA [131]) under IID and Non-IID data. The proposed algorithm is also compared with various state-of-the-art FL methods namely, FedAvg [5], FedProx [135], and CMFL [136]. The results of several experiments demonstrated that the proposed *FedCO* technique outperforms the state-of-the-art FL approaches (i.e. FedAvg, FedProx and CMFL), in minimizing communication overhead and attaining higher accuracy in both IID and Non-IID scenarios.

In **Paper V**, a *group-personalized FL (GP-FL)* has been proposed and evaluated on two real-world HAR data (REALWORLD [132] and HHAR [133]). GP-FL has been compared to FedAvg and CFL. The experiments show that our method outperforms two baseline FL algorithms in terms of both model performance and conver-

gence speed.

Paper VI introduces straightforward and efficient FL approaches that illustrate the evaluation of client behavior during the training phase. This is demonstrated using two established FL models in **Papers I** and **V**, respectively. Our proposed FL models have been compared to the Deletion Approach [137] and FL-Cohort [138] on three LEAF datasets. These LEAF datasets (Synthetic [134], FEMNIST [130] and CelebA [131]) are used to validate the approaches.

4.5 Validity Threats

In this section, we present different types of validity threats that may have arisen for the results of the thesis in four dimensions, including internal, external, construct and conclusion, along with the strategies implemented to address them.

4.5.1 Internal Validity

Internal validity refers to the impact of the experimental setup on the results [145, 146]. In this thesis, the threat of selection bias is presented, which can often be remedied by random sampling [147]. We split the experimental dataset into different sets. Specifically, in **Papers I** and **II**, we used 10 different test sets (cross-validation) of the dataset in the conducted experiments to avoid selection bias. 3-fold cross-validation on each training set was performed in **Paper IV**. In addition, in **Paper V**, we performed 3-fold cross-validation on each experimental dataset. Finally, 3 and 5-fold cross-validations were performed on each experimental dataset for several communication rounds in **Paper VI**. Selection bias is not seen as a concern in **Paper III** as it is a literature review.

4.5.2 External Validity

External validity refers to the extent to which the results of the experiment can be applied or generalized [145, 146] in a different scenario. The experiments carried out in all the included studies are carefully designed to reduce such threats. Although many studies in thesis focus on different FL tasks, such as HAR, image classification, and/or cluster identification, they often use multiple datasets to evaluate the effectiveness of the proposed FL algorithm and prevent results that are specific to a particular scenario. Nevertheless, the limited number of datasets may not suffice to ensure generalizability/ applicability to all real-world settings. All of our studies have used at least two types of datasets for evaluation, except for **Paper III**, which is a review of the literature.

4.5.3 Construct Validity

Construct validity deals with issues surrounding the extent to which the outcomes align with the intended conceptual goals [148]. If the algorithm fails to generate results comparable to the one created during the development stage, these threats could materialize. In order to mitigate these threats, the research group discusses the proposed algorithms and setups prior to commencing the implementation phase. Throughout the implementation stage, regular tests are carried out to verify that the code functions correctly. This practice is essential to prevent the occurrence of run-time errors that are harder to detect than compile-time errors. **Papers I, II** and **IV** employ the partitioning technique k-medoids to group participants into similar groups according to model parameters, necessitating the pre-definition of the parameter k. The SI method was used to determine the optimal number of clusters k. **Paper III** applies the DBSCAN algorithm to categorize keywords from the analyzed studies into clusters of keywords that are semantically related. Although DBSCAN does not necessitate prior knowledge of the number of clusters, it does require the specification of a parameter (eps). Through our experimentation with various eps values, we found that 0.3 resulted in the most well-balanced clustering without any outliers. **Paper V** applies the Markov clustering technique to divide participants with similar empirical probability vectors into similar clusters. In order to assess the quality of clustering, we conducted clustering with varying inflation values. Modularity was computed for each clustering iteration to determine the optimal inflation value for the given graph. In **Paper VI**, SI and modularity computation methods are used to determine the number of clusters for CA-FL and GP-FL, respectively. Similar to commonly used methods in **Papers I, II, IV** and **V**.

4.5.4 Conclusion Validity

Conclusion validity pertains to the efficiency of the study in the handling of the data, the experimental procedures, the evaluation, and the results [148]. In order to mitigate validity threats and limitations in our thesis, we provide a detailed description of the procedures followed to obtain the research results. A detailed description of the design, implementation, setup, and evaluation is provided to provide the necessary understanding. Furthermore, **Papers I-VI** have been through a peer-review procedure and have been published (except **Paper VI**, which is presently under review) in different conferences and journals, which confirms the validity of the experimental approach, analysis, and conclusions utilized in the studies.

5 Results and Analysis

This chapter provides an overview of the results of the thesis. Four main directions have been identified: resource-aware FL, Personalized FL, evaluation client behavior in FL, and the study of edge-based AI for SNs. The results of the thesis are summarized and deliberated in these directions.

5.1 Resource-aware Federated Learning

The results presented in this thesis, which are relevant to the Resource-aware FL direction, are detailed in **Papers I, II, IV, V** and **VI**. Mitigating communication overheads through FL directly ties into resources-aware, hindering the scalability and efficiency of FL systems (see Section 1.1). **Papers I** and **IV** propose a novel FL model to reduce communication overhead for the FL process, namely *Reduce communication overhead of Federated Learning through Clustering Analysis (CA-FL)*, and *FedCO: Communication-Efficient Federated Learning via Clustering Optimization*. It is important to note that **Paper IV** is an extended version of **Paper I**. Specifically, **Paper IV** is inspired by the idea introduced in **Paper I**. In **Paper I**, we have developed a regression model in ML and assessed the CA-FL model utilizing only FedAvg for datasets related to HAR. CA-FL model utilized clustering analysis to reduce FL communication overhead by only sending the most representative updates to the central server. On the other hand, in **Paper IV**, we have improved the initial CA-FL framework by incorporating a dynamic clustering method that decreases communication overhead and speeds up the convergence of the global model. Various datasets and baseline algorithms were used to demonstrate the robustness and effectiveness of our proposed model. The improvements have led to the development of a new version of a DL-based framework called FedCO. Our proposed FedCO evaluates the local updates of the cluster representatives during each communication round, and consequently reallocates certain workers to different clusters. The result of this cluster-updating process is the possibility of clusters appearing or disappearing. Our approach is adept at detecting and managing such situations. Furthermore, it implements a splitting process that conducts additional fine calibration of the clustering for recently uploaded updates. Figure 5.1 demonstrates the properties of the clustering optimization approach during the initial five global communication rounds of the

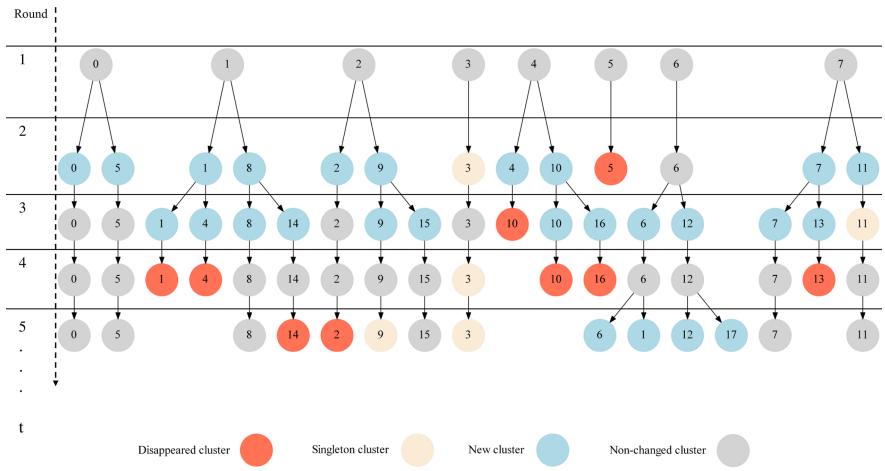


Figure 5.1: The clustering updates in the first five global communication rounds of the proposed FedCO algorithm applied on the Non-IID FashionMNIST dataset. Notice that the number in the circle represents the cluster label. The figure is copied from **Paper IV**.

FedCO algorithm. The cluster optimizations mentioned in **Paper IV** will continue in a similar way for the next communication rounds. Workers' partitions are dynamically adapted in each round of communication to reflect the new local updates from the representatives.

Since each data transfer consumes energy, optimizing the FL process affects network bandwidth and lessens energy expenditures, a crucial consideration for communication networks. Thus, the aspiration for energy-efficient systems complements the drive for resource-aware FL. In FL, the client's computing resources are directly involved in the local training. For this reason, **Paper II** examined the issue of energy consumption and the energy budget of FL edge nodes in SNs. In this regard, a proposed algorithm named *Energy-aware Multi-Criteria Federated Learning (EaMC-FL)* takes into account the balance of the performance of the model and the energy usage of the sensor nodes and compares its effectiveness to the conventional FedAvg algorithm. The EaMC-FL algorithm has been evaluated under six use cases on the same HAR datasets used in **Paper I**. The experimental results indicate that EaMC-FL surpasses FedAvg in terms of total energy consumption, energy budget, and model precision.

Traditional approaches to assessing client contributions require an independent evaluation of each client's contribution outside the initial FL procedure. This leads to increased use of computational resources and longer processing times, as stated above. Furthermore, these methods do not take into account dynamic data during the training process. **Paper VI** introduced measuring contribution through evaluation of client behavior as part of the training process using existing FL models, namely the CA-FL and GP-FL models used in **Paper I** and **V**, respectively. The results of the experiments have demonstrated that our approaches (CA-FL and GP-FL) can accu-

rately evaluate the client's contribution to the overall FL model without significant communication and computation costs.

5.2 Personalized Federated Learning

Although PFL is promising, it does not benefit from the potential for collaborative learning between participants, which presents a great problem due to two aspects. First, PFL cannot learn effectively with a small amount of data on each client device. Furthermore, it fails to leverage device similarity with respect to tasks or data.

Paper V introduces a *Group-Personalized Federated Learning approach for Human Activity Recognition using Cluster Eccentricity Analysis* in the context of HAR applications. The objective of the GP-FL model proposed in this study is to address the challenge of data heterogeneity and to achieve a balance between the global model and local models. Figure 5.2 depicts the various ways to model FL. The conventional FL scenario, presented in the central figure, assumes a federation of decentralized clients, each of which has its own private data. The objective of the GP-FL model proposed in this study is to tackle the challenge of data heterogeneity and achieve a trade-off between the two extreme cases described above.

Clearly, the Group-Personalized FL (GP-FL) algorithm can train multiple global models simultaneously, each corresponding to a group of clients that share similar activity patterns. During every training iteration, the empirical probability vector of each client is updated to reflect the data from their latest batch. Furthermore, the eccentricity analysis of the group [80] is applied to the current grouping of clients. Consequently, in the subsequent round, certain clients might change clusters, or entirely new singleton clusters could appear. The HAR problem is well suited for our FL

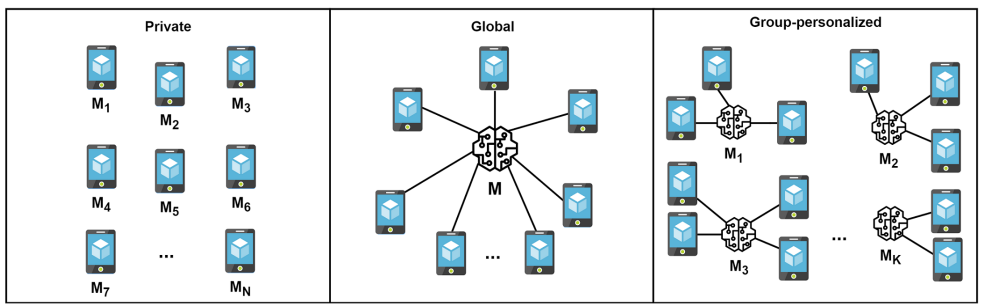


Figure 5.2: Comparison of three distinct FL scenarios: (i) The plot on the left illustrates a setting where every model is trained using the private data of the worker; (ii) The middle plot depicts a situation in which a global model is generated using the models trained by different clients; (iii) The plot on the right demonstrates a setting that considers the similarity among participants and create a global model based on the local models of each group of similar devices. The figure is copied from **Paper V**.

situations since different activities often exhibit common patterns while also being highly unique [149, 150]. The GP-FL algorithm has been evaluated through a set

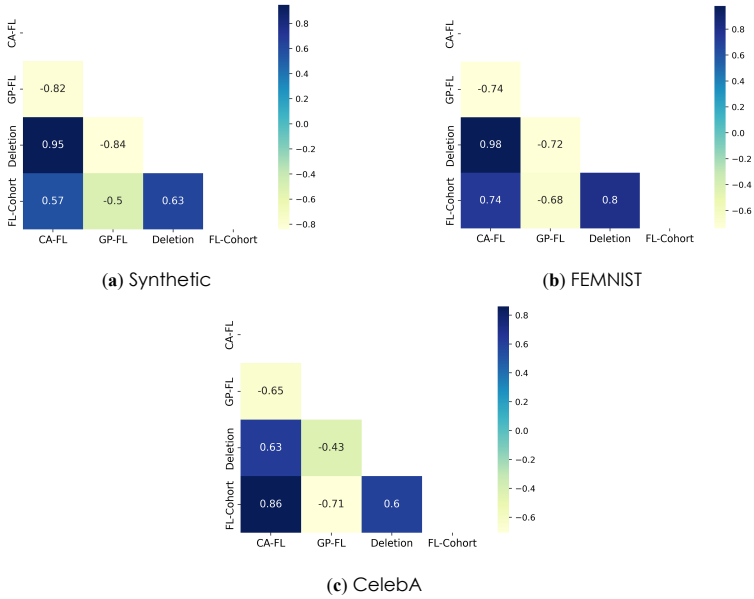
of experiments conducted in the HAR domain. The experimental results show that GP-FL surpasses two baseline FL algorithms (FedAvg and CFL) in terms of both the performance of the model and the speed of convergence.

5.3 Evaluation of Client Behavior

Although the main emphasis of **Papers I** and **V** differed, it was noted that their proposed FL models were capable of assessing the client's contribution. **Paper VI** proposed *Contribution Prediction in Federated Learning via Client Behavior Evaluation* by using these FL models, namely CA-FL and GP-FL models, to demonstrate that client behavior evaluation can be used to measure client contribution. The primary idea involves training a global model with the participation of a specific group of clients while also assessing the client's behavior (e.g., reliable versus unreliable) simultaneously during the training phase.

The CA-FL model outlined in **Paper I** computes the frequency of each client being selected as a representative cluster throughout the training phase. During every training iteration, one client with the best performing model is selected in each cluster. Consequently, each client assigns a score that reflects their level of reliability. A higher score signifies a more significant impact (contribution) on the overall model, and these scores are valuable for classifying clients based on their dependability. In contrast to the CA-FL approach, the GP-FL technique (in **Paper V**) provides an opportunity to detect clients exhibiting unstable behavior while undergoing training. The GP-FL algorithm initially divides the clients into several clusters based on the similarity between their class distributions. This clustering is continuously updated throughout the training process by assessing at each training round the clients' assignment among the clusters, and potentially reassigning some to new clusters. In evaluating client behavior, we calculate the frequency with which each client changes a cluster between clusters during the FL training. This can be seen as an indicator of the instability of the client's data, reflecting the client's lack of unreliability. These efficient FL models can help reduce the resources and time needed to assess the behavior of the clients involved as part of the training process. Our proposed FL models have been evaluated against the Deletion Approach and FL-Cohort using various LEAF datasets. It is important to notice that various criteria can be applied to categorize clients into three (or more) clusters (such as highly dependable clients, clients with a moderate level of dependability, and unreliable clients) based on the scores generated by CA-FL or GP-FL. For instance, dividing thresholds can be established by examining the scores, or binning techniques can be employed.

The Figure 5.3 illustrates the heatmaps displaying Kendall's tau correlation ranking scores for the rankings generated by four methods: CA-FL, GP-FL, Deletion approach, and FL-Cohort, across three different datasets. The stronger the correlation between the two methods, the darker the color, and conversely. These correlations



(a) Synthetic

(b) FEMNIST

(c) CelebA

Figure 5.3: Heatmaps of Kendall's tau ranking scores of CA-FL, GP-FL, Deletion approach, and FL-Cohort on Synthetic, FEMNIST and CelebA datasets. The figure is copied from **Paper VI**.

help us to determine the similarity between the rankings produced by our two FL algorithms and the two baseline methods. For the Synthetic dataset 5.3(a), it is evident that there is a highly positive correlation of 0.95 between CA-FL and the deletion approach. This indicates that our method (CA-FL), which demands significantly fewer computational resources, can assess the clients' input in a comparable manner to the deletion method. In the FEMNIST dataset, as depicted in Figure 5.3(b), as one can see that CA-FL exhibits stronger correlation with the Deletion method compared to its correlation with the FL-Cohort, similar to the Synthetic dataset. On the other hand, within the CelebA dataset, CA-FL exhibits a higher correlation score with the FL-Cohort compared to the Deletion method, as shown in Figure 5.3(c).

5.4 Edge-based Artificial Intelligence for Sensor Networks

Upon reviewing the research questions of previous studies, it became clear that identifying the current AI-based solutions in context-aware edge intelligence systems posed a significant challenge. Hence, in **Paper III**, we conduct an in-depth analysis of the literature on context-aware edge-based AI models that leverage sensor technology, uncovering their applications, related challenges, and motivations for adopting AI solutions, along with identifying existing research gaps. Another aspect of this research is the use of a semantic-based method to identify subjects relevant to

the survey. In particular, the method is based on the examination of the keywords of the articles. Initially, all unique keywords from the extracted articles are collected. The total count of unique keywords is 637. Subsequently, this count is reduced to focus on the keywords that appear most frequently. Each keyword is assigned a score based on how often it appears among the keywords in the articles. Then, any

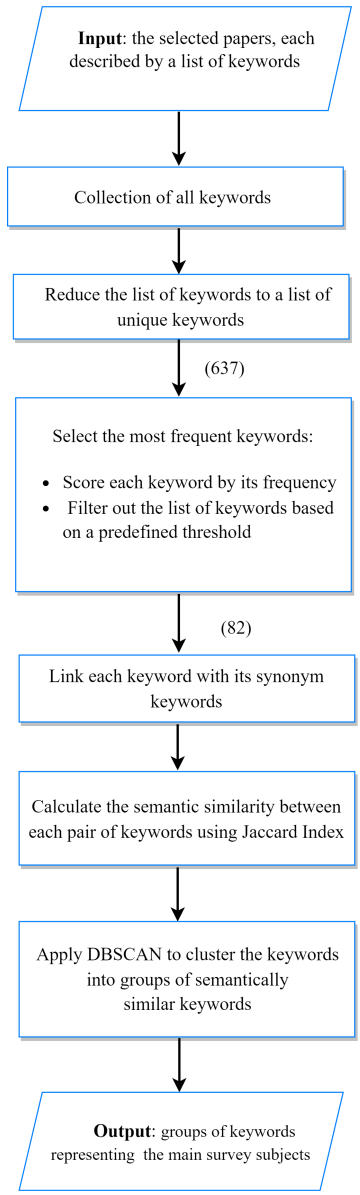


Figure 5.4: Flowchart describing the different steps of the semantic-aware approach applied to identify the main subjects covered by the included papers. The figure is copied from **Paper III**.

keywords with scores lower than the defined threshold value are excluded, resulting

in only the top 82 most frequent keywords being retained. This method relies on assessing the semantic similarity among keywords to identify the main research or application topics covered in the survey. Figure 5.4 shows the flowchart of the procedure to identify the primary survey subjects. The latter precisely identifies eleven primary research topics supported by the articles included in the study. Furthermore, the relative percentage of cluster size generated by using DBSCAN with eps value of 0.3 on the 82 most frequent keywords. The parameter eps determines how close keywords should be to each other to be classified as part of a cluster. Different values of eps were experimented with, and 0.3 produced the most evenly distributed clusters with no outliers. The selected articles were analyzed regarding identified subjects to get a deeper understanding of the limitations and gaps in this study. These aspects are examined from various perspectives to address five main research questions. Potential future research directions are also deliberated.

AI, ML and DL, edge computing and smart monitoring, smart healthcare, and smart and wearable devices are eleven topics that have been recognized. In the analysis carried out, we have also discovered that healthcare, smart cities, autonomous driving, environmental monitoring, and transportation are the top five domains of application. Enhancing recognition quality, optimizing management effectiveness, improving quality of service (QoS) and efficiency, and guaranteeing higher security are the primary motivations for implementing intelligent applications in context-aware systems.

The included papers have explored a range of AI-based solutions. Unsupervised and semi-supervised algorithms, along with transfer learning techniques, are highlighted as areas that have not received significant attention from researchers in many context-aware scenarios. Additionally, a promising collaborative framework, such as FL has not been thoroughly investigated. Reviews also lack research on location-based services, indicating a need for more studies that focus more on these issues.

5.5 Summary

In this section, all the research questions formulated in this thesis are stated and answered.

RQ 1: *How we can develop FL models that reduce resource consumption without sacrificing the model performance?*
The reduction of resource consumption we have shown to be tackled by decreasing the number of data transferred to the server while maintaining the model performance. In **Paper I**, a new FL model named CA-FL based on clustering analysis is introduced, aiming to minimize communication overhead by interacting only with a cluster representative. Thus improving the speed of learning convergence. Specifically, we

first cluster clients into groups based on the similarity measures of model parameters. Based on the ED similarity. Then, we select one representative of each cluster that realizes higher performance in terms of F1-score to communicate with the server. Subsequently, we perform a clustering evaluation of the representatives based on the SI.

Paper II proposed a multi-criteria client FL model in WNs, namely EaMC-FL. Specifically, we first split the clients into groups in a similar way used in **Paper I** and then select representatives of groups according to the multi-criteria evaluation of clients. Then, we define a client selection metric informed by several criteria, such as client resources and model performance. In the same way as described in **Paper I**, only a representative of a cluster interacts with the server to train a unified model. At each subsequent iteration, the SI also updates the grouping of clients by evaluating whether representatives are still closely tied to their respective clusters.

In **Paper IV**, the CA-FL method was further developed by introducing a clustering optimization technique to improve model aggregation and reduce communication expenses. This was achieved by applying clustering optimization for each representative. The proposed FedCO method used updating clusters by iteratively assessing and splitting clusters when needed to enhance the client partitioning. These FL models in **Papers I, II** and **IV** use only group representatives during their FL process, making them resource efficient.

RQ 2: *How the clients' behaviour can be efficiently evaluated during the FL process?*
We addressed this research question in **Paper VI** by proposing a way to measure contribution in FL by evaluating client behavior. The proposed FL models are based on the FL models proposed in **Papers I** and **V**. Even though **Papers I** and **V** had different main focus, it was observed that these models (CA-FL and GP-FL) could be applied to assess the client's contribution. In the previous CA-FL model, the evaluation involves determining the frequency at which each client is selected as a cluster representative to participate in generating the shared model. This process can be considered as an indicator of the reliability of the client's data. In the GP-FL model, we determine the frequency with which each client changes clusters during FL training, indicating the client's instability, and suggesting unreliability.

RQ 3: *How we can personalize FL models to achieve robust model performance?*
A group-personalized FL (GP-FL) model is proposed for investigating this research question in **Paper V**. The GP-FL proposed in this study aims to address the challenge of data heterogeneity and achieve a balance between the global joint model and local models. Three evaluations are compared using GP-FL: a single global performance assessed by computing the accuracy or F1 score generated by the joint model on the individual device's data. Group performance evaluates the precision or F1 score attained by the global model of each group on the individual data of each client within

the group. The personal's performance is assessed by calculating the accuracy or F1 score achieved by the client's local model with its private data. This study introduces a method based on grouping for personalized FL within the context of HAR. Specifically, the proposed GP-FL algorithm constructs multiple global ML models, each iteratively trained on a dynamic group of clients with homogeneous class probability estimations.

RQ 4: *What AI-based solutions are underrepresented in the recent state-of-the-art of context-aware edge intelligence systems?*
In this thesis, we have presented a literature review of the recent development of context-aware edge-based AI methods in SNs. In **Paper III**, an extensive review of the literature is carried out to explore the applications, associated challenges, and motivations to implement context-aware AI solutions in SNs. Furthermore, our goal is to identify any current research gaps. Several AI-based solutions have been studied in the included papers. Interestingly, we found that a collaborative AI framework such as FL has not been adequately examined. Another aspect of this research involves employing a semantic-based method to identify subjects relevant to the survey. In particular, the method is based on analyzing the keywords of the articles.

6 Conclusion and Future Directions

6.1 Conclusion

This thesis introduced new resource-aware FL and personalized models employing cluster analysis to enhance the effectiveness and robustness of FL in the context of diverse and dynamic data. In particular, four primary directions have been identified, including resource-aware FL, personalized FL, evaluation of client behavior in FL, and exploration of edge-based AI for SNs.

Traditional FL approaches, despite being privacy-preserving, consume high computational resources. We develop resource-efficient approaches within an FL system considering the diversity of data quality, computing, and communication capabilities of FL clients. We further personalize FL models to achieve robust model performance, effectively managing heterogeneity in data, and adapting to dynamic data within conventional FL. In addition, we effectively analyze the client's behavior throughout the FL process to evaluate the client's contribution. This approach requires fewer resources than traditional FL methods and is commonly performed as part of FL procedures.

Finally, to determine underrepresented AI-based solutions in recent context-aware edge intelligence systems, we conducted a thorough literature review to identify AI-based solutions focusing on context-aware systems, particularly within SNs.

6.2 Future Directions

In our future work, we will build on the advances made in previous studies that introduced novel resource-conscious FL models. To achieve this aim, we will explore different resource-aware distributed (federated) AI methods to develop accurate and reliable asset fault detection and diagnosis solutions. Horizontal FL, vertical FL, and transfer FL are the focus of interest in our future work. Several potential directions for future research are outlined below.

- We plan to develop accurate and energy-efficient vertical FL models that can exploit different types of inaccurate data collected from various IoUT (e.g.,

cables, devices, machines, sensors) and in IIoT contexts. That will be done in a setting containing labelled and unlabelled data to detect and diagnose faults, thereby supporting the decisions regarding maintenance activities.

- We plan to develop accurate, energy-efficient horizontal FL models based on multi-source data to identify and detect potential faults of critical assets on the ground. We will investigate multiple types of sensor readings (e.g., temperature, acoustic samples, etc.) that will be collected and analysed by our horizontal FL models.
- We plan to develop a fault prediction FL model built on multiple data sources (e.g., distributed sensors and drones) to facilitate maintenance planning and guarantee the continuous operation of industrial production sites, underwater vehicles, and assets. In addition, employing FL transfer learning techniques will be investigated in this project, allowing for transferring knowledge and pre-trained FL models from IoUT devices to new devices with different data domains.

Bibliography

[1] K. M. Hazelwood, S. Bird, D. M. Brooks, S. Chintala, U. Diril, D. Dzhulgakov, M. Fawzy, B. Jia, Y. Jia, A. Kalro, J. Law, K. Lee, J. Lu, P. Noordhuis, M. Smelyanskiy, L. Xiong, and X. Wang. "Applied Machine Learning at Facebook: A Datacenter Infrastructure Perspective". In: *2018 IEEE International Symposium on High Performance Computer Architecture (HPCA)* (2018), pp. 620–629.

[2] X. Qiu, T. Parcollet, D. J. Beutel, T. Topal, A. Mathur, and N. D. Lane. "A first look into the carbon footprint of federated learning". In: *ArXiv* abs/2010.06537 (2020).

[3] A. Deshpande, C. Guestrin, S. Madden, J. M. Hellerstein, and W. Hong. "Model-based approximate querying in sensor networks". In: *The VLDB Journal* 14 (2005), pp. 417–443.

[4] F. Bonomi, R. A. Milito, J. Zhu, and S. Addepalli. "Fog computing and its role in the internet of things". In: *MCC '12*. 2012.

[5] H. B. M. et al. "Communication-Efficient Learning of Deep Networks from Decentralized Data". In: *International Conference on Artificial Intelligence and Statistics*. 2016.

[6] J. Konecný, H. B. McMahan, D. Ramage, and P. Richtárik. "Federated Optimization: Distributed Machine Learning for On-Device Intelligence". In: *ArXiv* abs/1610.02527 (2016).

[7] Y. Zhang, D. Ramage, Z. Xu, Y. Zhang, S. Zhai, and P. Kairouz. "Private Federated Learning in Gboard". In: *ArXiv* abs/2306.14793 (2023).

[8] Q. Yang, Y. Liu, T. Chen, and Y. Tong. "Federated Machine Learning: Concept and Applications". In: *arXiv: Artificial Intelligence* (2019).

[9] B. S. Guendouzi, S. Ouchani, H. E. Assaad, and M. E. Zaher. "A systematic review of federated learning: Challenges, aggregation methods, and development tools". In: *J. Netw. Comput. Appl.* 220 (2023), p. 103714.

[10] M. hany mahmoud, A. Albaseer, M. M. Abdallah, and N. Al-Dhahir. “Federated Learning Resource Optimization and Client Selection for Total Energy Minimization Under Outage, Latency, and Bandwidth Constraints With Partial or No CSI”. In: *IEEE Open Journal of the Communications Society* 4 (2023), pp. 936–953.

[11] P. e. a. Kairouz. “Advances and Open Problems in Federated Learning”. In: *Found. Trends Mach. Learn.* 14 (2019), pp. 1–210. URL: https://api.semanticscholar.org/CorpusID:209202606.

[12] O. Shahid, S. Pouriyeh, R. M. Parizi, Q. Z. Sheng, G. Srivastava, and L. Zhao. “Communication Efficiency in Federated Learning: Achievements and Challenges”. In: *ArXiv* abs/2107.10996 (2021).

[13] S. Huang, W. Shi, Z. Xu, I. W.-H. Tsang, and J. Lv. “Efficient federated multi-view learning”. In: *Pattern Recognit.* 131 (2022), p. 108817.

[14] M. Xu, J. Liu, Y. Liu, F. X. Lin, Y. Liu, and X. Liu. “A First Look at Deep Learning Apps on Smartphones”. In: *The World Wide Web Conference* (2018).

[15] K. Simonyan and A. Zisserman. “Very Deep Convolutional Networks for Large-Scale Image Recognition”. In: *CoRR* abs/1409.1556 (2014).

[16] C. Wu, F. Wu, L. Lyu, Y. Huang, and X. Xie. “Communication-efficient federated learning via knowledge distillation”. In: *Nature Communications* 13 (2021).

[17] J. Mills, J. Hu, and G. Min. “Communication-Efficient Federated Learning for Wireless Edge Intelligence in IoT”. In: *IEEE Internet of Things Journal* 7 (2020), pp. 5986–5994.

[18] H. McMahan, E. Moore, D. Ramage, and B. A. y Arcas. “Federated Learning of Deep Networks using Model Averaging”. In: *arXiv preprint arXiv:1602.05629* (2016).

[19] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth. “Practical Secure Aggregation for Privacy-Preserving Machine Learning”. In: *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security* (2017).

[20] Q. Li, Y. Diao, Q. Chen, and B. He. “Federated Learning on Non-IID Data Silos: An Experimental Study”. In: *2022 IEEE 38th International Conference on Data Engineering (ICDE)* (2021), pp. 965–978.

[21] 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”. In: *Inf. Fusion* 88 (2021), pp. 263–280.

[22] S. P. Karimireddy, S. Kale, M. Mohri, S. J. Reddi, S. U. Stich, and A. T. Suresh. “SCAFFOLD: Stochastic Controlled Averaging for Federated Learning”. In: *International Conference on Machine Learning*. 2019.

[23] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang. “On the Convergence of FedAvg on Non-IID Data”. In: *ArXiv* abs/1907.02189 (2019).

[24] Y. Zhao et al. “Federated Learning with Non-IID Data”. In: *ArXiv* 1806.00582 (2018).

[25] A. K. Sahu, T. Li, M. Sanjabi, M. Zaheer, A. Talwalkar, and V. Smith. “Federated Optimization in Heterogeneous Networks”. In: *arXiv: Learning* (2018).

[26] V. Kulkarni, M. Kulkarni, and A. Pant. “Survey of Personalization Techniques for Federated Learning”. In: *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)* (2020), pp. 794–797.

[27] C. T. Dinh, N. H. Tran, and T. D. Nguyen. “Personalized Federated Learning with Moreau Envelopes”. In: *ArXiv* abs/2006.08848 (2020).

[28] A. Fallah, A. Mokhtari, and A. E. Ozdaglar. “Personalized Federated Learning: A Meta-Learning Approach”. In: *ArXiv* abs/2002.07948 (2020).

[29] H. B. M. et al. “Communication-Efficient Learning of Deep Networks from Decentralized Data”. In: *AISTATS*. 2017.

[30] W. Bao, C. Wu, S. Guleng, J. Zhang, K.-l. A. Yau, and Y. Ji. “Edge computing-based joint client selection and networking scheme for federated learning in vehicular IoT”. In: *China Communications* 18 (2021), pp. 39–52.

[31] M. Hu, D. Wu, Y. Zhou, X. Chen, and M. Chen. “Incentive-Aware Autonomous Client Participation in Federated Learning”. In: *IEEE Transactions on Parallel and Distributed Systems* PP (2022), pp. 1–1.

[32] Q. Wu, K. He, and X. Chen. “Personalized Federated Learning for Intelligent IoT Applications: A Cloud-Edge Based Framework”. In: *IEEE Open Journal of the Computer Society* 1 (2020), pp. 35–44.

[33] H. Ren, J. Deng, and X. Xie. “Privacy Preserving Text Recognition with Gradient-Boosting for Federated Learning”. In: *ArXiv* abs/2007.07296 (2020).

[34] Z. Iqbal and H. Y. Chan. “Concepts, Key Challenges and Open Problems of Federated Learning”. In: *International Journal of Engineering* (2021).

[35] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith. “Federated Learning: Challenges, Methods, and Future Directions”. In: *IEEE Signal Processing Magazine* 37 (2019), pp. 50–60.

[36] R. Gosselin, L. Vieu, F. Loukil, and A. Benoit. “Privacy and Security in Federated Learning: A Survey”. In: *Applied Sciences* (2022).

[37] L. Melis, C. Song, E. D. Cristofaro, and V. Shmatikov. “Exploiting Unintended Feature Leakage in Collaborative Learning”. In: *2019 IEEE Symposium on Security and Privacy (SP)* (2018), pp. 691–706.

[38] S. Luo, X. Chen, Q. Wu, Z. Zhou, and S. Yu. “HFEL: Joint Edge Association and Resource Allocation for Cost-Efficient Hierarchical Federated Edge Learning”. In: *IEEE Transactions on Wireless Communications* 19 (2020), pp. 6535–6548.

[39] X. Zhang, Z. Chang, T. Hu, W. Chen, X. Zhang, and G. Min. “Vehicle Selection and Resource Allocation for Federated Learning-Assisted Vehicular Network”. In: *IEEE Transactions on Mobile Computing* (2023).

[40] T. T. Vu, D. T. Ngo, N. H. Tran, H. Q. Ngo, M. N. Dao, and R. H. Middleton. “Cell-Free Massive MIMO for Wireless Federated Learning”. In: *IEEE Transactions on Wireless Communications* 19 (2019), pp. 6377–6392.

[41] J. Sun, A. Li, L. DiValentin, A. Hassanzadeh, Y. Chen, and H. H. Li. “FL-WBC: Enhancing Robustness against Model Poisoning Attacks in Federated Learning from a Client Perspective”. In: *Neural Information Processing Systems*. 2021.

[42] A. N. Bhagoji, S. Chakraborty, P. Mittal, and S. B. Calo. “Analyzing Federated Learning through an Adversarial Lens”. In: *International Conference on Machine Learning*. 2018.

[43] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov. “How To Backdoor Federated Learning”. In: *ArXiv* abs/1807.00459 (2018).

[44] Y. Wen, J. Geiping, L. H. Fowl, M. Goldblum, and T. Goldstein. “Fishing for User Data in Large-Batch Federated Learning via Gradient Magnification”. In: *ArXiv* abs/2202.00580 (2022).

[45] H. B. M. et al. “Communication-efficient learning of deep networks from decentralized data”. In: *arXiv preprint arXiv:1602.05629* (2016).

[46] D. Jatain, V. Singh, and N. Dahiya. “A contemplative perspective on federated machine learning: Taxonomy, threats & vulnerability assessment and challenges”. In: *J. King Saud Univ. Comput. Inf. Sci.* 34 (2021), pp. 6681–6698.

[47] L. Yang, Z. Meng, and L. Wang. “A multi-layer two-dimensional convolutional neural network for sentiment analysis”. In: *Int. J. Bio Inspired Comput.* 19 (2022), pp. 97–107.

[48] G. Drainakis, K. V. Katsaros, P. Pantazopoulos, V. Sourlas, and A. J. Amditis. “Federated vs. Centralized Machine Learning under Privacy-elastic Users: A Comparative Analysis”. In: *2020 IEEE 19th International Symposium on Network Computing and Applications (NCA)* (2020), pp. 1–8.

[49] T. Kraska, A. Talwalkar, J. C. Duchi, R. Griffith, M. J. Franklin, and M. I. Jordan. “MLbase: A Distributed Machine-learning System”. In: *Conference on Innovative Data Systems Research*. 2013.

[50] J. Konecný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon. “Federated Learning: Strategies for Improving Communication Efficiency”. In: *ArXiv* abs/1610.05492 (2016).

[51] A. Li, L. Zhang, J. Wang, F. Han, and X. Li. “Privacy-Preserving Efficient Federated-Learning Model Debugging”. In: *IEEE Transactions on Parallel and Distributed Systems* 33 (2022), pp. 2291–2303.

[52] M. N. Fekri, K. Grolinger, and S. Mir. “Distributed load forecasting using smart meter data: Federated learning with Recurrent Neural Networks”. In: *International Journal of Electrical Power & Energy Systems* (2021).

[53] J. X. et al. “Ternary Compression for Communication-Efficient Federated Learning”. In: *IEEE Transactions on Neural Networks and Learning Systems* 33 (2020), pp. 1162–1176.

[54] J. Liu and Y. Jin. “Multi-objective Search of Robust Neural Architectures against Multiple Types of Adversarial Attacks”. In: *Neurocomputing* 453 (2021), pp. 73–84.

[55] N. Zeng, D. Song, H. Li, Y. You, Y. Liu, and F. E. Alsaadi. “A competitive mechanism integrated multi-objective whale optimization algorithm with differential evolution”. In: *Neurocomputing* 432 (2021), pp. 170–182.

[56] H. Zhu, J. Xu, S. Liu, and Y. Jin. “Federated Learning on Non-IID Data: A Survey”. In: *ArXiv* abs/2106.06843 (2021).

[57] W. Zhang, X. Wang, P. Zhou, W. Wu, and X. Zhang. “Client Selection for Federated Learning With Non-IID Data in Mobile Edge Computing”. In: *IEEE Access* 9 (2021), pp. 24462–24474.

[58] T.-C. Chiu, Y.-Y. Shih, A.-C. Pang, C.-S. Wang, W. Weng, and C.-T. Chou. “Semisupervised Distributed Learning With Non-IID Data for AIoT Service Platform”. In: *IEEE Internet of Things Journal* 7 (2020), pp. 9266–9277.

[59] E. E. Absalom, A. M. Ikotun, O. N. Oyelade, L. M. Abualigah, J. O. Agushaka, C. I. Eke, and A. A. Akinyelu. “A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects”. In: *Eng. Appl. Artif. Intell.* 110 (2022), p. 104743.

[60] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu. “An Efficient k-Means Clustering Algorithm: Analysis and Implementation”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 24 (2002), pp. 881–892.

[61] W. Xiao and J. Hu. “A Survey of Parallel Clustering Algorithms Based on Spark”. In: *Sci. Program.* 2020 (2020), 8884926:1–8884926:12.

[62] A. K. Jain, M. N. Murty, and P. J. Flynn. “Data clustering: a review”. In: *ACM Comput. Surv.* 31 (1999), pp. 264–323.

[63] P. Arora and S. Varshney. “Analysis of K-Means and K-Medoids Algorithm For Big Data”. In: *Procedia Computer Science* 78 (2016), pp. 507–512.

[64] C. Briggs, Z. Fan, and P. András. “Federated learning with hierarchical clustering of local updates to improve training on non-IID data”. In: *2020 International Joint Conference on Neural Networks (IJCNN)* (2020), pp. 1–9.

[65] Y. Kim, E. A. Hakim, J. Haraldson, H. Eriksson, J. M. B. da Silva, and C. Fischione. “Dynamic Clustering in Federated Learning”. In: *ICC 2021 - IEEE International Conference on Communications* (2020), pp. 1–6.

[66] Y. Xiao, H.-B. Li, and Y.-p. Zhang. “DBGSA: A Novel Data Adaptive Bregman Clustering Algorithm”. In: *ArXiv* abs/2307.14375 (2023).

[67] J. B. MacQueen. “Some methods for classification and analysis of multivariate observations”. In: *In Lucien M. Le Cam and Jerzy Neyman, editors, Proceedings of the Berkley symposium on mathematical statistics and probability* 1 (1967), pp. 281–297.

[68] S. van Dongen. “Graph clustering by flow simulation”. In: 2000.

[69] H.-S. Park and C.-H. Jun. “A simple and fast algorithm for K-medoids clustering”. In: *Expert Syst. Appl.* 36 (2009), pp. 3336–3341.

[70] A. Gordon. “Measures of similarity and dissimilarity”. In: 1999.

[71] D. B. Bisandu, R. Prasad, and M. M. Liman. “Data clustering using efficient similarity measures”. In: *Journal of Statistics and Management Systems* 22 (2019), pp. 901–922.

[72] S. kiran Vangipuram and R. Appusamy. “A SURVEY ON SIMILARITY MEASURES AND MACHINE LEARNING ALGORITHMS FOR CLASSIFICATION AND PREDICTION”. In: *International Conference on Data Science, E-learning and Information Systems 2021* (2021).

[73] X. Wang, A. A. Mueen, H. Ding, G. Trajcevski, P. Scheuermann, and E. J. Keogh. “Experimental comparison of representation methods and distance measures for time series data”. In: *Data Mining and Knowledge Discovery* 26 (2010), pp. 275–309.

[74] P. Jaccard. “Étude comparative de la distribution florale dans une portion des Alpes et du Jura”. In: *Bulletin del la Société Vaudoise des Sciences Naturelles* (1901).

[75] S. Kolouri et al. “Optimal Mass Transport: Signal processing and machine-learning applications”. In: *IEEE Signal Processing Magazine* 34 (2017), pp. 43–59.

[76] P. Rousseeuw. “Silhouettes: a graphical aid to the interpretation and validation of cluster analysis”. In: *Journal of Computational and Applied Mathematics* 20 (1987), pp. 53–65.

[77] O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J. M. Pérez, and I. Perona. “An extensive comparative study of cluster validity indices”. In: *Pattern Recognit.* 46 (2013), pp. 243–256.

[78] M. Brun, C. Sima, J. Hua, J. Lowey, B. Carroll, E. Suh, and E. R. Dougherty. “Model-based evaluation of clustering validation measures”. In: *Pattern Recognit.* 40 (2007), pp. 807–824.

[79] C. G. Bezerra, B. S. J. Costa, L. A. Guedes, and P. P. Angelov. “A new evolving clustering algorithm for online data streams”. In: *2016 IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS)* (2016), pp. 162–168.

[80] P. Angelov. “Anomaly detection based on eccentricity analysis”. In: *2014 IEEE Symposium on Evolving and Autonomous Learning Systems (EALS)*. 2014, pp. 1–8. DOI: 10.1109/EALS.2014.7009497.

[81] C. G. Bezerra et al. “An evolving approach to data streams clustering based on typicality and eccentricity data analytics”. In: *Information Sciences* 518 (2020), pp. 13–28. ISSN: 0020-0255. DOI: https://doi.org/10.1016/j.ins.2019.12.022. URL: https://www.sciencedirect.com/science/article/pii/S0020025519311363.

[82] J. G. Saw et al. “Chebyshev Inequality With Estimated Mean and Variance”. In: *The American Statistician* 38 (1984), pp. 130–132.

[83] I. Škrjanc et al. “Evolving fuzzy and neuro-fuzzy approaches in clustering, regression, identification, and classification: A Survey”. In: *Inf. Sci.* 490 (2019), pp. 344–368.

[84] W. Chen, S. Horváth, and P. Richtárik. “Optimal Client Sampling for Federated Learning”. In: *Trans. Mach. Learn. Res.* 2022 (2020).

[85] M. Mitzenmacher. “The Power of Two Choices in Randomized Load Balancing”. In: *IEEE Trans. Parallel Distributed Syst.* 12 (2001), pp. 1094–1104.

[86] H. T. Nguyen, V. Sehwag, S. Hosseinalipour, C. G. Brinton, M. Chiang, and H. V. Poor. “Fast-Convergent Federated Learning”. In: *IEEE Journal on Selected Areas in Communications* 39 (2020), pp. 201–218.

[87] Z. C. et al. “Dynamic Attention-based Communication-Efficient Federated Learning”. In: *ArXiv* abs/2108.05765 (2021).

[88] R. Balakrishnan, T. Li, T. Zhou, N. Himayat, V. Smith, and J. A. Bilmes. "Diverse Client Selection for Federated Learning via Submodular Maximization". In: *International Conference on Learning Representations*. 2022.

[89] S. Abdulrahman, H. Tout, A. Mourad, and C. Talhi. "FedMCCS: Multicriteria Client Selection Model for Optimal IoT Federated Learning". In: *IEEE Internet of Things Journal* 8 (2021), pp. 4723–4735.

[90] L. Wang, W. Wang, and B. Li. "CMFL: Mitigating Communication Overhead for Federated Learning". In: *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)* (2019), pp. 954–964.

[91] T. H. T. Le, N. H. Tran, Y. K. Tun, M. N. H. Nguyen, S. R. Pandey, Z. Han, and C. S. Hong. "An Incentive Mechanism for Federated Learning in Wireless Cellular Networks: An Auction Approach". In: *IEEE Transactions on Wireless Communications* 20 (2020), pp. 4874–4887.

[92] J. Zhang, Y. Wu, and R. Pan. "Incentive Mechanism for Horizontal Federated Learning Based on Reputation and Reverse Auction". In: *Proceedings of the Web Conference 2021* (2021).

[93] F. S. et al. "Robust and Communication-Efficient Federated Learning From Non-i.i.d. Data". In: *IEEE Transactions on Neural Networks and Learning Systems* 31 (2019), pp. 3400–3413.

[94] K. Ozkara, N. Singh, D. Data, and S. N. Diggavi. "QuPeD: Quantized Personalization via Distillation with Applications to Federated Learning". In: *Neural Information Processing Systems*. 2021.

[95] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic. "QSGD: Communication-Efficient SGD via Gradient Quantization and Encoding". In: *Neural Information Processing Systems*. 2016.

[96] W. Wen, C. Xu, F. Yan, C. Wu, Y. Wang, Y. Chen, and H. Li. "TernGrad: Ternary Gradients to Reduce Communication in Distributed Deep Learning". In: *NIPS*. 2017.

[97] Y. He, M. Zenk, and M. Fritz. "CosSGD: Nonlinear Quantization for Communication-efficient Federated Learning". In: *ArXiv* abs/2012.08241 (2020).

[98] Y. Ren, Y. Cao, C. Ye, and X. Cheng. "Two-layer accumulated quantized compression for communication-efficient federated learning: TLAQC". In: *Scientific Reports* 13 (2023).

[99] A. F. Aji and K. Heafield. "Sparse Communication for Distributed Gradient Descent". In: *arXiv preprint arXiv:1704.05021* (2017).

[100] N. Dryden, T. Moon, S. A. Jacobs, and B. C. V. Essen. “Communication Quantization for Data-Parallel Training of Deep Neural Networks”. In: *2016 2nd Workshop on Machine Learning in HPC Environments (MLHPC)* (2016), pp. 1–8.

[101] T. Chen, G. B. Giannakis, T. Sun, and W. Yin. “LAG: Lazily Aggregated Gradient for Communication-Efficient Distributed Learning”. In: *Neural Information Processing Systems*. 2018.

[102] H. Wang, M. Yurochkin, Y. Sun, D. Papailiopoulos, and Y. Khazaeni. “Federated Learning with Matched Averaging”. In: *ArXiv* abs/2002.06440 (2020).

[103] J. Liu, J. H. Wang, C. Rong, Y. Xu, T. Yu, and J. Wang. “FedPA: An adaptively partial model aggregation strategy in Federated Learning”. In: *Comput. Networks* 199 (2021), p. 108468.

[104] X. Wu, Z. Liang, and J. Wang. “FedMed: A Federated Learning Framework for Language Modeling”. In: *Sensors (Basel, Switzerland)* 20 (2020).

[105] M. Asad, A. Moustafa, and M. Aslam. “CEEP-FL: A comprehensive approach for communication efficiency and enhanced privacy in federated learning”. In: *Appl. Soft Comput.* 104 (2021), p. 107235.

[106] A. Z. Tan et al. “Towards Personalized Federated Learning”. In: *IEEE transactions on neural networks and learning systems* PP (2021).

[107] Y. Mei, B. Guo, D. Xiao, and W. Wu. “FedVF: Personalized Federated Learning Based on Layer-wise Parameter Updates with Variable Frequency”. In: *2021 IEEE International Performance, Computing, and Communications Conference (IPCCC)* (2021), pp. 1–9.

[108] X. Ma, J. Zhang, S. Guo, and W. Xu. “Layer-wised Model Aggregation for Personalized Federated Learning”. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), pp. 10082–10091.

[109] M. G. Arivazhagan, V. Aggarwal, A. K. Singh, and S. Choudhary. “Federated Learning with Personalization Layers”. In: *ArXiv* abs/1912.00818 (2019).

[110] X. Ni, X. Shen, and H. Zhao. “Federated optimization via knowledge codistillation”. In: *Expert Syst. Appl.* 191 (2021), p. 116310.

[111] E. Jeong and M. Kountouris. “Personalized Decentralized Federated Learning with Knowledge Distillation”. In: *ICC 2023 - IEEE International Conference on Communications* (2023), pp. 1982–1987.

[112] J. Jang, H. Ha, D. Jung, and S. Yoon. “FedClassAvg: Local Representation Learning for Personalized Federated Learning on Heterogeneous Neural Networks”. In: *Proceedings of the 51st International Conference on Parallel Processing* (2022).

[113] C. Li, G. Li, and P. K. Varshney. “Decentralized Federated Learning via Mutual Knowledge Transfer”. In: *IEEE Internet of Things Journal* 9 (2020), pp. 1136–1147.

[114] T. Wan, W. Cheng, D. Luo, W. Yu, J. Ni, L. Tong, H. Chen, and X. Zhang. “Personalized Federated Learning via Heterogeneous Modular Networks”. In: *2022 IEEE International Conference on Data Mining (ICDM)* (2022), pp. 1197–1202.

[115] V. Smith, C.-K. Chiang, M. Sanjabi, and A. Talwalkar. “Federated Multi-Task Learning”. In: *Neural Information Processing Systems*. 2017.

[116] L. Corinzia and J. M. Buhmann. “Variational Federated Multi-Task Learning”. In: *ArXiv* abs/1906.06268 (2019).

[117] Y. Huang, L. Chu, Z. Zhou, L. Wang, J. Liu, J. Pei, and Y. Zhang. “Personalized Cross-Silo Federated Learning on Non-IID Data”. In: *AAAI Conference on Artificial Intelligence*. 2020.

[118] F. Hanzely and P. Richtárik. “Federated Learning of a Mixture of Global and Local Models”. In: *ArXiv* abs/2002.05516 (2020).

[119] Y. Deng, M. M. Kamani, and M. Mahdavi. “Adaptive Personalized Federated Learning”. In: *ArXiv* abs/2003.13461 (2020).

[120] F. Sattler, K.-R. Müller, and W. Samek. “Clustered Federated Learning: Model-Agnostic Distributed Multitask Optimization Under Privacy Constraints”. In: *IEEE Transactions on Neural Networks and Learning Systems* 32 (2019), pp. 3710–3722.

[121] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran. “An Efficient Framework for Clustered Federated Learning”. In: *IEEE Transactions on Information Theory* 68 (2020), pp. 8076–8091.

[122] M. Duan, D. Liu, X. Ji, R. Liu, L. Liang, X. Chen, and Y. Tan. “FedGroup: Efficient Federated Learning via Decomposed Similarity-Based Clustering”. In: *2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)* (2020), pp. 228–237.

[123] Z. Gao, Y. Yang, C. Zhao, and Z. Mo. “CFedPer: Clustered Federated Learning with Two-Stages Optimization for Personalization”. In: *2022 18th International Conference on Mobility, Sensing and Networking (MSN)* (2022), pp. 171–177.

[124] Y. J. Cho, J. Wang, T. Chirvolu, and G. Joshi. “Communication-Efficient and Model-Heterogeneous Personalized Federated Learning via Clustered Knowledge Transfer”. In: *IEEE Journal of Selected Topics in Signal Processing* 17 (2023), pp. 234–247.

[125] O. Baños, R. García, J. A. H. Terriza, M. Damas, H. Pomares, I. Rojas, A. Saez, and C. Villalonga. “mHealthDroid: A Novel Framework for Agile Development of Mobile Health Applications”. In: *IWAAL*. 2014.

[126] A. Reiss and D. Stricker. “Introducing a New Benchmarked Dataset for Activity Monitoring”. In: *2012 16th International Symposium on Wearable Computers* (2012), pp. 108–109.

[127] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324. DOI: 10.1109/5.726791.

[128] H. Xiao, K. Rasul, and R. Vollgraf. “Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms”. In: *ArXiv* abs/1708.07747 (2017).

[129] A. Krizhevsky. “Learning Multiple Layers of Features from Tiny Images”. In: 2009.

[130] G. Cohen, S. Afshar, J. C. Tapson, and A. van Schaik. “EMNIST: Extending MNIST to handwritten letters”. In: *2017 International Joint Conference on Neural Networks (IJCNN)* (2017), pp. 2921–2926.

[131] Z. Liu, P. Luo, X. Wang, and X. Tang. “Deep Learning Face Attributes in the Wild”. In: *2015 IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 3730–3738. DOI: 10.1109/ICCV.2015.425.

[132] T. Sztyler and H. Stuckenschmidt. “On-body localization of wearable devices: An investigation of position-aware activity recognition”. In: *2016 IEEE International Conference on Pervasive Computing and Communications (PerCom)* (2016), pp. 1–9.

[133] A. Stisen et al. “Smart Devices are Different: Assessing and MitigatingMobile Sensing Heterogeneities for Activity Recognition”. In: *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems* (2015).

[134] S. Caldas, P. Wu, T. Li, J. Konecný, H. B. McMahan, V. Smith, and A. Talwalkar. “LEAF: A Benchmark for Federated Settings”. In: *ArXiv* abs/1812.01097 (2018).

[135] A. K. Sahu, T. Li, M. Sanjabi, M. Zaheer, A. Talwalkar, and V. Smith. “Federated Optimization in Heterogeneous Networks”. In: *arXiv: Learning* (2018).

[136] L. Wang, W. Wang, and B. Li. “CMFL: Mitigating Communication Overhead for Federated Learning”. In: *IEEE 39th International Conference on Distributed Computing Systems (ICDCS)* (2019), pp. 954–964.

[137] G. Wang, C. X. Dang, and Z. Zhou. “Measure Contribution of Participants in Federated Learning”. In: *2019 IEEE International Conference on Big Data (Big Data)* (2019), pp. 2597–2604.

[138] C. Düsing and P. Cimiano. “Towards predicting client benefit and contribution in federated learning from data imbalance”. In: *Proceedings of the 3rd International Workshop on Distributed Machine Learning* (2022).

[139] D. J. Hand. “Assessing the Performance of Classification Methods”. In: *International Statistical Review* 80 (2012).

[140] A. Nilsson, S. Smith, G. Ulm, E. Gustavsson, and M. Jirstrand. “A Performance Evaluation of Federated Learning Algorithms”. In: *Proceedings of the Second Workshop on Distributed Infrastructures for Deep Learning* (2018).

[141] S. Divi, Y.-S. Lin, H. Farrukh, and Z. B. Celik. “New Metrics to Evaluate the Performance and Fairness of Personalized Federated Learning”. In: *ArXiv* abs/2107.13173 (2021).

[142] H. Snyder. “Literature review as a research methodology: An overview and guidelines”. In: *Journal of Business Research* (2019).

[143] J. Paul and A. R. Criado. “The art of writing literature review: What do we know and what do we need to know?” In: *International Business Review* 29 (2020), p. 101717.

[144] M. Berndtsson, J. Hansson, B. Olsson, and B. Lundell. “Developing your Objectives and Choosing Methods”. In: 2002.

[145] R. Feldt and A. Magazinius. “Validity Threats in Empirical Software Engineering Research - An Initial Survey”. In: *International Conference on Software Engineering and Knowledge Engineering*. 2010.

[146] V. M. Erthal, B. P. de Souza, P. Santos, and G. H. Travassos. “Characterization of continuous experimentation in software engineering: Expressions, models, and strategies”. In: *Sci. Comput. Program.* 229 (2023), p. 102961.

[147] E. C. Weyant. “Research Design: Qualitative, Quantitative, and Mixed Methods Approaches, 5th Edition”. In: *Journal of Electronic Resources in Medical Libraries* 19 (2022), pp. 54–55.

[148] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, and B. Regnell. “Experimentation in Software Engineering”. In: *Springer Berlin Heidelberg*. 2012.

[149] E. Sannara et al. “Evaluation and comparison of federated learning algorithms for Human Activity Recognition on smartphones”. In: *Pervasive Mob. Comput.* 87 (2022), p. 101714.

[150] E. Sannara et al. “Evaluation of federated learning aggregation algorithms: application to human activity recognition”. In: *In ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiCompISWC ’20)* (2020).