İZMİR BAKIRÇAY ÜNİVERSİTESİ

LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ A.B.D.

COMPARISON OF FEDERATED LEARNING FRAMEWORKS FOR MEDICAL IMAGE DOMAIN

YÜKSEK LİSANS TEZİ

Kenan KOCADURDU

Tez Danışmanı: Prof. Dr. Adil ALPKOÇAK

Temmuz 2023

COMPARISON OF FEDERATED LEARNING FRAMEWORKS FOR MEDICAL IMAGE DOMAIN

Yüksek Lisans Tezi

Kenan KOCADURDU İzmir 2023

COMPARISON OF FEDERATED LEARNIGN FRAMEWORKS FOR MEDICAL IMAGE DOMAIN

Kenan KOCADURDU

YÜKSEK LİSANS TEZİ

Bilgisayar Mühendisliği Anabilim Dalı
Danışman: Prof.Dr. Adil ALPKOÇAK

İzmir
İzmir Bakırçay Üniversitesi
Lisansüstü Eğitim Enstitüsü
Temmuz 2023

JÜRİ VE ENSTİTÜ ONAYI

İzmir Bakırçay Üniversitesi Lisansüstü Eğitim Enstitüsü Bilgisayar Mühendisliği Anabilim dalında öğrenim görmekte olan Kenan KOCADURDU'nun "Medikal Görüntü Alanı İçin Federe Öğrenme Çerçevelerinin Karşılaştırılması" başlıklı tezi 05/07/2023 tarihinde aşağıdaki jüri tarafından değerlendirilerek "İzmir Bakırçay Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliği"nin ilgili maddeleri uyarınca, Bilgisayar Mühendisliği Anabilim dalında Yüksek Lisans olarak kabul edilmiştir.

Jüri Üyeleri	Unvanı Adı Soyadı	İmza
Üye (Tez Danışmanı)	Prof. Dr. Adil ALPKOÇAK	
Üye	Dr. Emre ŞATIR	
Üye	Dr. Murat UÇAR	
Üye		
Üye		

…………………………………………..

Prof. Dr. Özge TÜZÜN ÖZMEN

Lisansüstü Eğitim Enstitüsü Müdürü

FINAL APPROVAL FOR THESIS

This thesis titled “Comparison of Federated Learning Frameworks for Medical Image Domain” has been prepared and submitted by Kenan KOCADURDU in partial fulfilment of the requirements in “İzmir Bakırçay University Directive on Graduate Education and Examination” for the Degree of Master of Science in Computer Engineering Department has been examined and approved on 05 / 07 / 2023

Committee Members	Title, Name and Surname	Signature
Member (Supervisor)	Prof. Dr. Adil ALPKOÇAK	
Member	Dr. Emre ŞATIR	
Member	Dr. Murat UÇAR	
Member		
Member		

………………………………………..

Prof. Dr. Özge TÜZÜN ÖZMEN

Director of Graduate Education Institute

ÖZET

MEDİKAL GÖRÜNTÜ ALANI İÇİN FEDERE ÖĞRENME ÇERÇEVELERİNİN KARŞILAŞTIRILMASI

Kenan KOCADURDU

Bilgisayar Mühendisliği Anabilim Dalı

İzmir Bakırçay Üniversitesi, Lisansüstü Eğitim Enstitüsü, Temmuz 2023

Danışman: Prof. Dr. Adil ALPKOÇAK

Tıbbi görüntüleme, hastalıkların teşhis ve tedavisinde kritik bir rol oynamakta olup, karmaşık ve heterojen verilerin büyük miktarlarını üretmektedir. Tıbbi görüntüleme verileri üzerinde makine öğrenimi modelleri eğitmek, veri karmaşıklığı, kıtlığı ve gizlilik düzenlemeleri gibi zorluklarla karşılaşmaktadır. Federated Learning (FL), gizliliği riske atmadan birden fazla cihaz veya veri merkezi üzerinde dağıtılmış verilerle modellerin eğitilmesini sağlayan bir dağıtık makine öğrenimi çözümü olarak ortaya çıkmıştır. Bu tezde, tıbbi görüntüleme alanında FL çerçevelerinin kapsamlı bir analizini sunmaktayım. MobileNetV2 CNN mimarisi ve göğüs röntgeni görüntülerinden oluşan bir veri kümesi kullanarak dört çerçevenin (FedML, FLARE, Flower ve OpenFL) performansını karşılaştırıyorum. Değerlendirme metrikleri arasında hassasiyet, geri çağırma, F1 puanı, doğruluk ve AUC-ROC bulunmaktadır. Sonuçlar, çerçevelerin sınıflandırma doğruluğunda farklılıklar olduğunu göstermektedir, FedML üstün performans sergilerken, onu FLARE ve Flower takip etmektedir. OpenFL ise daha düşük performans sergilemiştir. Bu bulgular, doğru FL çerçevesinin seçiminin doğru tıbbi görüntü sınıflandırması için önemini vurgulamaktadır. Çalışma, araştırmacılara ve uygulayıcılara bilgi sağlayarak çerçeve seçimine yardımcı olmakta ve tıbbi görüntü analizini geliştirmektedir. İleriki araştırmalar, gelişmiş sağlık teşhisleri için FL'yi ilerletmek için ek metrikler, mimariler ve veri setleri üzerinde çalışabilir.

Anahtar Sözcükler: Federe Öğrenme; Medikal Görüntü; Değerlendirme Metrikleri.

ABSTRACT

COMPARISON OF FEDERATED LEARNING FRAMEWORKS FOR MEDICAL IMAGE DOMAIN

Kenan KOCADURDU

Department of Computer Engineering

İzmir Bakırçay University, Graduate Education Institute, July 2023

Supervisor: Prof. Dr. Adil ALPKOÇAK

Medical imaging plays a crucial role in the diagnosis and treatment of diseases, generating large amounts of complex and heterogeneous data. Training machine learning models on medical imaging data faces challenges such as data complexity, scarcity, and privacy regulations. Federated Learning (FL) has emerged as a solution for distributed machine learning, allowing models to be trained on data distributed across multiple devices or data centers without compromising privacy. In this paper, we provide a comprehensive analysis of FL frameworks for the medical image domain. We compare the performance of four frameworks (FedML, FLARE, Flower, and OpenFL) using a dataset of chest X-ray images and the MobileNetV2 CNN architecture. Evaluation metrics include precision, recall, F1-score, accuracy, and AUC-ROC. The results indicate variations in the frameworks' classification accuracy, with FedML demonstrating superior performance, followed by FLARE and Flower. OpenFL exhibited lower performance. These findings emphasize the importance of selecting the appropriate FL framework for accurate medical image classification. The study contributes insights for researchers and practitioners, aiding in framework selection and improving medical image analysis. Further research can explore additional metrics, architectures, and datasets, advancing FL in the medical domain for enhanced healthcare diagnostics.

Keywords: Federated Learning Framework; Medical Image; Evaluation Metrics.

ACKNOWLEDGEMENT

I would like to express my deepest gratitude to my supervisor, Prof. Dr. Adil ALPKOÇAK, for his unwavering support, guidance, and expertise throughout the completion of this thesis. His mentorship and constructive feedback have been instrumental in shaping this work, and I am truly grateful for their dedication and patience.

I would also like to extend my heartfelt appreciation to my loving spouse, İlknur, and my lovely daughter, Ayça. Their constant love, encouragement, and understanding have been a tremendous source of strength and motivation for me during this journey. Their unwavering belief in my abilities has pushed me to overcome challenges and strive for excellence. I am deeply thankful for the sacrifices my family has made, understanding the long hours and countless late nights I have dedicated to this endeavor. Their support and understanding have allowed me to pursue my academic goals wholeheartedly, and I am forever grateful for their unwavering presence by my side.

To my supervisor, Prof. Dr. Adil ALPKOÇAK, and my beloved family, İlknur and Ayça, thank you for being my pillars of support, for believing in me, and for inspiring me to reach new heights. This accomplishment would not have been possible without your encouragement, and sacrifices.

Thank you.

05 / 07 /2023

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Bu tezin bana ait, özgün bir çalışma olduğunu; çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı; bu çalışma kapsamında elde edilen tüm veri ve bilgiler için kaynak gösterdiğimi ve bu kaynaklara kaynakçada yer verdiğimi; bu çalışmanın İzmir Bakırçay Üniversitesi tarafından kullanılan "Turnitin" bilimsel intihal tespit programıyla tarandığını ve hiçbir şekilde "intihal içermediğini" beyan ederim. Herhangi bir zamanda, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçları kabul ettiğimi bildiririm.

Kenan KOCADURDU

05 / 07 /2023

STATEMENT OF COMPLIANCE WITH ETHICAL PRINCIPLES AND RULES

I hereby truthfully declare that this thesis is an original work prepared by me; that I have behaved in accordance with the scientific ethical principles and rules throughout the stages of preparation, data collection, analysis and presentation of my work; that I have cited the sources of all the data and information that could be obtained within the scope of this study, and included these sources in the references section; and that this study has been scanned for plagiarism with “Turnitin” scientific plagiarism detection program used by İzmir Bakırçay University, and that “it does not have any plagiarism” whatsoever. I also declare that, if a case contrary to my declaration is detected in my work at any time, I hereby express my consent to all the ethical and legal consequences that are involved.

Kenan KOCADURDU

TABLE OF CONTENTS

PAGE

TABLES

FIGURES

SYMBOLS AND SHORTCUTS REFERENCE

AI : Artificial Intelligence

CT : Computed Tomography

FL : Federated Learning

GDPR : European General Data Protection Regulation

ILSVRC : ImageNet Large-Scale Visual Recognition Competition

MRI : Magnetic Resonance Imaging

1. INTRODUCTION

Medical imaging is a rapidly growing field that plays a crucial role in the diagnosis and treatment of various diseases. Medical imaging modalities such as X-ray, MRI, and CT generate large amounts of data that contain valuable information about the patient's condition [1]–[3]. However, training machine learning models on medical imaging data poses several challenges. First, medical imaging data is highly complex and heterogeneous, and it requires specialized preprocessing and analysis techniques. Second, medical imaging data is often scarce, and it can be challenging to collect enough data to train accurate and robust models. Third, medical imaging data is sensitive and subject to strict data privacy regulations, which limits the sharing of data between institutions [4], [5].

Artificial intelligence has regained a significant level of interest with the emergence of breakthrough architectures like AlexNet [6], which won the ILSVRC in 2012 [7]. This event sparked a resurgence in machine learning, a subfield of AI, and led to remarkable achievements in computer vision and natural language processing [8], [9]. A critical factor in this success is the exponentially increasing volume of data, which has been made possible by advancements in technology. However, this growing amount of data has brought about new challenges, such as the need for more resources and long training times [10]. As the amount of data continues to increase exponentially, machine learning model training has evolved from centralized architectures to more distributed ones. Furthermore, regulations such as the GDPR [11] added another layer of complexity to data collection and sharing. All this has given rise to new paradigms, including FL, which was first introduced by MacMahan in 2016 [12]. Despite its relatively short lifespan, a significant amount of work has been done on FL algorithms, architectures, and frameworks [13] [14] [15] .

In recent years, there has been a growing interest in using FL in the medical domain, particularly in medical imaging. However, choosing the appropriate framework to implement these algorithms and architectures has become a critical issue. Researchers such as Kholod I [16] and Novikova E [17] have evaluated the data obtained from IoT devices using open-source FL frameworks in their studies. Meanwhile, Liu X [18] and Riviera W [19] have compared a significant number of frameworks based on various parameters they defined.

Additionally, Chen D et al. [20] have proposed a new framework and compared it with several existing ones in terms of their effectiveness and efficiency.

In this thesis, I investigate a comprehensive analysis of FL frameworks for the medical image domain. While the use of FL in this field has been increasing, there has been a lack of detailed analysis of the frameworks used. Our work aims to address this gap by providing researchers and providers with a comparative analysis of the training process and model performance of various FL frameworks for the medical image domain. I believe that our findings will contribute to the advancement of FL in the medical imaging field and assist researchers and providers in improving their products.

This thesis is structured as follows. In Section 2, I provide basic definitions for federated learning. In Section 3, I review related works on comparison of FL frameworks. In Section 4, I introduce FL frameworks. Section 5 presents the results of our experiments and evaluations, where I compare the performance of several FL frameworks. Finally, in Section 6, I provide conclusions and suggestions for future research directions.

2. BASIC DEFINITIONS

Machine learning has seen tremendous advancements, particularly in deep learning techniques. These advancements have enabled more complex models to be developed and trained, leading to better performance and accuracy in various tasks such as image recognition, natural language processing, and speech recognition [8], [9]. However, these advanced models demand an enormous amount of computational power, which has not kept pace with the rapid advancements in machine learning and the increasing volume of data. Consequently, training these models on a single machine or even a cluster of machines can be impractical or unfeasible. Furthermore, with the rise of data, there is a pressing need to manage and process large amounts of data. Centralized machine learning systems that require all data to be stored and processed in one location can be inefficient, slow, and raise privacy concerns. To address these challenges, distributed machine learning architectures have been proposed, which distribute both the computation and the data across multiple machines [10]. This approach accelerates the training process, and also addresses privacy concerns by allowing data to remain on the individual machines where it was collected, while still enabling the creation of a global model. FL is one such paradigm for distributed machine learning that has gained popularity because it enables training of models on data distributed across multiple devices or data centers, without the need to transfer data to a central location [12].

2.1. Federated Learning

The purpose of federated systems is collaboration, and this concept is utilized in various fields, including government, business, and sports organizations. FL [12] is a collaborative machine learning technique designed to ensure data privacy. This technique was first proposed by McMahan in 2016 to provide consumer data privacy as specified in the 2012 White House Report. McMahan defined FL as a collaborative model built with distributed data on mobile devices.

Distributed machine learning architectures aims to raise a common model with different data collections at multiple points in parallel [10]. FL is a special type of distributed machine learning and the obvious differences with other distributed architectures are Non-IID (Non-Identical Independent Distribution), data privacy and unequal state per node [21] [22]. Centralized machine learning poses risks to data privacy because it involves collecting data from different sources and storing it in a central location for training. This centralized storage creates a single point of failure where data can be compromised. Additionally, it may not be feasible to collect and store all data in one location due to regulatory or privacy constraints [21]. FL addresses these concerns by enabling distributed machine learning without requiring the data to be shared or moved to a central location. In this way, data remains on the devices where it was collected, and only model updates are shared, reducing the risk of data exposure. FL also allows for local training of models on sensitive data, such as medical records, that cannot be shared due to privacy concerns. This makes FL a crucial technique for protecting data privacy in the era of big data and machine learning [12].

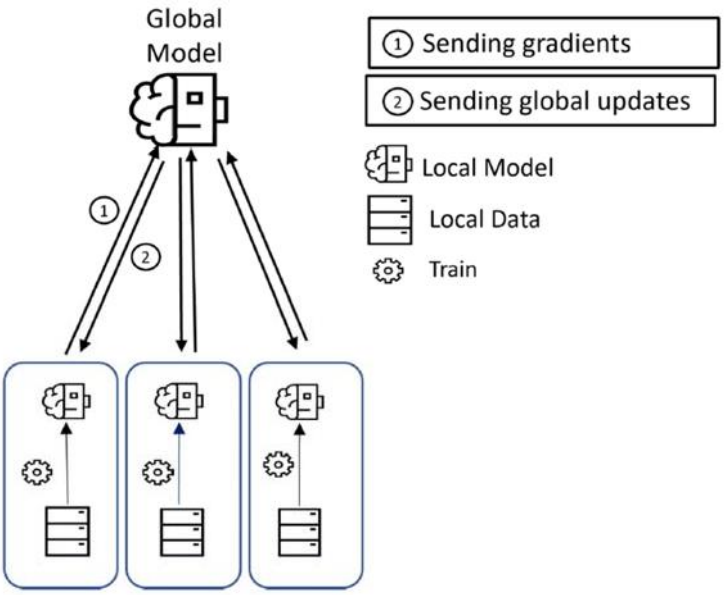

Figure 2.1. Federated Learning Topology [23].

Basic operation of FL, assume we have a dataset D={x,y} where x represents the input and y represents the output. We want to learn a function $f_{(x)}$ that can predict the output y given an input x. In the traditional centralized machine learning approach, we would use this dataset D to train a model on a centralized server. However, in FL, we want to train the model using distributed data that is stored on multiple devices (clients) without compromising the privacy of the data. Let F be the set of all possible models that can be learned from the data. We define the loss function L(f (x), y) as the difference between the predicted output and the actual output. In FL, we want to find the optimal model f * that minimizes the average loss across all clients:

$$f^* = \underset{f(\in F)}{argmin} \frac{1}{N} * \sum_{i=0}^{n} L_i(f(x_i), y_i)$$

where N is the number of clients, x_i and y_i represent the local data at client i, and L_i is the local loss function for client i. The optimization problem is solved in a distributed manner using an iterative process that involves the following steps:

- The server selects a subset of clients to participate in the training process
- The selected clients download the current global model from the server and train the model locally on their data
- The clients upload their updated models to the server
- The server aggregates the models from all clients and updates the global model
- The process is repeated until convergence is reached [12].

FL is a critical technique for protecting data privacy in applications that involve sensitive data, such as healthcare and finance, where data sharing is restricted due to regulatory and privacy concerns.

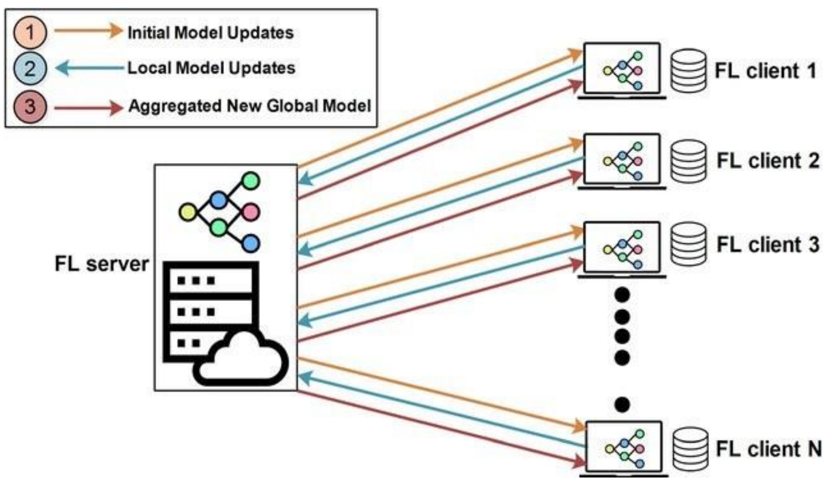

Figure 2.2. General working process of federated learning [24]

2.2. Categorization of Federated Learning

FL can be classified into several categories based on factors such as data distribution and privacy mechanisms [13] [15]. In this evaluation, we focus on three categories of FL based on data distribution: Horizontal Federated Learning, Vertical Federated Learning, and Transfer Federated Learning.

Horizontal Federated Learning involves data that has the same features and output labels across all clients. An example of this category is the use of labeled mammography images with and without cancer obtained from different radiological imaging centers.

Vertical Federated Learning is characterized by data that has the same output but different features. For example, combining data from radiological imaging centers with data obtained from insurance companies falls under this category.

Transfer Federated Learning differs from the other two categories as it involves neither common output nor features. An example is the use of information or models developed for output B to obtain output A. For instance, using a segmentation model trained for a segmentation task in space to create a similar model using radiology center data would be an example of Transfer Federated Learning.

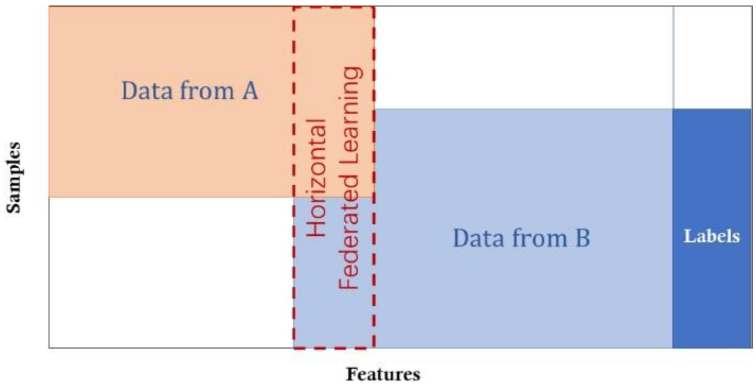

Figure 2.3. Horizontal Federated Learning [15]

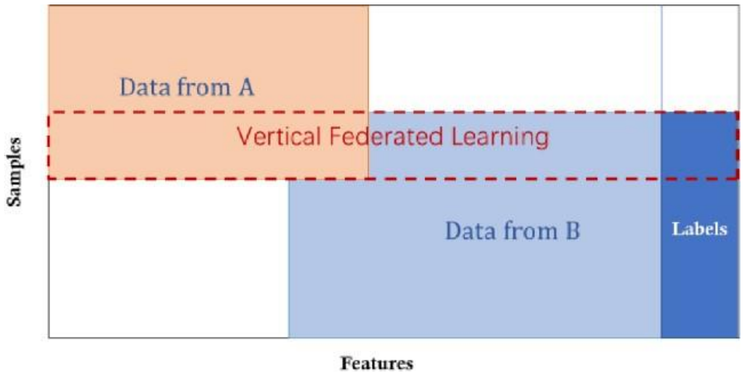

Figure 2.4. Vertical Federated Learning [15]

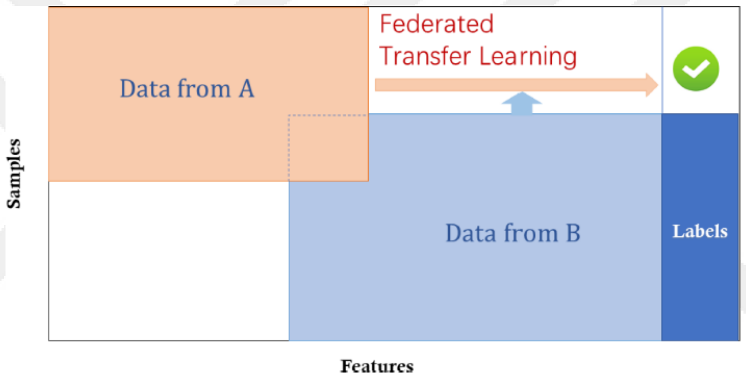

Figure 2.5. Transfer Federated Learning [15]

2.3. Challenges in Federated Learning

Federated learning (FL) faces unique challenges that differ from those of centralized machine learning. One major challenge is non-IID data, where data is not identically distributed across all clients participating in the FL process. Techniques such as data normalization and model personalization can be used to address this challenge [25]. Another significant challenge in FL is ensuring the privacy and security of client data, which is distributed across many devices in a decentralized manner and thus vulnerable to attacks. To protect client data during model training, various privacy-preserving techniques, such as differential privacy and secure aggregation, have been proposed. In addition, FL poses challenges related to communication efficiency and convergence speed, which can impact overall system performance. To mitigate these challenges, techniques such as model compression and parallel processing can be employed [26], [27]

2.4. Federated Learning Frameworks

In this paper, I investigated existing four open-source FL frameworks that are ambitious in the medical image analysis: FedML [28], Flower [29] , NVIDIA Flare [30], OpenFL [31].

2.4.1. FedML

FedML [28] is an open-source research library and benchmark for federated machine learning that provides a common interface for researchers and practitioners to easily develop, benchmark, and compare various FL algorithms.

FedML was developed by researchers from several institutions and is maintained by a community of contributors. It consists of a comprehensive set of building blocks, including data loading, model construction, communication, and optimization modules, to enable the development of customized FL algorithms for different use cases. These building blocks provide a flexible and extensible framework for developing novel federated learning algorithms. In addition to providing a framework for developing FL algorithms, FedML also includes a benchmark suite that enables users to compare the performance of different FL algorithms across various datasets and settings. The benchmark suite includes several standard datasets commonly used in machine learning research, as well as several evaluation metrics for measuring the performance of FL algorithms, such as accuracy, convergence speed, and communication efficiency. FedML also supports a wide range of FL algorithms, including Federated Averaging (FedAvg), Federated Stochastic Gradient Descent (FedSGD), Federated Learning with Differential Privacy (FedDP), and Federated Proximal Gradient Descent (FedProx). These algorithms can be customized and adapted to meet the specific requirements of different applications. One of the key features of FedML is its modular and extensible design. The library is built on top of PyTorch, which makes it easy to integrate with existing PyTorch workflows and models. FedML's modular design allows users to swap out different components, such as optimization algorithms or communication protocols, to tailor the library to their specific use case.

Overall, FedML provides a powerful and flexible framework for developing and benchmarking FL algorithms. Its modular and extensible design, along with its support for a wide range of FL algorithms and evaluation metrics, make it a valuable resource for researchers and practitioners working on federated machine learning

2.4.2. Flower

Flower [29] is an open-source federated learning framework that provides a flexible and modular approach to developing and evaluating FL algorithms. The framework emphasizes flexibility, scalability, and ease of use, and enables developers to define custom client and server logic and communication protocols that can be optimized for different use cases. One of the main strengths of Flower is its support for a variety of FL approaches, including Federated Averaging, Federated Stochastic Gradient Descent, and Federated Distillation.

Overall, Flower is a powerful and flexible FL framework that offers a user-friendly and modular approach to building and evaluating FL algorithms. With its support for various FL approaches and privacy-preserving techniques, as well as its modular architecture and easy-to-use tools, Flower is a great option for developers and researchers interested in building and evaluating FL systems

2.4.3. NVIDIA Flare

NVIDIA Flare (NVIDIA Federated Learning Application Runtime Environment) [30] is a FL framework designed to enable efficient training of machine learning models in distributed environments. The framework is designed to work in diverse environments, ranging from simulation to real-world settings. It can support a wide range of use cases, including autonomous vehicles, healthcare, and industrial IoT.

The Flare framework provides several key features, including distributed data management, privacy-preserving techniques, and scalable model aggregation. The distributed data management feature allows Flare to work with data stored across multiple devices or servers, which can be beneficial in scenarios where data cannot be centralized. The privacy-preserving techniques implemented in the framework include differential privacy and secure multi-party computation, which help to protect sensitive data during the training process. The scalable model aggregation feature allows Flare to aggregate model updates from a large number of devices, which can be useful in applications with a large number of participants. One of the unique features of Flare is its ability to operate in a federated simulation environment. This allows developers to test and refine their models in simulated environments before deploying them in the real world. The framework also provides a comprehensive set of tools for monitoring and debugging FL processes, making it easier to identify and resolve issues that may arise during training.

Overall, NVIDIA Flare is a powerful and flexible FL framework that can be used in a wide range of applications. Its support for privacy-preserving techniques and distributed data management makes it an excellent choice for organizations that need to protect sensitive data during the training process. Its ability to operate in a federated simulation environment is also a significant advantage, as it allows developers to test and refine their models before deploying them in the real world.

2.4.4. OpenFL

OpenFL [31] is an open-source federated learning framework developed by Intel AI, aimed at facilitating the development and deployment of distributed machine learning models that preserve data privacy and security. It provides support for various types of federated learning, including federated averaging, federated boosting, and federated transfer learning, and allows users to customize and configure their own FL algorithms, as well as utilize a set of pre-defined algorithms. Additionally, OpenFL offers features such as differential privacy and secure aggregation to protect sensitive data.

One of the unique features of OpenFL is its support for federated transfer learning, which enables the transfer of knowledge learned from one FL task to another, reducing the training time and improving the accuracy of the models. Another important aspect of OpenFL is its focus on scalability, which allows the framework to handle large-scale FL scenarios with multiple clients and data sources.

Overall, OpenFL is a comprehensive and flexible FL framework that provides users with the tools and features necessary to develop and deploy FL algorithms for various use cases while ensuring data privacy and security. Its support for various types of FL and its compatibility with different platforms make it a powerful tool for researchers and practitioners working in the field of federated learning.

3. RELATED WORKS

FL has gained significant attention from researchers and developers in recent years, leading to the development of many open-source FL frameworks. This section presents a brief overview of some of the most relevant works in this field.

In [16], the authors conducted a comparative analysis of several open-source FL frameworks that are suitable for IoT devices. The authors evaluated several open-source FL frameworks, including TensorFlow Federated Learning, and Pysyft, based on their communication overhead, privacy and security features, and ease of use. The study highlighted the strengths and weaknesses of each framework, providing valuable insights for developers and researchers interested in FL for IoT devices. In [19], the authors presented a user-centric evaluation of FL frameworks, comparing them based on their features, ease of use, and user satisfaction. The study involved 70 participants who tested several frameworks, including TensorFlow Federated and FL Framework. The results showed that users preferred frameworks that were easy to install and had good documentation, highlighting the importance of user experience in the adoption of FL techniques. UniFed [18] is a benchmark suite that evaluates the performance of FL frameworks. The authors developed several benchmarks that measure the scalability, robustness, and efficiency of frameworks, which can be used to compare different FL approaches. The benchmarks included synthetic and real-world datasets, providing a comprehensive evaluation of FL frameworks in different settings. In [17], the authors analyzed the privacy-enhancing technologies used in open-source FL frameworks for driver activity recognition. The study evaluated several frameworks based on their privacy features and showed that some frameworks provide better privacy guarantees than others. The study highlighted the importance of privacy in FL applications and the need for privacy-enhancing technologies in FL frameworks. OpenFed [20] is a comprehensive and versatile open-source FL framework. It provides a user-friendly interface for developers to train and deploy FL models, as well as several privacy-enhancing technologies to ensure data privacy and security.

It is important to note that while these works provide valuable insights into the performance and usability of open-source FL frameworks, they do not specifically focus on medical image domains. Given the unique challenges and requirements of medical image analysis, it is crucial to conduct specialized evaluations of FL frameworks in this field

4. MATERIALS &METHODS

In this section, I provided information about dataset (Curated Dataset for COVID-19), cnn architecture (MobilenetV2) and evaluation methods and metrics (accuracy, precision, recall, F1 score and ROC AUC analysis) I used.

4.1. Experiment Dataset

This dataset [32] is a curated collection of COVID-19 Chest X-ray images obtained by combining 15 publicly available datasets, which are listed in the references section. The dataset consists of the following categories: 1281 COVID-19 X-rays, 3270 Normal X-rays, 1656 Pneumonia-Viral X-rays, and 3001 Pneumonia-Bacterial X-rays. I conducted my experiments by getting only the data labeled COVID-19 and Normal from the database.

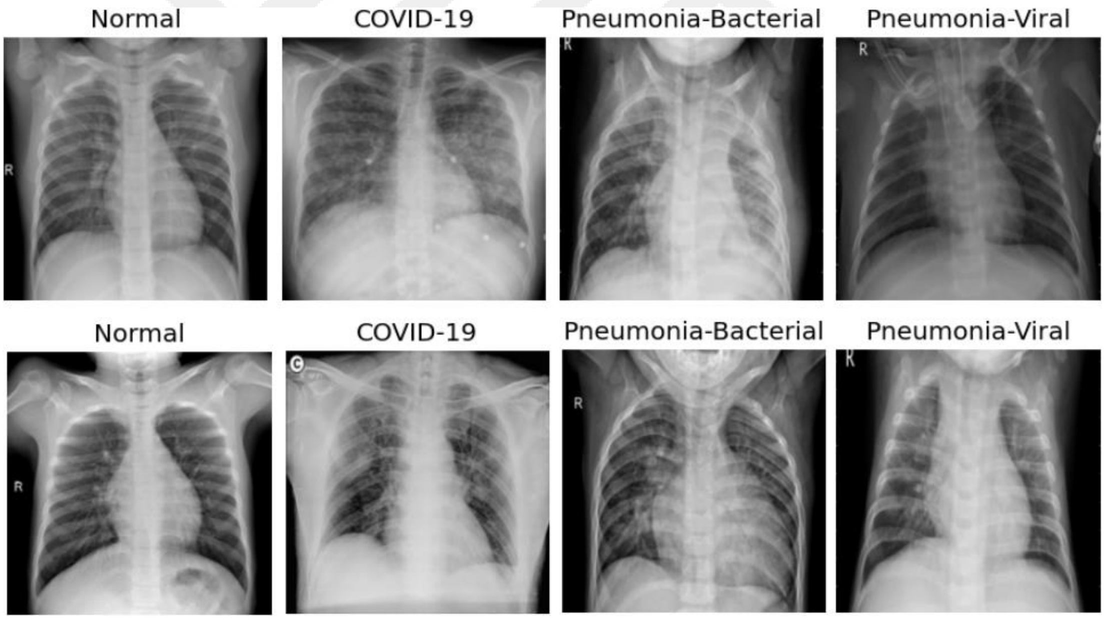

Figure 4.1. Samples of the Curated Collection of COVID-19 Chest X-ray

4.2. CNN Architecture (MobilenetV2)

MobileNetV2 [33], is a state-of-the-art convolutional neural network architecture specifically designed for efficient and accurate image classification on mobile and embedded devices. Introduced by Sandler et al. in their seminal paper, "MobileNetV2: Inverted

Residuals and Linear Bottlenecks," this architecture builds upon the success of its predecessor, MobileNetV1 [34], and incorporates novel features to further optimize resource usage while maintaining high performance. Introduction with the increasing demand for real-time image processing on mobile and edge devices, there is a need for lightweight yet powerful deep learning models. MobileNetV2 addresses this challenge by introducing architectural improvements that enable efficient and accurate image classification.

The core idea behind MobileNetV2 is the use of inverted residuals, which are lightweight and efficient modules for capturing and enhancing feature representations. Inverted residuals consist of a sequence of depthwise separable convolutions followed by 1x1 pointwise convolutions. This design significantly reduces the computational cost while maintaining the expressiveness of the model.

The architecture of MobileNetV2 also incorporates linear bottlenecks to further enhance the representational power of the network. By inserting 1x1 pointwise convolutions with a linear activation function between non-linear layers, the model can preserve more information within the limited dimensions of the bottleneck layer. This technique helps in reducing information loss during the forward pass.

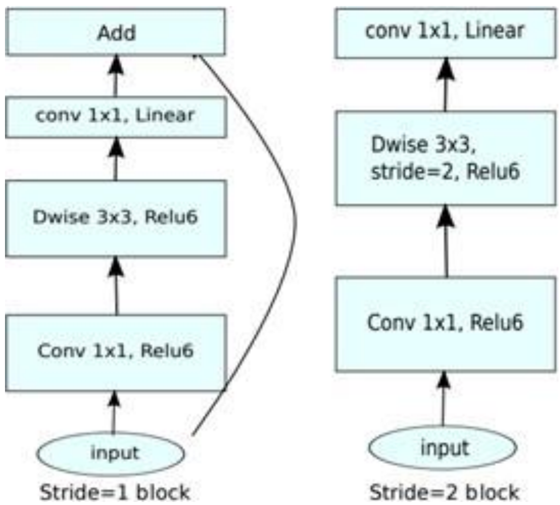

Figure 4.2. Two different components in MobilenetV2 [33]

In addition to the inverted residuals and linear bottlenecks, MobileNetV2 employs other design principles such as residual connections and the use of expansion factors and the stride parameter in depthwise separable convolutions. These elements contribute to the overall efficiency and effectiveness of the architecture, enabling MobileNetV2 to achieve high accuracy on image classification tasks while operating at lower computational costs.

Input	Operator	t	c	n	s
$224^2 \times 3$	conv2d	-	32	1	2
$112^2 \times 32$	bottleneck	1	16	1	1
$112^2 \times 16$	bottleneck	6	24	2	2
$56^2 \times 24$	bottleneck	6	32	3	2
$28^2 \times 32$	bottleneck	6	64	4	2
$14^2 \times 64$	bottleneck	6	96	3	1
$14^2 \times 96$	bottleneck	6	160	3	2
$7^2 \times 160$	bottleneck	6	320	1	1
$7^2 \times 320$	conv2d 1x1	-	1280	1	1
$7^2 \times 1280$	avgpool 7x7	-	-	1	-
$1 \times 1 \times 1280$	conv2d 1x1	-	k	-	

Figure 4.3. MobilenetV2 layers [33]

4.3. Evaluation Metrics

These metrics help quantify and evaluate different aspects of a classification model's performance, allowing for informed decision-making and comparison between different models or parameter settings

Where:

- TP (True Positive) is the number of correctly predicted positive instances.
- TN (True Negative) is the number of correctly predicted negative instances.
- FP (False Positive) is the number of incorrectly predicted positive instances.
- FN (False Negative) is the number of incorrectly predicted negative instances.

4.3.1. Accuracy

Accuracy measures the overall correctness of predictions made by a classification model. It is calculated as the ratio of the number of correct predictions (true positives and true negatives) to the total number of predictions made.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

4.3.2. Precision

Precision quantifies the model's ability to correctly classify positive instances out of all the instances that the model predicted as positive. It is calculated as the ratio of true positives to the sum of true positives and false positives.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

4.3.3. Recall (Sensitivity)

Recall measures the model's ability to correctly identify positive instances out of all the actual positive instances. It is calculated as the ratio of true positives to the sum of true positives and false negatives.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

4.3.4. F1 Score

The F1 score combines precision and recall into a single metric that balances both measures. It is the harmonic mean of precision and recall and provides a single value to evaluate the model's performance.

$$\text{F1 Score} = 2 \text{ x } \frac{\text{Precision x Recall}}{\text{Precision} + \text{Recall}}$$

4.3.5. ROC - AUC

ROC (Receiver Operating Characteristic) curve and AUC (Area Under the Curve) are used to evaluate binary classification models. The ROC curve plots the true positive rate (TPR) against the false positive rate (FPR) as the classification threshold changes. AUC represents the area under the ROC curve and provides a single value to measure the model's performance.

The true positive rate (TPR) is synonymous with recall, and the false positive rate (FPR) is calculated as the ratio of false positives to the sum of false positives and true negatives.

ROC AUC is the area under the ROC curve, which ranges from 0 to 1, with higher values indicating better model performance. An AUC of 1 represents a perfect model, while an AUC of 0.5 suggests a random classifier

5. EXPERIMENTATION & EVALUATION

In this section, I present the experimental setup and results of our comparison of federated learning frameworks for the medical image domain using a chest X-ray dataset and a single CNN architecture, MobileNetV2. We evaluate four frameworks: FedML, FLARE, Flower, and OpenFL. The performance of each framework is assessed based on various evaluation metrics, including precision, recall, F1-score, accuracy, and ROC-AUC.

5.1. Dataset Preparation

I utilized a chest X-ray dataset comprising 9,208 images for training and evaluation purposes. I resized the images in the dataset to 224 x 224 pixels. It was then divided into four clients, each of which was assigned a subset of the data. The dataset was divided into two different distributions: evenly division and unevenly division. To perform the dataset splitting, we employed the train_test_split function from the sklearn library. Initially, we separated the test set with a ratio of 0.2. For an equal division among the four clients, I used a test set ratio of 0.5 for both divisions, with the random_state parameter set to 43. In the case of uneven division, we applied test set ratios of 0.2, 0.1, and 0.1, respectively, while also setting the random_state parameter to 43.

Table 5.1. Class distribution after the preparation process

	Train	Validation	Test	Total
COVID-19	807	207	267	**1281**
Normal	2080	519	671	**3270**
Pneumonia-Bacterial	1918	494	589	**3001**
Pneumonia-Viral	1087	254	315	**1656**
Total	**5892**	**1474**	**1842**	**9208**

After Evenly Division

Table 5.2. Class distribution after evenly division the dataset to 4 clients

	Train				Validation			
	Client 1	Client 2	Client 3	Client 4	Client 1	Client 2	Client 3	Client 4
COVID-19	161	238	205	203	46	57	54	50
Normal	560	494	528	498	124	130	116	149
Pneumonia-Bacterial	472	474	481	491	133	124	120	117
Pneumonia-Viral	280	267	259	281	65	58	78	53
Total	**1473**	**1473**	**1473**	**1473**	**368**	**369**	**368**	**369**

Figure 5.1.a. Train set class distribution (after evenly division)

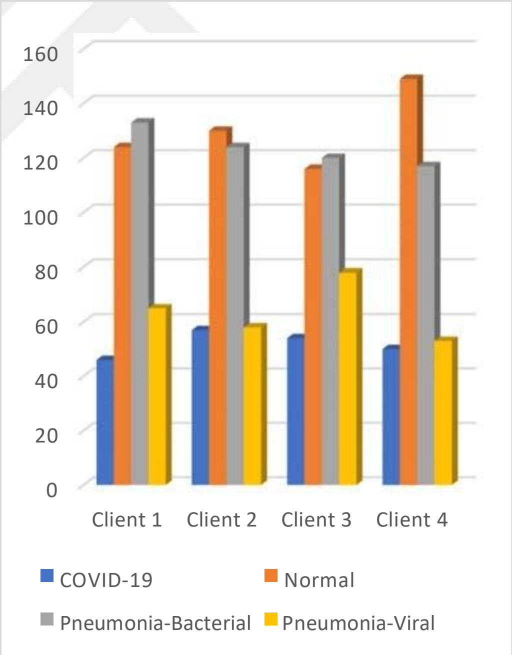

Figure 5.1.b. Validation set class distribution (after evenly division)

After Unevenly Division

Table 5.3. Class distribution after unevenly division the dataset to 4 clients

	Train				Validation			
	Client 1	Client 2	Client 3	Client 4	Client 1	Client 2	Client 3	Client 4
COVID-19	575	61	157	14	156	12	37	2
Normal	1473	184	373	50	372	42	91	14
Pneumonia-Bacterial	1396	146	342	34	357	40	85	12
Pneumonia-Viral	797	81	189	20	176	24	52	2
Total	**4241**	**472**	**1061**	**118**	**1061**	**118**	**265**	**30**

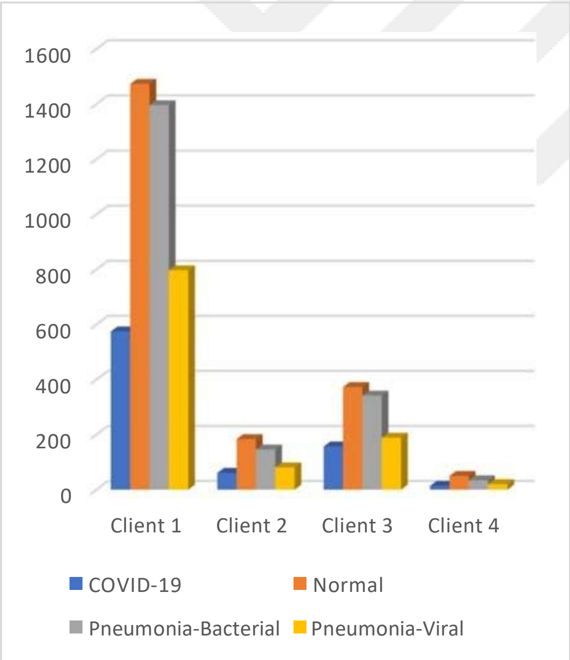

Figure 5.2.a. Train set class distribution (after unevenly division)

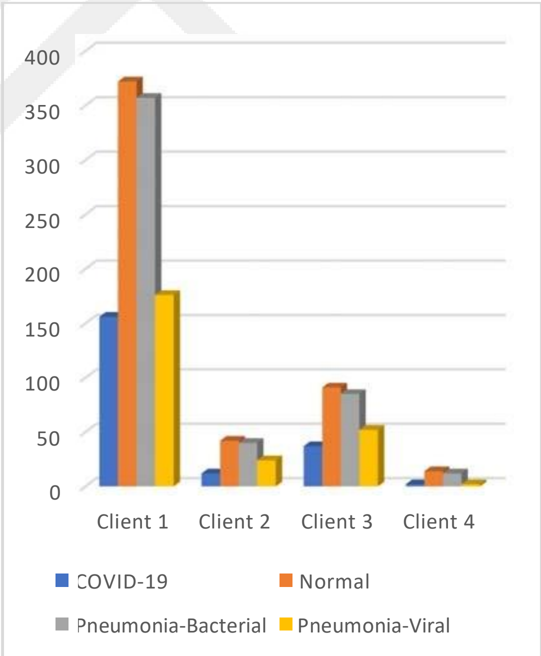

Figure 5.2.b. Validation set class distribution (after unevenly division)

5.2. Experimental Setup

To ensure the consistency of comparison metrics, I conducted all comparisons using identical hardware and, whenever possible, the same versions of commonly used software. The experiments were performed on a device equipped with an Nvidia GeForce RTX 3080 8 GB graphics card, an Intel Core i7 11800H processor, and 32 GB of RAM. Each framework was set up on a separate operating system within the Windows Subsystem for Linux (WSL2) platform.

For model training, I used on the Python programming language and the PyTorch deep learning platform. The following environment details were utilized:

-Windows Subsystem for Linux 2 (Windows 11 Pro)

-Operating System: Ubuntu 20.04.5 LTS (Focal Fossa)

-Python 3.8.10

-Torch 1.13

-Torchvision 0.14.1

-Torchaudio 0.13.1

-Scikit-Learn 1.2.2

-Matplotlib 3.7.1

These specific environments provided the necessary tools and resources for training the models, conducting medical image analysis, and generating the desired results.

5.3. Experimental Procedure

I employed the MobileNetV2 architecture as our base CNN model due to its efficiency and strong performance in image classification tasks. Since our dataset consists of 4 classes, we set the number of output features in the classifier part of the architecture to 4.

Each framework was configured to run with similar hyperparameters, including the number of communication rounds and epochs in round, cross entropy loss function, learning rate (1e-04), batch size (1), and aggregation method. We ensured that the federated averaging algorithm was used for model aggregation in all frameworks.

Each framework was initialized with the same MobileNetV2 model, and the chest X-ray dataset was divided into clients. The federated learning process was performed for 5 communication rounds, with each round consisting of 5 epochs. After the training process, I selected the best models from each framework and evaluated their performance on the test set.

I evaluated the performance of the federated learning frameworks based on the following metrics: Precision, Recall, F1-score, Accuracy and ROC-AUC.

5.4. Results

Below are the results of our experiments, where the performance of each framework is evaluated using the aforementioned metrics. These results provide insights into the effectiveness of each framework in the medical image domain.

Table 5.4. After Training results of accuracy

	FedML		Flare		Flower		OpenFL	
	evenly	unevenly	evenly	unevenly	evenly	unevenly	evenly	unevenly
Acuuracy	0.84	0.86	0.78	0.86	0.81	0.79	0.35	0.67

Table 5.5. After Training results of precision

	Precision							
	FedML		Flare		Flower		OpenFL	
	evenly	unevenly	evenly	unevenly	evenly	unevenly	evenly	unevenly
COVID-19	0.96	0.94	0.92	0.97	0.93	0.90	0.31	0.51
Normal	0.94	0.95	0.91	0.92	0.91	0.95	1.00	0.83
P.-Bacterial	0.78	0.82	0.75	0.79	0.76	0.76	0.56	0.63
P.-Viral	0.59	0.65	0.46	0.73	0.56	0.49	0.18	0.43

Table 5.6. After Training results of recall

	Recall							
	FedML		Flare		Flower		OpenFL	
	evenly	unevenly	evenly	unevenly	evenly	unevenly	evenly	unevenly
COVID-19	0.95	0.96	0.93	0.95	0.91	0.96	0.88	0.65
Normal	0.97	0.95	0.89	0.97	0.94	0.86	0.38	0.86
P.-Bacterial	0.78	0.83	0.76	0.86	0.76	0.79	0.01	0.71
P.-Viral	0.56	0.62	0.47	0.53	0.54	0.52	0.46	0.19

Table 5.7. After Training results of F1 Score

	F1 Score							
	FedML		Flare		Flower		OpenFL	
	evenly	unevenly	evenly	unevenly	evenly	unevenly	evenly	unevenly
COVID-19	0.95	0.95	0.92	0.96	0.92	0.93	0.46	0.57
Normal	0.95	0.95	0.90	0.94	0.93	0.90	0.55	0.84
P.-Bacterial	0.78	0.83	0.75	0.82	0.76	0.77	0.02	0.67
P.-Viral	0.57	0.63	0.47	0.62	0.55	0.50	0.26	0.26

Table 5.8. After Training results of ROC-AUC

	ROC-AUC							
	FedML		Flare		Flower		OpenFL	
	evenly	unevenly	evenly	unevenly	evenly	unevenly	evenly	unevenly
COVID-19	0.9706	0.9762	0.9571	0.9709	0.9478	0.9727	0.7734	0.7731
Normal	09665	0.9585	0.9187	0.9593	0.9447	0.9164	0.6874	0.8775
P.-Bacterial	0.8394	0.8733	0.8188	0.8774	0.8235	0.8366	0.5026	0.7559
P.-Viral	0.7400	0.7732	0.6799	0.7460	0.7254	0.7024	0.5095	0.5694

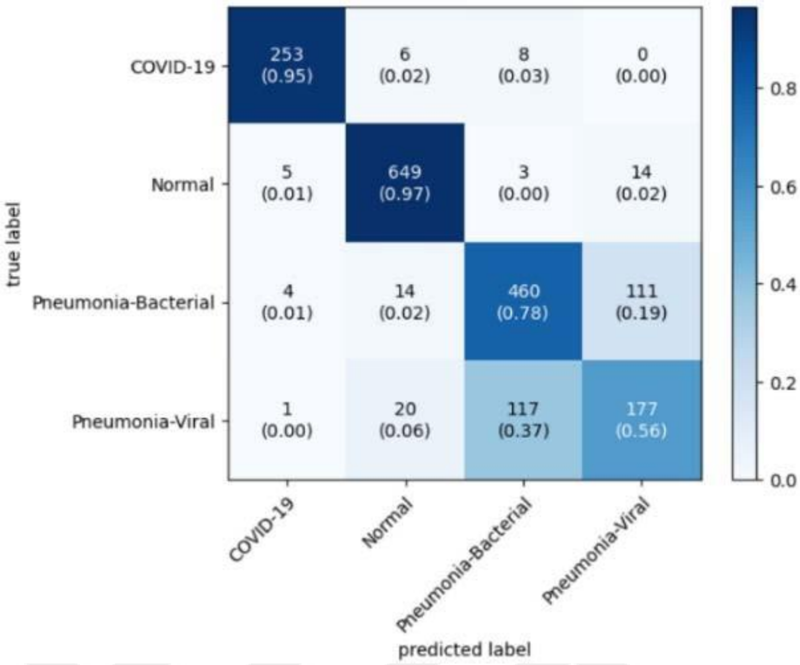

Figure 5.3.a. FedML confussion matrix, evenly distributing

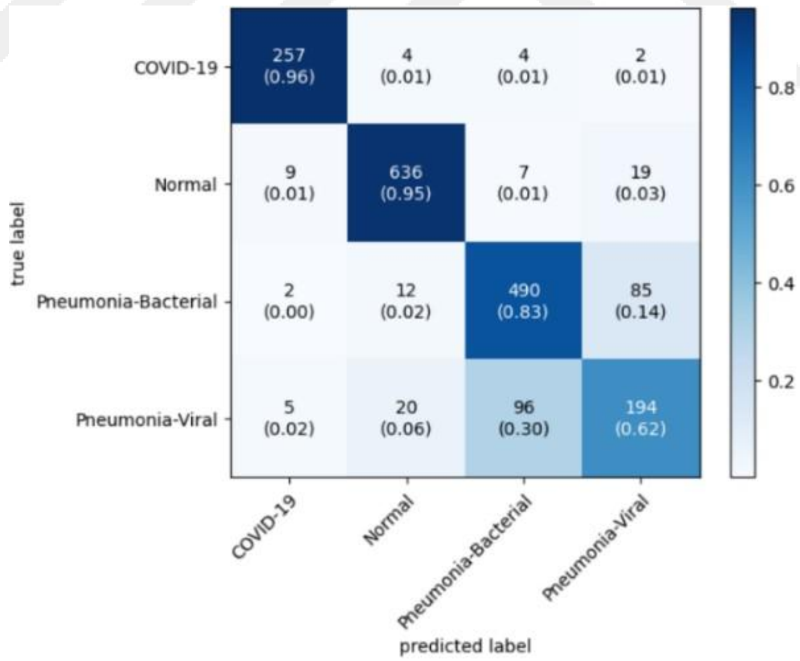

Figure 5.3.b. FedML confussion matrix, unevenly distributing

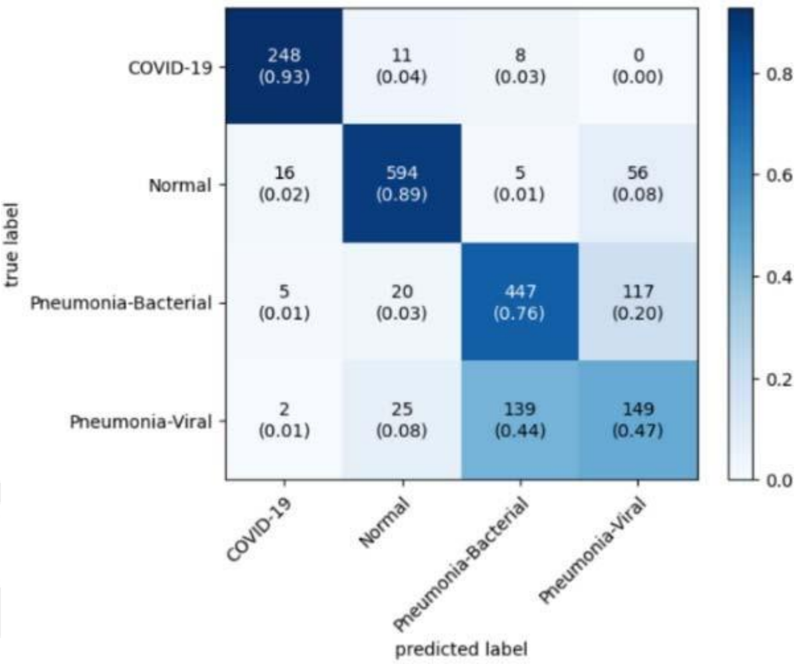

Figure 5.4.a. Flare confussion matrix, evenly distributing

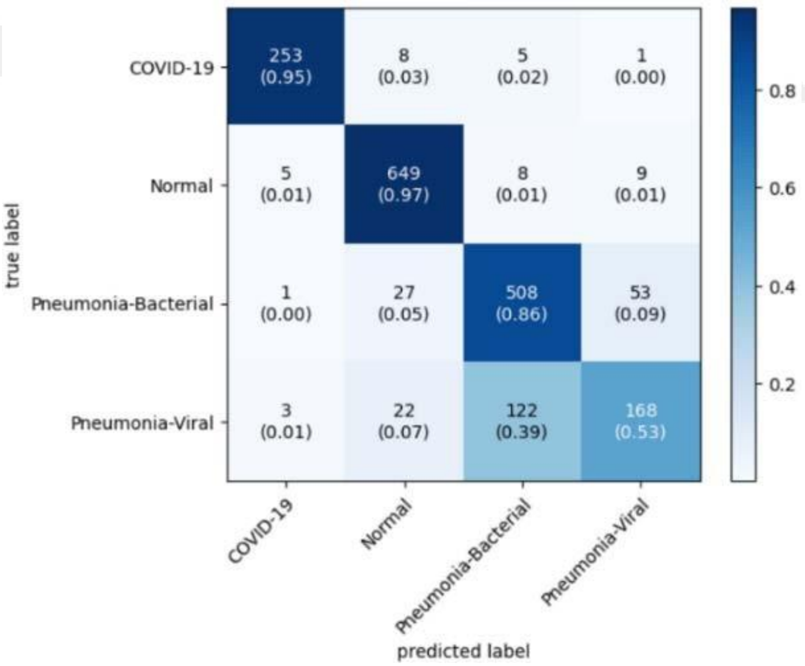

Figure 5.4.b. Flare confussion matrix, unevenly distributing

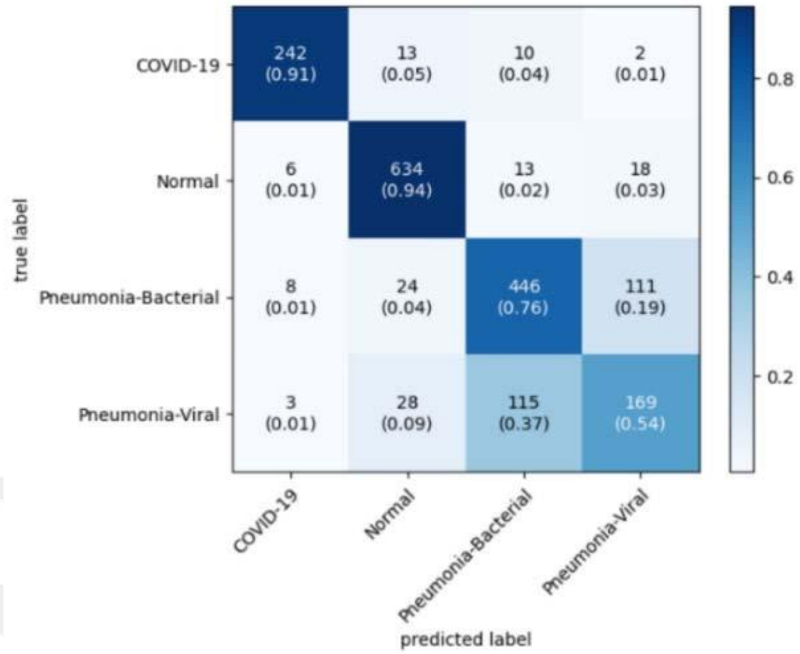

Figure 5.5.a. Flower confussion matrix, evenly distributing

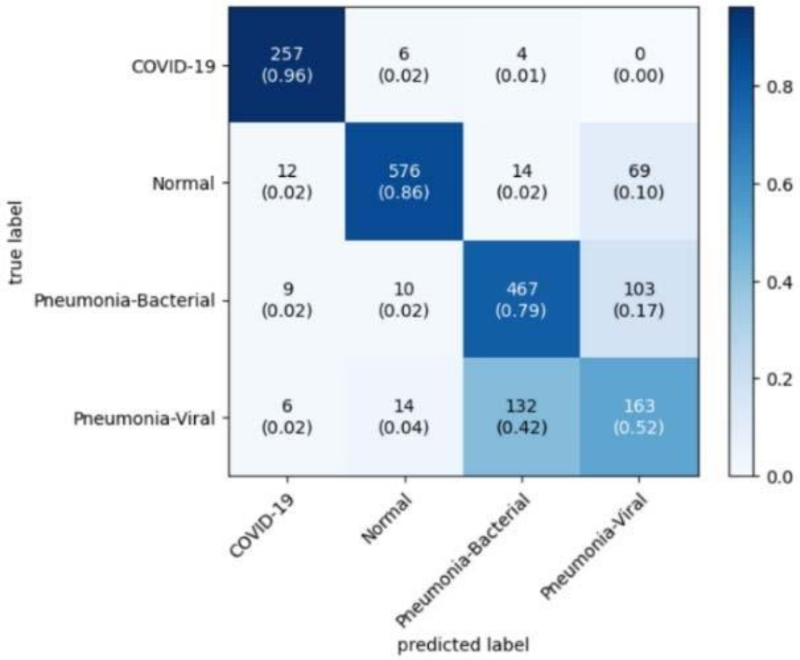

Figure 5.5.b. Flower confussion matrix, unevenly distributing

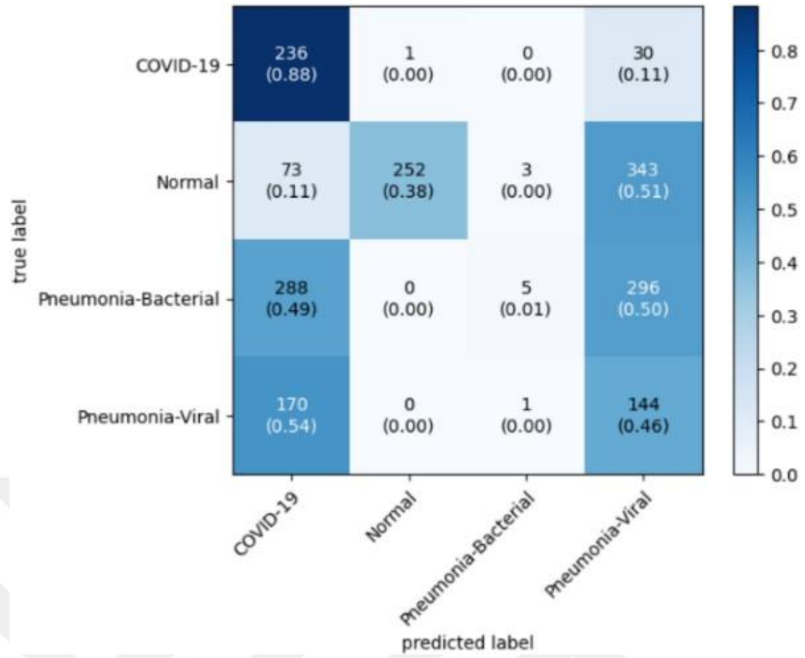

Figure 5.6.a. OpenFL confussion matrix, evenly distributing

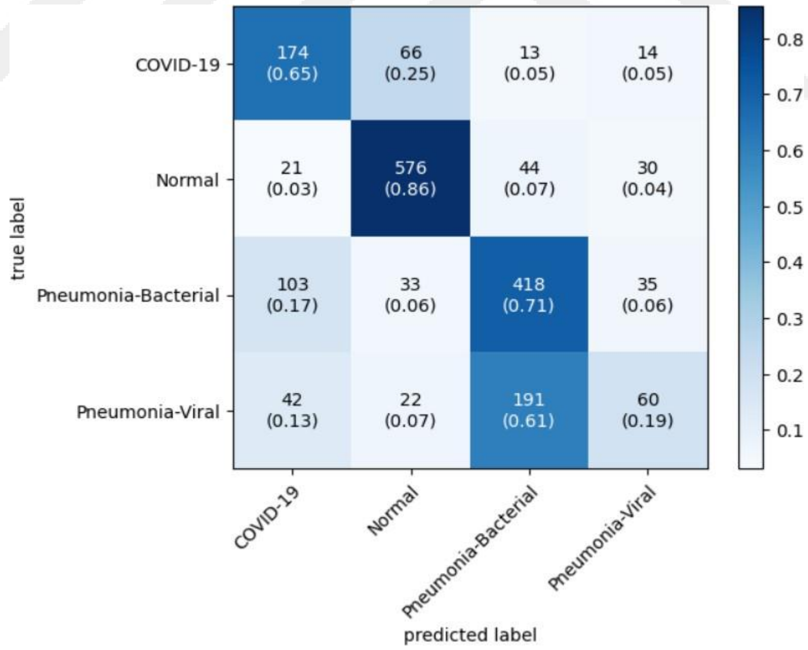

Figure 5.6.b. OpenFL confussion matrix, unevenly distributing

6. CONCLUSION

In this study, we compared four federated learning frameworks, namely FedML, Flare, Flower, and OpenFL, using a dataset of chest X-ray images and a single CNN architecture (MobileNetV2). The experimentation phase focused on evaluating the frameworks based on precision, recall, F1-score, accuracy, and ROC-AUC metrics.

The results of our experiments shed light on the effectiveness of each framework in the context of the medical image domain. Based on the performance metrics, we observed variations among the frameworks in terms of their ability to accurately classify chest X-ray images.

Among the frameworks, FedML demonstrated superior performance in terms of precision, recall, F1-score, accuracy, and ROC-AUC. FLARE and Flower also exhibited competitive performance, although slightly lower than FedML. OpenFL, displayed a fairly lower performance compared to the other frameworks.

These findings suggest that the choice of federated learning framework can significantly impact the accuracy and effectiveness of medical image classification tasks. Researchers and practitioners should consider the performance characteristics of the frameworks, particularly in the context of precision, recall, F1-score, accuracy, and ROC-AUC, when selecting the most suitable framework for their specific medical image analysis needs.

Overall, this study provides valuable insights into the comparative performance of federated learning frameworks in the medical image domain. The results contribute to the existing body of knowledge and can serve as a guideline for researchers and practitioners working on similar tasks, aiding in the selection of the most appropriate framework for achieving optimal results in medical image analysis.

It is important to note that further research and experimentation can be conducted to explore additional performance metrics, consider different CNN architectures, and evaluate the frameworks on larger and more diverse medical image datasets. These efforts can provide a more comprehensive understanding of the strengths and limitations of federated learning frameworks in the medical domain, ultimately advancing the field and facilitating improved healthcare diagnostics and decision-making processes.

REFERENCES

[1] M. Adnan, S. Kalra, J. C. Cresswell, G. W. Taylor, and H. R. Tizhoosh, "Federated learning and differential privacy for medical image analysis," *Sci Rep*, vol. 12, no. 1, Dec. 2022, doi: 10.1038/s41598-022-05539-7.

[2] D. Yang *et al.*, "Federated semi-supervised learning for COVID region segmentation in chest CT using multi-national data from China, Italy, Japan," *Med Image Anal*, vol. 70, May 2021, doi: 10.1016/j.media.2021.101992.

[3] A. Vaid *et al.*, "Federated learning of electronic health records to improve mortality prediction in hospitalized patients with COVID-19: Machine learning approach," *JMIR Med Inform*, vol. 9, no. 1, Jan. 2021, doi: 10.2196/24207.

[4] I. Scholl, T. Aach, T. M. Deserno, and T. Kuhlen, "Challenges of medical image processing," in *Computer Science - Research and Development*, Feb. 2011, pp. 5–13. doi: 10.1007/s00450-010-0146-9.

[5] J. Weese and C. Lorenz, "Four challenges in medical image analysis from an industrial perspective," *Medical Image Analysis*, vol. 33. Elsevier B.V., pp. 1339–1351, Oct. 01, 2016. doi: 10.1016/j.media.2016.06.023.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun ACM*, vol. 60, no. 6, pp. 84–90, Jun. 2017, doi: 10.1145/3065386.

[7] O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," Sep. 2014, [Online]. Available: http://arxiv.org/abs/1409.0575

[8] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep Learning for Computer Vision: A Brief Review," *Computational Intelligence and Neuroscience*, vol. 2018. Hindawi Limited, 2018. doi: 10.1155/2018/7068349.

[9] D. W. Otter, J. R. Medina, and J. K. Kalita, "A Survey of the Usages of Deep Learning in Natural Language Processing," Jul. 2018, [Online]. Available: http://arxiv.org/abs/1807.10854

[10] J. Verbraeken, M. Wolting, J. Katzy, J. Kloppenburg, T. Verbelen, and J. S. Rellermeyer, "A Survey on Distributed Machine Learning," *ACM Computing Surveys*,

vol. 53, no. 2. Association for Computing Machinery, Jun. 01, 2020. doi: 10.1145/3377454.

[11] C. J. Hoofnagle, B. van der Sloot, and F. Z. Borgesius, "The European Union general data protection regulation: What it is and what it means," *Information and Communications Technology Law*, vol. 28, no. 1, pp. 65–98, Jan. 2019, doi: 10.1080/13600834.2019.1573501.

[12] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," Feb. 2016, [Online]. Available: http://arxiv.org/abs/1602.05629

[13] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," *Knowl Based Syst*, vol. 216, Mar. 2021, doi: 10.1016/j.knosys.2021.106775.

[14] Q. Li *et al.*, "A Survey on Federated Learning Systems: Vision, Hype and Reality for Data Privacy and Protection," Jul. 2021, doi: 10.1109/TKDE.2021.3124599.

[15] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Trans Intell Syst Technol*, vol. 10, no. 2, Jan. 2019, doi: 10.1145/3298981.

[16] I. Kholod *et al.*, "Open-source federated learning frameworks for IoT: A comparative review and analysis," *Sensors (Switzerland)*, vol. 21, no. 1, pp. 1–22, Jan. 2021, doi: 10.3390/s21010167.

[17] E. Novikova, D. Fomichov, I. Kholod, and E. Filippov, "Analysis of Privacy-Enhancing Technologies in Open-Source Federated Learning Frameworks for Driver Activity Recognition," *Sensors*, vol. 22, no. 8, Apr. 2022, doi: 10.3390/s22082983.

[18] X. Liu *et al.*, "UniFed: A Benchmark for Federated Learning Frameworks," Jul. 2022, [Online]. Available: http://arxiv.org/abs/2207.10308

[19] Walter Riviera et al., "FeLebrities: a user-centric assessment of Federated Learning frameworks," 2022, doi: 10.36227/techrxiv.21263013.v1.

[20] D. Chen, V. Tan, Z. Lu, and J. Hu, "OpenFed: A Comprehensive and Versatile Open-Source Federated Learning Framework," Sep. 2021, [Online]. Available: http://arxiv.org/abs/2109.07852

[21] J. Liu *et al.*, "From Distributed Machine Learning to Federated Learning: A Survey," Apr. 2021, doi: 10.1007/s10115-022-01664-x.

[22] L. Li, Y. Fan, M. Tse, and K.-Y. Lin, "A review of applications in federated learning," 2020.

[23] E. U. Soykan, L. Karacay, F. Karakoc, and E. Tomur, "A Survey and Guideline on Privacy Enhancing Technologies for Collaborative Machine Learning," *IEEE Access*, 2022, doi: 10.1109/ACCESS.2022.3204037.

[24] A. Qammar, J. Ding, and H. Ning, "Federated learning attack surface: taxonomy, cyber defences, challenges, and future directions," *Artif Intell Rev*, vol. 55, no. 5, pp. 3569–3606, Jun. 2022, doi: 10.1007/s10462-021-10098-w.

[25] P. Kairouz *et al.*, "Advances and Open Problems in Federated Learning," Dec. 2019, [Online]. Available: http://arxiv.org/abs/1912.04977

[26] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated Learning: Challenges, Methods, and Future Directions," Aug. 2019, doi: 10.1109/MSP.2020.2975749.

[27] Z. Li, V. Sharma, and S. P. Mohanty, "Preserving Data Privacy via Federated Learning: Challenges and Solutions," *IEEE Consumer Electronics Magazine*, vol. 9, no. 3, pp. 8–16, May 2020, doi: 10.1109/MCE.2019.2959108.

[28] C. He *et al.*, "FedML: A Research Library and Benchmark for Federated Machine Learning," Jul. 2020, [Online]. Available: http://arxiv.org/abs/2007.13518

[29] D. J. Beutel *et al.*, "Flower: A Friendly Federated Learning Research Framework," Jul. 2020, [Online]. Available: http://arxiv.org/abs/2007.14390

[30] H. R. Roth *et al.*, "NVIDIA FLARE: Federated Learning from Simulation to Real-World," Oct. 2022, [Online]. Available: http://arxiv.org/abs/2210.13291

[31] G. A. Reina *et al.*, "OpenFL: An open-source framework for Federated Learning," May 2021, [Online]. Available: http://arxiv.org/abs/2105.06413

[32] U. Sait *et al.*, "A deep-learning based multimodal system for Covid-19 diagnosis using breathing sounds and chest X-ray images," *Appl Soft Comput*, vol. 109, Sep. 2021, doi: 10.1016/j.asoc.2021.107522.

[33] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," Jan. 2018, [Online]. Available: http://arxiv.org/abs/1801.04381

[34] A. G. Howard *et al.*, “MobileNetV1 : MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications,” Apr. 2017, [Online]. Available: http://arxiv.org/abs/1704.04861

ÖZGEÇMİŞ

Kişisel Bilgiler	
Adı Soyadı	Kenan KOCADURDU
E-Posta	

Eğitim	
Lisans	2015, Süleyman Demirel Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği
Yüksek Lisans	2023, Bakırçay Üniversitesi, Lisansüstü Eğitim Enstitüsü, Bilgisayar Mühendisliği
Doktora	

Yayın Listesi	
Makale	Caglayan, A., Horsanali, M. O., Kocadurdu, K., Ismailoglu, E., & Guneyli, S. (2022). Deep learning model-assisted detection of kidney stones on computed tomography. International braz j urol, 48, 830-839.
Bildiri	
Bilimsel Faaliyetler	2023, Bilimsel Araştırma Projesi, Deep Learning Methods to Automatically Interpret Blood Culture Gram Stains, Bakırçay Üniversitesi Çiğli Eğitim ve Araştırma Hastanesi
Ödüller	