ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ

ΠΟΛΥΤΕΝΧΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ &
ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Τεχνολογίες IoT και αρχιτεκτονικές για Federated Learning: Συγκριτική μελέτη

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΧΡΗΣΤΟΥ ΠΑΠΑΔΟΠΟΥΛΟΥ

Επιβλέπων: Γεώργιος Φραγκούλης
Καθηγητής

ΚΟΖΑΝΗ/ΜΑΡΤΙΟΣ/2024

HELLENIC DEMOCRACY
UNIVERSITY OF WESTERN MACEDONIA

FUCULTY OF ENGINEERING
DEPARTMENT OF ELECTRICAL &
COMPUTER ENGINEERING

A Comprehensive Literature Review of Federated Learning: Principles, Technologies, and for IoT Applications

THESIS

CHRISTOS PAPADOPOULOS

SUPERVISOR: George Fragoulis

Professor

KOZANI/MARCH/2024

ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
& ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ece.uowm.gr

ΔΗΛΩΣΗ ΜΗ ΛΟΓΟΚΛΟΠΗΣ ΚΑΙ ΑΝΑΛΗΨΗΣ ΠΡΟΣΩΠΙΚΗΣ ΕΥΘΥΝΗΣ

Δηλώνω ρητά ότι, σύμφωνα με το άρθρο 8 του Ν. 1599/1986 και τα άρθρα 2,4,6 παρ. 3 του Ν. 1256/1982, η παρούσα Διπλωματική Εργασία με τίτλο "Τεχνολογίες IoT και αρχιτεκτονικές για Federated Learning: Συγκριτική μελέτη" καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας και αναφέρονται ρητώς μέσα στο κείμενο που συνοδεύουν, και η οποία έχει εκπονηθεί στο Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Δυτικής Μακεδονίας, υπό την επίβλεψη του μέλους του Τμήματος κ. Γεώργιο Φραγκούλη αποτελεί αποκλειστικά προϊόν προσωπικής εργασίας και δεν προσβάλλει κάθε μορφής πνευματικά δικαιώματα τρίτων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή / και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή. Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα. Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και μόνο.

Υπογραφή Φοιτητή: ______________

Περίληψη

Αυτή η διατριβή αναλύει την Ομοσπονδιακή Μάθηση (Federated Learning – FL), ένα αποκεντρωμένο παράδειγμα μηχανικής μάθησης που δίνει προτεραιότητα στην ιδιωτικότητα των δεδομένων. Ξεκινά με μια εισαγωγή στην FL, τονίζοντας τη διαφοροποίησή της από την παραδοσιακή κατανεμημένη μάθηση. Η εργασία υπογραμμίζει τις μετασχηματιστικές δυνατότητες της FL σε τομείς όπως τα έξυπνα τηλέφωνα (smartphones), η υγειονομική περίθαλψη και η άμυνα, αντιμετωπίζοντας παράλληλα και προκλήσεις όπως η ετερογένεια της επικοινωνίας και οι αντίπαλες απειλές. Παρέχεται μια ολοκληρωμένη σύγκριση μεταξύ της FL και των παραδοσιακών συγκεντρωτικών μεθόδων μηχανικής μάθησης, καλύπτοντας διάφορους τύπους μοντέλων. Η συζήτηση μεταβαίνει στα ειδικά χαρακτηριστικά της Ομοσπονδιακής Μάθησης, δίνοντας έμφαση στην ιδιωτικότητα των δεδομένων και εξατομίκευση, και παρουσιάζει τους επικρατέστερους αλγορίθμους και εργαλεία.

Ένα σημαντικό τμήμα είναι αφιερωμένο στις πιθανές στρατηγικές επίθεσης σε συστήματα FL, προσφέροντας πληροφορίες για απειλές όπως η αντιστροφή μοντέλου, η υποκλοπή και η δηλητηρίαση δεδομένων, μαζί με τις στρατηγικές αντιμετώπισής τους. Η διατριβή ολοκληρώνεται παρέχοντας μια συνολική επισκόπηση της τρέχουσας κατάστασης της FL και των μελλοντικών επιπτώσεων, λειτουργώντας ως ένας συνοπτικός οδηγός για την πολυπλοκότητα της Ομοσπονδιακής Μάθησης.

Λέξεις κλειδιά:

Ομοσπονδιακή Μάθηση, Αποκεντρωμένη Μηχανική Μάθηση, Απόρρητο Δεδομένων, Στρατηγικές Επίθεσης, Στρατηγικές Μετριασμού

Abstract

This dissertation delves into Federated Learning (FL), a decentralized machine learning paradigm that prioritizes data privacy. It commences with an introduction to FL, highlighting its differentiation from traditional distributed learning. The work underscores FL's transformative potential across sectors like smartphones, healthcare, and defense, while also addressing the challenges it faces, such as communication heterogeneity and adversarial threats. A comprehensive comparison between FL and traditional centralized machine learning methods is provided, covering various model types. The discussion transitions to FL's specific characteristics, emphasizing data privacy and personalization, and introduces the prevalent algorithms and tools.

A significant section is dedicated to potential attack strategies in FL systems, offering insights into threats like model inversion, eavesdropping, and data poisoning, alongside their mitigation strategies. The dissertation culminates by offering a synthesized overview of FL's current state and its future implications, serving as a concise guide to the complexities of Federated Learning.

Keywords:

Federated Learning, Decentralized Machine Learning, Data Privacy, Attack Strategies, Mitigation Strategies

Thanks to

I extend my deepest gratitude to Professor George Fragoulis, whose guidance and expertise have been instrumental in shaping this thesis on Federated Learning. His unwavering support, insightful feedback, and encouragement have truly enriched my academic journey.

A heartfelt thank you also goes to Mr. Konstantinos Kollias (PhD Candidate), whose collaboration and valuable insights significantly contributed to the depth and quality of this work. The collaborative spirit and intellectual exchange have been invaluable throughout the research process.

I want to express a special thanks to my family for their boundless support, understanding, and encouragement. Their enduring patience and belief in my pursuit of knowledge have been the bedrock of my academic endeavors. To my parents and siblings, your love and encouragement have sustained me through the challenges, and this achievement is as much yours as it is mine. Thank you for being my pillars of strength.

This thesis is a culmination of shared efforts and unwavering support, and I am profoundly grateful for the contributions of everyone who has been part of this academic journey.

Table of Contents

ΠΕΡΙΛΗΨΗ 7

ABSTRACT 9

THANKS TO 11

LIST OF FIGURES 18

LIST OF TABLES 20

FOREWORD 21

CHAPTER 1: INTRODUCTION 22

1.1 Introduction to FL 22

1.2 FL Basics 24

1.3 Definition and Alternative Nomenclatures in Distributed Learning 26

1.4 Thesis Overview 27

CHAPTER 2: GENERAL OPPORTUNITIES, CHALLENGES & APPLICATIONS 29

2.1 Opportunities 29

2.2 Areas of Application 32

2.2.1 Smartphones 32

2.2.2 Organizations 32

2.2.3 Internet of Things (IoT) 33

2.2.4 Healthcare 33

2.2.5 Personalized Advertising 33

2.2.6 Automotive Sector 34

2.2.7 Defense Against Financial Fraud 34

2.2.8 Insurance and FL 34

2.2.9 Military and Police forces 34

2.3 Main Challenges 35

2.3.1 Communication and System Heterogeneity 36

2.3.2 Threats and Adversarial Attacks 36

2.3.3 Infrastructure and Bandwidth Issues 36

CHAPTER 3: FL & CENTRALIZED LEARNING: COMPARISON AND CLASSIFICATION 38

3.1 Comparison with Traditional/ Centralized Machine Learning 38

3.1.1 Linear Models 40

3.1.2 Tree Models 42

3.1.3 Neural Network Models 43

3.1.4 Reinforcement Learning 47

3.2 Types of FL 49

3.2.1 Horizontal FL (HFL) 50

3.2.2 Vertical FL (VFL) 51

3.2.3 Federated Transfer Learning 53

3.2.4 Federated Reinforcement Learning 55

CHAPTER 4: CHARACTERISTICS 56

4.1 Data Privacy 56

4.1.1 Introduction to Data Privacy in FL 56

4.1.2 Motivations for Enhanced Data Privacy 57

4.1.3 Differential Privacy in FL 57

4.1.4 Secure Multi-Party Computation (SMPC or SMC) 58

4.1.5 Homomorphic Encryption in FL 59

4.1.6 Federated Identity and Access Management (FIAM) 60

4.1.7 Privacy-Preserving Data Aggregation 61

4.1.8 Privacy Challenges and Limitations in FL 62

4.1.9 Best Practices and Standards 63

4.1.10 Future Directions 64

4.2 Personalization 65

4.2.1 Localized Model Training 65

4.2.2 User Data Privacy 65

4.2.3 Improved User Experiences 66

4.2.4 Aggregated Global Model 66

4.2.5 Data Efficiency 66

4.2.6 Collaborative Personalization 66

4.2.7 Dynamic Adaptation 67

4.2.8 Challenges and Trade-offs 67

4.3 Non-IID Data in FL 67

4.3.1 Understanding Non-IID Data 68

4.3.2 The Implications and Challenges of Non-IID Data 68

4.3.3 Non-IID: Potential Strategies 68

4.3.4 Real-world Implications across Sectors 69

4.3.5 Synchronous vs. Asynchronous Updates 69

4.3.6 Privacy and Security in Non-IID Settings 69

4.4 Design Space 69

4.4.1 Data Distribution and Management 70

4.4.2 Model Architectures 70

4.4.3 Communication Strategies 70

4.4.4 Privacy and Security 70

4.4.5 Scalability and Robustness 71

CHAPTER 5: ALGORITHMS & FRAMEWORKS 72

5.1 FL Algorithms 72

5.1.1 Introduction 72

5.1.2 Federated Averaging (FedAvg) 73

5.1.3 Federated Stochastic Gradient 75

5.1.4 Model Aggregation Methods 77

5.1.5 Communication-Efficient Algorithms 80

5.1.6 Adaptive Learning Rates in Federated Contexts 81

5.1.7 Asynchronous Methods and Algorithms 84

5.1.8 State-of-The-Art FL Algorithms: An Indicative Comparative Benchmark against FedAvg 87

5.2 Frameworks and Tools 93

5.2.1 TensorFlow Federated (TFF) 93

5.2.2 PySyft 93

5.2.3 FATE (Federated AI Technology Enabler) 94

CHAPTER 6: ATTACK STRATEGIES 95

6.1 Introduction 95

6.1.1 Contextualizing Attacks in FL 95

6.1.2 Significance of Addressing Attack Vectors 96

6.2 Model Inversion Attacks 97
6.2.1 Understanding Model Inversion 97
6.2.2 Implications of Model Inversion in FL 98
6.2.3 Mitigating Model Inversion Attacks 99
6.3 Membership Inference Attacks 100
6.3.1 The Nature of Membership Inference 100
6.3.2 Threats to Data Privacy and Impacts 100
6.3.3 Strategies to Counter Membership Inference Attacks 101
6.4 Eavesdropping and Man-in-the-Middle Attacks 102
6.4.1 Characterizing Eavesdropping in a Federated Setting 103
6.4.2 Potential Damages and Consequences 104
6.4.3 Secure Communication Protocols 105
6.5 Data Poisoning 106
6.5.1 Introduction to Data Poisoning in FL 106
6.5.2 Impact on Model Integrity and Performance 107
6.5.3 Defense Mechanisms against Poisoning 108
6.6 Model and Gradient Tampering Attacks 108
6.6.1 Identifying Adversarial Tampering Techniques 108
6.6.2 Potential Threats to Model Training and Convergence 109
6.6.3 Solutions and Countermeasures 110
6.7 Backdoor Attacks 111
6.7.1 Understanding the Nature of Backdoor Attacks 111
6.7.2 Implications of Backdoor Attacks in FL 112
6.7.3 Overcoming the Threat of Backdoor Attacks 113
6.8 Sybil Attacks 114
6.8.1 The Principle of Sybil Attacks in Distributed Systems 114
6.8.2 Consequences in a FL Context 115
6.8.3 Defense Protocols against Sybil Attacks in FL 115
CHAPTER 7: CONCLUSIONS 118
7.1 Deconstructing FL: Insights and Implications 118
7.2 Future Implications and Directions 119
7.3 Concluding Note 120

List of Figures

Figure 1 - FL Architecture (3) 24

Figure 2: Typical Processes of FL (4) 25

Figure 3 - Data minimization in federated vs. centralized approaches (61) 39

Figure 4 - A visual representation of Reinforcement learning, where t is the timestep, S_t is the state of the environment, R_t is the reward obtained for this state-action pair, A_t is the action of the RL agent, R_{t+1} and S_{t+1} are the next state and next reward which occurred from A_t action (88). 48

Figure 5 – Types of FL 49

Figure 6 - Comparison of HFL and VFL frameworks, where s i represents the i th record, x i j and y i represent the j th feature and the label of the i th record, respectively (98) 52

Figure 7 - Inference process of VFL systems with a coordinator (98) 53

Figure 8 - Different data partition of horizontal FL, vertical FL, and federated transfer learning (103) 54

Figure 9 - Side-by-side comparison of an I.I.D. and non-I.I.D. distribution taken from one of the clients (148) 91

Figure 10 - Posterior probability distribution of a correlated Bayesian t-test between classifiers A and B. It defines a region of practical equivalence where the mean difference in accuracy is no more than ±1%. (148) 91

Figure 11 - Comparisons between federated optimization algorithms and centralized learning in the form of posterior distributions on MNIST I.I.D. and non-I.I.D. Each algorithm ran three iterations of 5-fold cross-validation. Note that the x-axes have different scales (148)................92

List of Tables

Table 1- Algorithm 1 Federated averaging (148)..74

Table 2 - SVRG Algorithm (148) 88

Table 3 – CO-OP Algorithm (148) 89

Table 4 - Summary of algorithm comparisons, showing if the algorithm in a row is better (+), worse (−), or practically equivalent (=) compared to the algorithm in a column (148) 92

Foreword

In the vast realm of technology, where innovations shape our daily interactions, Federated Learning (FL) stands out as a beacon of change, ushering in a new era of privacy and efficiency. This thesis embarks on a journey to unravel the story behind FL – a story that goes beyond the complexities of algorithms and technical intricacies.

Imagine a world where your devices collaborate like a synchronized orchestra, learning from your interactions while ensuring the utmost privacy. This is the essence of Federated Learning. It's about letting your smartphone, smartwatch, or any connected device refine its understanding of you without ever exposing your personal data. It's the promise of technology working for you while respecting your privacy boundaries.

As we navigate through the pages of this thesis, we step into the heart of FL, discovering how this approach is reshaping the landscape of machine learning. At its core, FL is about creating harmony between the need for personalized technology and the imperative of safeguarding our individuality. It's a tale of striking that delicate balance, where privacy is not sacrificed at the altar of technological advancement.

We explore the algorithms behind FL – the driving force that allows your devices to collaboratively learn and adapt. However, fear not the jargon and complexities; instead, relish the simplicity of the idea that your gadgets can become smarter while keeping your secrets safe. This isn't just a scientific exploration; it's an unveiling of the magic that happens behind the scenes, ensuring that technology remains a trusted companion in our lives.

Beyond the technical nuances, this thesis aims to convey the broader significance of FL. It's about recognizing the potential of technology to adapt to our preferences without compromising our personal space. It's an acknowledgment that privacy and progress can coexist, laying the foundation for a future where our devices not only serve us better but also respect our need for discretion.

So, join us on this journey through the uncharted territories of Federated Learning. Let's explore the potential, the challenges, and the promise it holds for a world where our devices become smarter, more personalized, and yet, ever so respectful of our individuality.

Chapter 1: Introduction

1.1 Introduction to FL

Federated Learning (FL) is a pioneering approach to decentralized machine learning, which is particularly relevant in the age of modern mobile devices that amass a vast amount of user data. As highlighted by McMahan in 2016 in [1] these devices have the potential to greatly enhance the user experience by leveraging data to train models, such as language models for improving speech recognition or image models for photo selection. However, the nature of this data, often being voluminous and privacy-sensitive, may hinder its central storage and conventional training methodologies. Instead, FL proposes a paradigm where the data remains on the devices, and a shared model is developed by collating locally computed updates. The method is particularly adept at handling the unique challenges of unbalanced and non-IID data distributions, while substantially curtailing communication costs, making it an innovative solution in the field of machine learning.

Central to FL is the framework that permits devices or nodes (like computers and mobile phones) to learn autonomously from data they store locally. Instead of dispatching the entirety of their raw, often sensitive, data, they engage in on-site training. Subsequently, only the model's updates, which lack direct raw data insights, are shared with a central server. This server plays a crucial role in aggregating these myriad updates to refine and enhance a global model. The improved version of this global model is then relayed back to the various devices, paving the way for further iterative refinements.

FL is not just another algorithm or technique but a monumental shift in the world of machine learning. Unlike classic methods which necessitated the centralization of data regardless of its origin — be it from users on smartphones, sensor data from vehicles, or voice inputs from smart speakers — FL ushers in a decentralization era. It presents a novel paradigm where model training occurs directly at the data source, be it individual devices or nodes in a network, without raw data ever leaving its original location. This approach promises to address numerous concerns in the realm of data privacy, data transfer costs, and real-time processing [2].

FL is not confined to a specific domain, showcasing diverse applications. Its applicability spans from healthcare, emphasizing patient data privacy, to the domain of industrial IoT, where data transferability might be compromised due to bandwidth constraints. The depth of a dissertation can shed light on its potential uses across varied sectors. This importance is further magnified when considering the ethical implications. Centralized models, for instance, often carry the risk of bias since they may be trained on non-representative datasets. Through thorough research on FL, there's potential to offer solutions that ensure machine learning models become more inclusive and equitable, giving weight to data from a broad spectrum of sources.

However, the journey doesn't stop there. While FL stands tall in addressing the pressing issue of data privacy, it simultaneously unveils new challenges centered around model security. By diving into these aspects, the academic community stands a chance to develop models that are not only privacy-centric but also resistant to adversarial attacks. This venture into the federated realm also throws light on another dimension: optimization techniques. Traditional machine learning models, structured for centralized systems, might fall short in a federated environment. The focus of a dissertation could be pivotal in reviewing innovative algorithms and strategies tailored specifically for these settings.

Beyond the technical intricacies, there's a broader canvas to paint. FL, with its decentralized essence, has the promise to drastically reduce data transfer overheads. Such an understanding can pave the way for models that are more efficient, an essential trait, especially when the real-world scenario is riddled with limited bandwidth and computational resources. The regulatory landscape also plays a pivotal role. With stringent data protection regulations, such as the GDPR, federating learning emerges as a beacon, hinting at solutions to navigate the intricate maze of data transfer across borders. As researchers focus on it, they can not only decipher its potential but also shape the trajectory of future policy decisions.

The economic fabric of organizations also stands to gain. The shift from a centralized to a federated model can usher in substantial economic advantages. The simple act of not centralizing data can lead to savings in data storage, transfer, and processing, all of which can be accentuated through rigorous research. Furthermore, a dissertation on FL transcends the boundaries of mere technology. It has the power to bridge the divide between computer science, law, ethics, and even economics, sparking collaborations across various academic and industrial terrains.

Finally, by delving deep into FL through this dissertation, one can contribute to understanding its intricacies, potential benefits, and limitations. Furthermore, as industries worldwide become more data-driven yet more conscious about user privacy, the insights from such research can provide valuable guidelines for implementing decentralized machine learning while safeguarding users' sensitive information.

In today's digital age, where individuals are meticulously guarding their digital footprints, FL could be the paradigm shift needed for a more privacy-conscious global community. The societal implications of research in this domain are profound. It can drive a transformative shift in the perception and handling of data by both industries and consumers. Lastly, as the world pivots towards an era dominated by edge computing, where computations migrate closer to data sources, the importance of FL swells. A dissertation on this topic not only addresses current concerns but also lays the groundwork for future AI trajectories in a world that is progressively leaning towards decentralization.

1.2 FL Basics

As illustrated in Figure 1 by Sikandar et al. [3] and Figure 2 by Liu et al. [4], FL pioneers a unique approach: instead of ferrying data to a central point for analysis and model building, the training itself travels to where the data resides.

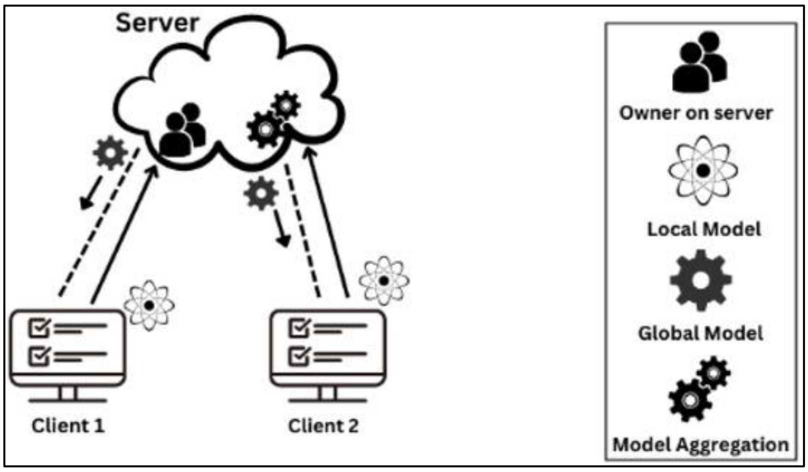


Figure 1 - FL Architecture [3]

As depicted, FL allows multiple users in a network to utilize their local data to contribute to an integrated, global model. This process, while ensuring data privacy, also leverages the diverse and expansive nature of the data residing on these devices, leading to more robust and generalizable models.

The FL process consists of three core parts: the learning algorithm and training method, privacy protection mechanisms, and incentive mechanisms [3].

1. Learning Algorithm and Training: The server trains a model by repeating several steps, like designing the learning algorithm, selecting clients, distributing the model, updating the client model, and server-side aggregation and model updates.

2. Privacy Protection Mechanism: FL can protect user data privacy by employing data encryption training, ensuring that the model doesn't reveal the original data. Additionally, encrypting the data transmission process ensures that only intermediary results without extra information are relayed during the training process.

3. Incentive Mechanism: Since FL relies on participation, it's essential to offer adequate incentives to participants. The incentive mechanism strives to equitably share the benefits of FL, motivating users to participate consistently and deterring malicious actors from dominating the process.

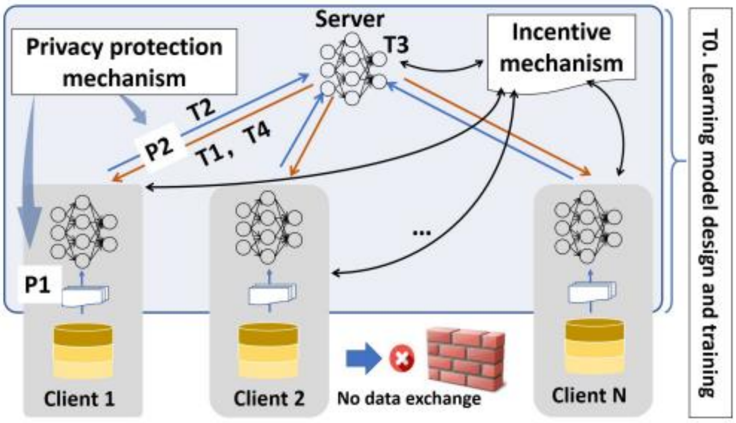


Figure 2: Typical Processes of FL [4]

More specifically, FL allows multiple users in a network to utilize their local data to contribute to an integrated, global model. As illustrated in Figure 2, the process begins with T0, where both client-side and server-side learning algorithms are designed per application needs. At T1, the server identifies qualifying clients and delivers them the model along with training settings. T2 sees the selected client updating the model using its training program. By T3, the server aggregates inputs based on the client's model or parameters. The process culminates at T4, where the server updates the shared model. For ensuring data privacy, P1 highlights the use of data encryption during training to prevent data inference. Meanwhile, P2 emphasizes encrypting data during transmission, allowing only essential intermediate results to be communicated [4].

A more analytical skeleton of the FL process can be streamlined into the following sequence [5]:

- **Initialization:** On a server, a global model is birthed.
- **Model Distribution:** This nascent global model is dispatched to the various participating nodes or devices.
- **Local Training:** Each node dedicates itself to training this model using its reservoir of local data.
- **Model Update Sharing:** Post this training, the nodes send back model updates to the central server.
- **Aggregation:** The central entity aggregates these updates, often employing techniques like Federated Averaging as proposed by [6].
- **Iteration:** The previous three steps (distribution, training, and aggregation) are reiterated until the model achieves an acceptable level of performance.

Yet, FL is not without its challenges. Despite its groundbreaking design, issues persist, such as ensuring the twin objectives of secure and efficient communication, navigating the heterogeneity of systems, grappling with skewed and non-IID data, and warding off malicious threats. Li T. et al. [7] specifically underscore the importance of data security and the need for efficient communication channels that prioritize the transfer of only the most pivotal model updates, minimizing data breach risks.

Several distinct concerns associated with the more orthodox centralized learning approach are addressed seamlessly by FL, which can be categorized in the following 3 classes:

1. Regulatory Hurdles: Globally, there's a surge in regulations such as GDPR in Europe, CCPA in California, and PDPB in India. These laws are stringent about the movement and amalgamation of sensitive data, especially when data protection standards vary across regions. FL sidesteps these regulatory mazes by minimizing data movement.

2. User Privacy: Users are increasingly concerned about the privacy of their data. Applications that entail inputting sensitive information like passwords or credit card details raise expectations that such details remain on the device, shielded from external servers. FL enshrines this expectation, reinforcing user trust.

3. Data Volume and Bandwidth: A growing number of devices, especially those like surveillance cameras, generate vast data volumes. Transferring this colossal amount of data centrally isn't just a logistical nightmare but often economically unviable. By enabling on-site training, FL substantially cuts down on bandwidth demands, making it a more scalable option.

The upcoming chapters and sections examine in detail all these characteristics, challenges, methods along with discussing the ongoing research efforts to mitigate them and optimize FL systems. Despite the obstacles, the transformational potential of FL remains unassailable, making it a promising direction in the field of machine learning and artificial intelligence.

1.3 Definition and Alternative Nomenclatures in Distributed Learning

As discussed, FL is a nascent yet rapidly evolving domain within the broader ambit of machine learning. However, in emerging academic fields like this, it's typical for a plethora of terminologies to arise, many of which might describe methods similar to or overlapping with FL. We will delve into each of these terms to pinpoint the specific area this dissertation will concentrate on. A comprehensive understanding of these terms is essential for scholars, practitioners, and industry experts to ensure accurate communication and knowledge dissemination.

By definition, FL is a distributed machine learning approach wherein a model is trained across multiple devices or nodes without the need to centralize the training data. This paradigm ensures data privacy and minimizes data transfer overheads, especially when dealing with vast amounts of decentralized data. Some of the relevant terms and related concepts include [8], [9], [10]:

1. **Collaborative Learning:** This term is occasionally used synonymously with FL. It primarily denotes a scenario where multiple parties collaborate in a learning task without sharing the raw data, emphasizing the cooperative aspect of model training.

2. **Distributed Machine Learning (DML):** A superset of FL, DML addresses the broader idea of dispersing machine learning computations across multiple nodes. The distinguishing factor for FL within DML is its explicit emphasis on data privacy and decentralization.

3. **Edge Learning or Edge Training:** This concept aligns with the tenets of FL but accentuates the training of machine learning models at the network's periphery, such as on mobile devices or IoT endpoints, rather than a centralized server.

4. **On-Device Learning:** This term foregrounds the locale of the learning process, typically a device like a smartphone, which undertakes learning tasks without offloading raw data to a primary server. It encapsulates a key aspect of the FL paradigm.

5. **Decentralized Machine Learning:** A term that spans a wide spectrum, it includes any machine learning technique not tethered to a central authority or server. FL, with its distinct focus on decentralized data sources and privacy preservation, nestles within this overarching concept.

6. **Privacy-Preserving Machine Learning:** An umbrella term, it encompasses methodologies aimed at safeguarding data privacy during the machine learning process. FL is one of its pillars, but the domain also integrates other techniques such as Homomorphic Encryption, Secure Multi-Party Computation (SMPC), and Differential Privacy.

In summation, while FL stands out for its unique combination of decentralized learning and privacy preservation, the academic landscape is replete with overlapping and adjacent terminologies. As research in this domain intensifies, a clear demarcation of these terms will aid in reducing ambiguities and fostering clearer dialogues in both academic and applied contexts. In this dissertation, our primary focus will be on delving deep into the nuances, challenges, and potential of FL.

1.4 Thesis Overview

This thesis embarks on an expansive journey to unravel the complexities, potentials, and challenges intrinsic to FL.

In **Chapter 1**, we lay the foundation by offering a concise introduction to the underpinnings of FL. This sets the stage by providing readers with fundamental definitions and essential contexts, ensuring a solid grasp of the overarching themes and concepts discussed throughout the thesis.

Chapter 2 is dedicated to presenting the broader landscape of FL, delineating its numerous opportunities and applications across diverse sectors such as mobile environments, organizational contexts, and the burgeoning world of the Internet of Things (IoT). Moreover, this chapter throws light on the inherent challenges posed by the decentralized nature of FL, serving as a reflection on the present state and potential future trajectories.

Chapter 3 offers a deep dive into the hallmark characteristics of FL, demarcating its unique stance against traditional centralized learning methods. The chapter meticulously delves into facets such as data privacy—with an in-depth exploration of mechanisms like Differential Privacy and Homomorphic Encryption—and personalization strategies that elevate user experiences in FL contexts. Moreover, a dedicated discussion on Non-IID data elucidates the inherent challenges and strategies for effective data processing within federated systems. The chapter concludes with a thorough exploration of FL's design space, touching upon critical elements from data distribution strategies to ensuring robustness in distributed learning.

Transitioning into **Chapter 4,** the narrative shifts to address the alignment and interplay between conventional machine learning models and FL. This chapter embarks on categorizing the various forms and structures of FL, dissecting methodologies ranging from Horizontal and Vertical FL to nuanced approaches like Federated Transfer Learning. An integral component of this chapter is also the analysis of potential threats specific to FL, offering a holistic view of both opportunities and vulnerabilities.

Chapter 5 moves into the technical heart of the domain, meticulously detailing the algorithms and frameworks pivotal for the implementation of FL. From the renowned Federated Averaging (FedAvg) algorithm to the avant-garde tools and frameworks such as TensorFlow Federated and PySyft, this chapter serves as a compendium for any researcher or practitioner looking to delve into the mechanics of FL.

In **Chapter 6,** we confront the potential threats and vulnerabilities within the FL environment. Through a systematic breakdown, this chapter elucidates various attack strategies, ranging from Model Inversion and Data Poisoning to the more complex Sybil attacks. For each identified threat, the implications, consequences, and defense mechanisms are extensively covered, ensuring a well-rounded perspective on security in FL.

With **Chapter 7**, we draw the narrative to a close, synthesizing the multifaceted discussions from previous chapters to offer a concluding perspective on FL. This culmination serves to highlight both the transformative potential of FL and the continuous need for research, innovation, and vigilance.

In essence, this thesis endeavors to provide a panoramic view of FL, guiding readers through its complexities, innovations, and future horizons. The emerging domain of FL offers a transformative approach to machine learning, advocating for decentralized data processing while respecting individual data privacy. By fostering a deep understanding of the field, it is our hope to inspire continued exploration and advancements within this dynamic realm of machine learning and artificial intelligence.

Chapter 2: General Opportunities, Challenges & Applications

In the evolving landscape of technology, recognizing opportunities, understanding challenges, and identifying potential areas of application becomes imperative. Chapter 2 delves deep into this triad, beginning with an exploration of the general opportunities that emerging technologies present. It then ventures into specific areas of application, shedding light on how modern innovations, from smartphones to organizational infrastructures and the burgeoning domain of the Internet of Things (IoT), are shaping the way we interact with the world. However, with every opportunity comes its set of challenges, and this chapter does not shy away from elucidating those. By the end of this chapter, readers will gain a holistic perspective on the interplay of opportunities, applications, and the challenges that lie therein, setting the stage for the subsequent chapters that further dissect each aspect.

2.1 Opportunities

FL, an emerging paradigm in the realm of machine learning, promises to reshape the landscape of data analytics by enabling model training across multiple devices or servers while keeping data localized. This decentralized approach holds the potential to revolutionize various sectors by offering more robust data privacy and efficient learning. Recent research from renowned sources elucidates the vast opportunities presented by FL. Several comprehensive surveys and studies in literature further elaborate on the opportunities of FL and cementing its role in the future of Machine Learning and Data Science [11], [12].

Charting FL's trajectory, it is evident that its journey is nascent yet promising. Overcoming impediments linked to data stewardship, communicative overheads, and bolstering security remain integral. Here, the blueprints provided by visionaries like McMahan [1] and Kairouz et al. [13] illuminate the path forward, emphasizing that transcending these challenges will spur FL's infiltration across a multitude of sectors.

Enhanced Security & Privacy

In today's age, data privacy has become a paramount concern, and rightly so, given the multiple data breaches and misuse we've witnessed over the years. FL offers a refreshing approach to this issue. Since raw data stays confined to the user's device, the sanctity of data is inherently maintained. This attribute of FL is especially commendable when you consider applications in sensitive domains like healthcare, where patient records can be confidential, or personal communications, which are ripe with intimate details. When the raw data does not venture outside its origin, it stands to reason that such data is far less vulnerable to unauthorized access or breaches [14], [15].

The allure of centralized data repositories, teeming with vast datasets, for malicious entities is undeniable. Such repositories often become targets, posing substantial security risks. FL, with its decentralized ethos, disrupts this vulnerability paradigm. By keeping data localized, the repercussions of potential breaches are contained. In the unlikely scenario where a device is compromised, the broader integrity of the system remains unaffected, showcasing the robustness of FL's architecture.

In all conventional systems, sharing data between entities often poses risks of data breaches or unauthorized access. However, by limiting data exchange to just model parameters, ideally encrypted, FL curtails these risks considerably. As a result, FL promises that users and organizations will be able to participate without the fear of compromising sensitive information. This paradigm of operational transparency [16], combined with rigorous data protection mechanisms, earns the trust of data contributors. The insights offered by Sikandar et al. [3] resonate with the potential and growth of this model, emphasizing the transformative impact it can have on machine learning practices and the broader data-driven industries. FL, as a methodology, places data privacy and security at its core, propelling a new era in distributed machine learning.

More importantly, sensitive sectors, such as healthcare and banking, operate under strict regulatory environments due to the critical nature of the data they handle. In such industries, even minor breaches can lead to significant consequences, both legally and reputationally. FL offers a solution, tailored to their needs. It provides an infrastructure where insights can be drawn from data without ever moving or exposing the raw data [11], [17]. Thus, hospitals can benefit from shared medical research without revealing patient identities, and banks can enhance fraud detection systems without compromising account details. The dual capability of FL to offer data-driven insights while maintaining data sanctity makes it an invaluable asset for these sectors [18].

Reduced Data Transfer Costs

Centralized cloud-based training models have traditionally been marred by exorbitant data transfer costs. Shuttling vast datasets between devices and central servers not only incurs financial implications but also has environmental consequences. FL offers a respite from this model. By emphasizing local data processing and only transmitting model updates, the volume of data transfer is dramatically diminished. This reduction translates to substantial cost savings and, by extension, reduces the carbon footprint associated with data transfer, thus presenting a more sustainable model in the face of global ecological concerns [19], [20].

Improved Model Personalization

The evolving digital landscape has rendered the generic, one-size-fits-all approach obsolete. Today's consumers, equipped with a plethora of choices, yearn for tailored experiences. FL is perfectly poised to cater to this demand. By continually refining models based on individual device data, FL crafts a learning model that is inextricably linked to user behavior. As these models evolve, they intuitively align with the user's preferences, resulting in experiences that are not just personalized but deeply resonant. This level of customization is paramount in ensuring user retention and satisfaction in an increasingly competitive market [21], [22].

Scalability

At its core, FL is designed for expansion. Its structure, which encourages the addition of devices to its network, is inherently scalable. Each device that becomes part of this network augments the collective intelligence without ever directly sharing its raw data with a centralized entity. This architecture ensures that as the network grows, so does its diversity and richness in data, all while maintaining a lightweight and efficient structure [23].

Utilization of Edge Devices

The fringes of our interconnected digital ecosystems are populated with devices, ranging from smartphones to IoT gadgets, that often brim with untapped computational potential. These edge devices, despite their capabilities, are frequently overlooked in traditional models. FL redresses this oversight by actively incorporating these devices into its learning matrix. By doing so, it not only optimizes computational resources but also democratizes the learning process, ushering in a more inclusive era of machine learning [14], [24].

Real-time Learning

The dynamic nature of certain sectors, such as autonomous driving or emergency medical response systems, necessitates real-time data processing. FL, with its real-time data processing capabilities on individual devices, addresses this need adeptly. Unlike traditional batch updates that might introduce latency, FL ensures models evolve in real-time, adapting promptly to emergent data. This rapid adaptation is crucial in sectors where even minute delays can have consequential implications [25].

Decentralization and Robustness

Centralized systems, despite their efficiency, often suffer from a critical flaw: they possess points of vulnerability that can be exploited. A single malfunction can jeopardize the entire system. FL, in its essence, is decentralized, thereby eliminating single points of failure. Its interwoven structure ensures that even if individual nodes encounter issues, the overarching system continues to function seamlessly, embodying true resilience [6], [26], [27].

Regulatory Compliance

In a world punctuated by stringent data protection norms, ensuring compliance is paramount for businesses. These regulations, designed to protect user data, often introduce operational challenges. FL emerges as a frontrunner in addressing these concerns. Its emphasis on data localization ensures that businesses can operate within regulatory confines without compromising on data-driven insights, making it an invaluable tool in today's regulatory landscape [15].

Optimized Network Traffic

Network congestion, primarily driven by voluminous data transfers, often leads to inefficiencies and latency. FL, by its very design, ameliorates this issue. By focusing on the transmission of model updates rather than extensive raw data sets, it ensures network traffic remains streamlined. This lean approach guarantees optimal bandwidth utilization, ensuring consistent operations devoid of unnecessary lags [28].

The vast potentialities and myriad benefits of FL have been extensively elaborated upon, underscoring its transformative role in the data analytics landscape. Yet, its true impact becomes palpable when observed through the lens of practical applications across diverse sectors. As we transition to the next section, we will embark on a journey through various application areas where FL has not only showcased its efficacy but also revolutionized operational paradigms. From healthcare to finance, from smart cities to autonomous driving, FL's footprint is expansive. Let's delve deeper into these sectors to unearth the tangible changes brought about by FL, offering a holistic understanding of its real-world implications.

2.2 Areas of Application

The potential of FL has spurred innovations across a wide range of application domains. We begin by exploring the broad-based applications of FL, touching upon ubiquitous platforms like smartphones and the expansive realms of organizations and the Internet of Things (IoT). Following this, we briefly highlight more niche areas of application. This compilation is by no means exhaustive, but it offers a glimpse into the vast landscape where FL can make a transformative impact.

2.2.1 Smartphones

The omnipresence of smartphones in our daily lives has heightened the necessity for robust data privacy measures. FL, with its decentralized approach to data processing, offers a revolutionary solution tailored to smartphones.

Applications such as predictive texting are fundamental examples where FL can make a profound difference. Such apps can leverage user datasets without compromising data privacy. Streaming apps can also harness FL's capabilities to suggest movies or songs, ensuring that the data never leaves the user's device [29], [30], [31].

Ek et al. [31] go into depth regarding FL’s application in the mobile domain. They discuss how FL not only ensures data privacy but also enhances system efficiency by sending only model updates to central servers. This means users enjoy prolonged battery life and reduced data consumption, further strengthening the case for FL in mobile contexts.

Moreover, with the increasing integration of Augmented Reality (AR) and Virtual Reality (VR) applications on smartphones, there's an even greater volume of sensitive data that can be processed using FL. These immersive technologies can benefit from personalized user interactions without jeopardizing privacy.

2.2.2 Organizations

Organizations, particularly in sensitive sectors like healthcare and finance, are perpetually in search of innovative data management solutions. FL offers such organizations a fresh, secure lens through which they can view and handle data.

Consider hospitals, which are repositories of vast amounts of confidential patient data. Rather than resorting to centralized storage systems fraught with risks, FL provides a decentralized modeling approach. This way, medical institutions can derive invaluable insights without ever exposing individual data. Such decentralized processing is pivotal, especially when one acknowledges the stringent data protection regulations in place like GDPR and HIPAA [15], [18].

The financial sector, laden with heaps of critical data from transactions to credit histories, can employ FL to circumvent the pitfalls of traditional data processing. Banks can synergize with other entities without ever having to share raw data, leading to enhanced model training and more nuanced predictions, especially in areas like credit assessments.

As global finance becomes more interconnected with emerging technologies like blockchain and cryptocurrency platforms, FL's decentralized methodology ensures seamless collaboration with reduced risk of data breaches [32].

2.2.3 Internet of Things (IoT)

The IoT universe, with its vast network of interconnected devices, is a goldmine for FL applications. Devices ranging from wearable health tech to smart home systems can vastly improve their functionality through decentralized learning.

Zhang et al. [12] provide a comprehensive overview of merging FL with IoT. They discuss the significant security benefits achieved through data decentralization. Beyond security, devices in the IoT realm that utilize FL can be more efficient and adaptive, thanks to real-time updates and collaborative learning across the network.

With the rapid proliferation of smart cities, where traffic management systems, energy grids, and public services are interconnected, there's a compelling case for FL's role in optimizing these complex systems. This ensures not only improved services but also enhanced security in urban environments [29], [32], [33].

2.2.4 Healthcare

The healthcare sector has historically been at the intersection of data-driven insights and ethical considerations. The Health Insurance Portability and Accountability Act (HIPAA), among other regulations, places stringent conditions on the dissemination and sharing of protected health information, emphasizing patient confidentiality [34]. FL provides an innovative solution to this conundrum. Rather than accumulating data in a central repository, which poses risks of breaches and unauthorized access, FL allows for the development of sophisticated AI algorithms directly on healthcare databases and devices [35]. The efficacy of this approach lies in its dual achievement: firstly, the patient data remains localized, preserving its integrity, and secondly, by pooling insights—not raw data—researchers and medical professionals can extract industry-wide patterns, facilitating improved patient care and treatments [30].

2.2.5 Personalized Advertising

The digital era's hallmark is arguably its personalized user experience. Algorithms curate advertisements, product suggestions, and content to fit individual preferences, enhancing user engagement [36]. This customization, however, is predicated on accessing vast swathes of user data, leading to mounting concerns over privacy breaches and data misuse. FL offers a resolution. By decentralizing the learning process, advertising platforms can glean user insights without directly accessing the granular data. This ensures that personalization remains robust while respecting user privacy, heralding a shift in digital advertising strategies [13], [37], [38], [39].

2.2.6 Automotive Sector

Autonomous vehicles represent not just the zenith of automotive engineering but also the pinnacle of real-time, data-driven decision-making. Traditional machine learning approaches necessitate the collection of data to a central hub for processing and model training. FL diverges from this by facilitating on-device training, enabling vehicles to learn from immediate environments and traffic conditions. Preliminary research suggests the potential for FL to revolutionize tasks such as predictive wheel steering, by significantly curtailing training durations [40], [41], [42]. As research progresses, the automotive industry may witness an accelerated transition to more efficient and safer self-driving vehicles.

2.2.7 Defense Against Financial Fraud

The digital age, despite its manifold benefits, has been paralleled by a surge in financial malpractices, ranging from credit card frauds to intricate money laundering operations [43]. In response, the banking sector is in dire need of robust, predictive models to preempt these illicit activities. FL emerges as a potent tool in this context. Traditional models necessitate data centralization, a risk in sensitive financial sectors. FL sidesteps this by allowing banks to collaboratively train models without sharing raw transactional data [44]. This approach not only bolsters the predictive accuracy of fraud detection systems but also reinforces customer trust in financial institutions.

2.2.8 Insurance and FL

At the heart of insurance operations lies risk modeling, an endeavor that relies heavily on diverse data sets, from medical records to financial histories [45], [46], [47], [48]. With escalating concerns around data privacy, insurance providers face the challenge of optimizing risk predictions without infringing on client confidentiality. FL offers a pathway. By enabling multi-party computations and insights sharing, without the direct transfer of raw data [23], insurance companies can refine their risk models. This ensures that while the industry progresses in its predictive capacities, individual data rights remain uncompromised.

2.2.9 Military and Police forces

Law enforcement agencies are leveraging FL as a powerful tool to enhance their machine learning models. By training collaboratively without sharing raw data, agencies ensure data privacy while achieving better model accuracy, translating to fewer false positives and negatives. This not only addresses critical challenges in financial crime detection but also potentially reduces operational costs, allowing for more strategic resource allocation [16].

In the realm of military operations, where timely and secure information exchange is pivotal, traditional centralized machine learning presents challenges. The military's adoption of FL shifts the focus from central servers to local storage on edge devices. This ensures more secure, efficient data exchange through tactical servers, reducing latency and optimizing bandwidth. Particularly in high-stakes situations, the efficiencies introduced by FL prove crucial. As data security becomes paramount in the modern era, FL stands out as an innovative solution in both law enforcement and military sectors, promoting effectiveness without compromising on data ethics [16].

2.3 Main Challenges

While FL stands as a harbinger of the next wave in artificial intelligence and machine learning, its transition from theory to practice is not without obstacles. The decentralized design, which emphasizes data privacy and distributed learning, is a marked departure from traditional, centralized machine learning models, introducing novel intricacies at every stage. Pioneering research has been instrumental in outlining the numerous challenges that mark the FL terrain, and it's imperative to understand these in detail if we are to harness the full potential of FL.

While FL offers promising outcomes, it's a departure from the familiar, conventional methodologies. This novelty, though exciting, brings in layers of complexity, especially in terms of implementation. Techniques that worked seamlessly in centralized systems may no longer be directly applicable. The expansion of FL across large and diverse networks brings forth complexities related to synchronization, data homogenization, and real-time collaboration. Additionally, the varied capabilities and implicit failures of devices, as discussed previously, add to the intricacies. To harness the full potential of FL, researchers and practitioners need to innovate, adapt, and develop new strategies tailored for this unique model. [7], [11], [12], [32], [33], [42]

The inherent design of FL promotes data security. However, the very nature of distributed systems introduces multiple points of potential vulnerability. Every interaction, every data exchange, regardless of how minimal, needs to be fortified against breaches. Integrating Secure Multiparty Computation (S.M.C.) with F.L. is one of the ways to enhance privacy and security in federated settings. However, consistently encrypting these interactions without affecting performance poses a challenge. As also noted by Zhang et al. (2021), the terrain of FL security, though advanced, still has pockets that remain unexplored and warrant attention to ensure holistic security [49], [50].

In a world characterized by diverse data sources, ensuring that FL models deliver consistent and accurate results is challenging. Variabilities in data quality, distribution, and computational capabilities can influence model performance. The inherent diversity among devices, including statistical variations in data sources, further complicates the landscape. While techniques such as data preprocessing and augmentation offer some respite, the inherent challenges of FL—like maintaining privacy while ensuring large-scale optimization—present a complex puzzle. Ensuring consistent model efficacy across such a varied landscape necessitates ongoing research and adaptation [51], [52].

Furthermore, adapting methodologies to the unique demands of specific projects, managing non-Independent and Identically Distributed (non-IID) data, and optimizing communication are just some of the hurdles to be crossed. However, as research intensifies, solutions to these challenges will inevitably emerge, paving the way for broader FL adoption [53], [54], [55].

2.3.1 Communication and System Heterogeneity

Communication is the bedrock of any distributed system, and FL is no exception . FL is a much-needed technology in this golden era of big data and Artificial Intelligence, due to its vital role in preserving data privacy, and eliminating the need to transfer and process huge amounts of data, while maintaining the numerous benefits of Machine Learning. As opposed to the typical central training process, FL involves the collaborative training of statistical models by exchanging learned parameter updates. However, wide adoption of the technology is hindered by the communication and computation overhead forming due to the demanding computational cost of training, and the large-sized parameter updates exchanged. In popular applications such as those involving Internet of Things, the effects of the overhead are exacerbated due to the low computational prowess of edge and fog devices, limited bandwidth, and data capacity of internet connections. Over the years, many research activities that target this particular issue were conducted but a comprehensive review of the fragmented literature is still missing. This paper aims at filling this gap by providing a systematic review of recent work conducted to improve the communication and/or computation efficiency in FL [12], [33], [51], [52], [56], [57].

Communication costs are the principal constraint, and we show a reduction in required communication rounds by 10-100x as compared to synchronized stochastic gradient descent. However, FL operates in an environment marked by diverse systems and protocols. As illustrated in Figure 1 and 2, managing the intricate process of interactions within such a heterogenous landscape is daunting. The variability in storage, processing power, and communication capabilities among federated components adds to this complexity. Each system might have its own limitations, capacities, and quirks, which can affect the overall synchronization and efficiency. While solutions like model compression and decentralized training, as suggested by Almanifi (2023), provide some answers, implementing these solutions seamlessly across a diverse network remains a challenge [51].

2.3.2 Threats and Adversarial Attacks

Most digital systems have vulnerabilities and FL, despite its advanced architecture, is not exempt. FL faces risks from adversarial attacks. [58], [59]. These attacks are sophisticated attempts to deceive or manipulate the learning process. The decentralized nature of FL makes it imperative to have robust defense mechanisms in place. Techniques such as Secure Function Evaluation, Homomorphic Encryption, and Differential Privacy, can be applied in federated settings to provide privacy and combat threats. As highlighted by Sikandar et al. (2023), these methods offer potential defenses. However, the arms race between defense mechanisms and adversarial techniques is ongoing, requiring constant vigilance and innovation. [60]

2.3.3 Infrastructure and Bandwidth Issues

For FL to function seamlessly, robust infrastructure is a prerequisite. However, challenges related to unstable networks, limited bandwidth, especially in edge devices, can pose significant barriers. These constraints can delay or inhibit model updates and training, potentially affecting real-time decision-making and overall system performance [24].

In conclusion FL is poised to redefine the contours of AI and machine learning. Its promise of enhanced data privacy, coupled with its potential to transform sensitive industries, sets it apart. However, like any transformative technology, it comes with its set of challenges. As researchers and practitioners navigate this landscape, their endeavors will shape the future trajectory of AI, with FL as a central pillar.

Chapter 3: FL & Centralized Learning: Comparison and Classification

In the evolving landscape of machine learning, FL emerges as a distinctive paradigm, replete with its methodologies and modalities. This chapter ventures into the categorical facets that define FL. Initially, we navigate the confluence between traditional machine learning techniques and their federated counterparts, shedding light on linear models, tree structures, neural networks, and the relatively newer domain of reinforcement learning. Subsequently, we disentangle the various forms of FL itself, dissecting the distinctions and synergies between Horizontal FL, Vertical FL, Federated Transfer Learning, and Federated Reinforcement Learning. By demystifying these classifications, this chapter aims to provide readers with a clearer map of the terrain, offering a structured perspective that elucidates the breadth and depth of FL's applications and methodologies..

3.1 Comparison with Traditional/ Centralized Machine Learning

Centralized and FL represent two distinct paradigms within the machine learning field, each presenting its unique benefits and challenges.

Centralized learning is a conventional approach to machine learning wherein all data required for training a model is amassed and housed in a central server or database. In this paradigm, data from varied sources or sensors is transmitted to a singular, centralized location. Once housed in this server or data center, a machine learning algorithm taps into this reservoir to train the model. Through iterative processes, the algorithm adjusts model parameters to minimize errors, refining its understanding of the data patterns. After the model is adeptly trained, it's primed for deployment, ready to make predictions on new, unseen data. This method stands in stark contrast to decentralized models like FL, where data remains anchored at its origin, and the training unfurls in a distributed manner, obviating the need to ferry raw data to a centralized hub. While the centralized model boasts advantages like simplified data access and potentially accelerated training speeds given the right computational firepower, it grapples with challenges. These range from data privacy concerns and data transfer overheads to the vulnerabilities introduced by a potential single point of system failure.

Conversely, FL adopts a decentralized methodology. Instead of sending raw data to a primary server, models are trained locally on individual devices or "nodes." Once the local training is complete, only model updates or essential insights—rather than raw data—are shared with the central server. Li B. et al. (2020) highlight that this design naturally offers enhanced security, especially in scenarios where data privacy and limited connectivity are significant concerns [28]. A primary advantage of FL is its inherent data privacy. Since data remains on its original device, there's a marked reduction in the risk of exposing sensitive details during the model training process. This aspect makes FL especially advantageous in sectors handling confidential data, like healthcare or finance, where rigorous privacy regulations are enforced.

In their paper, McMahan et al. (2017) explain that in centralized learning, all training data from various sources is gathered and processed by a central server to train a comprehensive model. This method offers easy access to vast quantities of data, enabling the use of robust algorithms and significant computational resources [28]. Nonetheless, centralized learning brings up concerns about data privacy, potential security breaches, and the requirements for transmitting data to a central hub.

Furthermore, central to FL is the principle of data minimization, which is aptly represented in Figure 3. Traditional systems often hoard more data than they actually need, posing unnecessary risks. FL, on the other hand, operates under the principle of using the least amount of data necessary to achieve the desired results. This approach not only ensures optimal privacy but also streamlines computational processes. In the age of information overload, this ability to discern and operate on the most relevant data efficiently, as emphasized by Zhang et al. [17], is a game-changer. It reorients the focus from data quantity to data quality and relevance, driving efficient and effective outcomes.

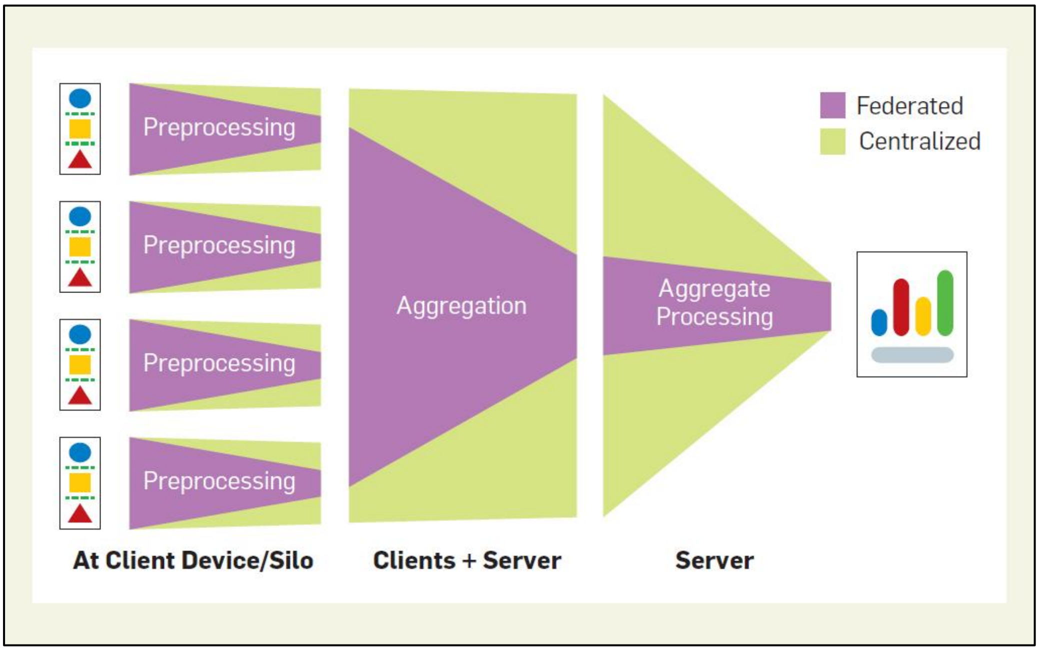


Figure 3 - Data minimization in federated vs. centralized approaches [61]

Furthermore, FL's scalability and efficiency are noteworthy. Distributing the learning process across multiple devices lets FL manage expansive datasets without centralizing the data. Bonawitz et al. (2019) emphasize that this decentralized approach minimizes necessary communication. Only model updates or summarized insights are transmitted, making FL optimal for scenarios with constrained bandwidth or sporadic connectivity [23]. However, it's essential to juxtapose this with the advantages centralized learning accrues from accessing extensive centralized datasets. The centralization allows the employment of powerful algorithms and ample computational resources to cultivate high-performing models. This strategy becomes particularly beneficial when concerns about data privacy are minimal, and there aren't stringent data-sharing prohibitions.

In summation, while federated and centralized learning each present unique strengths, they are best suited for different environments. FL stands out in situations where data privacy, security, and connectivity constraints are paramount. It facilitates collaborative model training without the necessity to share raw data. In contrast, centralized learning, with its access to vast centralized datasets, is apt for scenarios where concerns about data privacy are less significant. The selection between these approaches should consider the operational environment, data characteristics, and existing privacy restrictions.

Understanding the trajectory of FL requires a foundational grasp of traditional machine learning models. These models have over the years laid the groundwork, shaping the machine learning landscape and serving as the precursor to the emergence of FL. The following sections present in brief the primary machine learning paradigms and their most important relations to FL, setting the stage for subsequent discussions on FL.

3.1.1 Linear Models

Linear models are historically significant. Their hallmark lies in their sheer simplicity, computational frugality, and ability to allow humans to interpret their results. These models presuppose a linear relationship between the input and output. Famous among these are Linear Regression, which is essentially fitting a straight line to data, Support Vector Machines (SVM) that create decision boundaries, and Linear Discriminant Analysis (LDA) which is great for dimensionality reduction.

Linear models make predictions by finding relationships in the input features and a continuous target output. They assume that this relationship is linear, which means that a change in the input features results in a proportional change in the output. More specifically:

- **Linear Regression:** This is the most basic form of linear modeling. It tries to fit a line (in 2D), plane (in 3D), or hyperplane (in higher dimensions) to the data points. The best fit line is determined by minimizing the sum of squared differences between the actual and predicted values [62].
- **Support Vector Machines (SVM):** In a binary classification problem, SVMs aim to find the best hyperplane that separates the two classes. The "best" hyperplane is the one that maximizes the margin between the two classes. It's important to note that SVM can be extended to non-linear separations using the kernel way [63], [64].
- **Linear Discriminant Analysis (LDA):** LDA is used for dimensionality reduction and classification. It finds the linear combinations of features that best separate two or more classes in the dataset [65].

Yet, as insightful as they are, linear models have their set of constraints. When we venture into the world of localized data, which might differ greatly from a global data perspective, these models start to waver. This inconsistency in performance, as underscored by several authors

[66], [67], brings to light the risk of models overfitting to certain datasets while neglecting others. This is where FL, offers promise. It seeks to bridge the gap between varying data distributions, allowing for a more harmonized training approach across a spectrum of data sources.

In understanding the significance and nuances of linear models, it's imperative to appreciate both their strengths and limitations. They've been indispensable for many applications due to their transparency and ease of interpretability. When you're trying to understand the weightage and importance of each feature in your data, the coefficients in a linear model can provide clear insights. This transparency makes them valuable for applications where decisions need to be understood and justified, like in finance or healthcare.

Furthermore, due to their simplicity, linear models are computationally efficient. They do not require the extensive computational power or the storage capacities that some more complex models demand. This makes them especially relevant for applications that have limited computational resources or need quick results [67].

However, the simplicity of linear models is both their strength and their Achilles heel. The real world, with its myriad complexities, doesn't always operate linearly. While a linear approach works well when relationships in the data are straight-forward, they struggle when data possesses non-linear intricacies. Their assumption that a change in the input results in a proportional change in the output might oversimplify many real-world scenarios.

Consider, for instance, a situation where data is clustered into local subgroups, each with its unique characteristics. A global linear model might find a general "average" trend that doesn't accurately reflect any specific subgroup. In such situations, the model might perform adequately on average but poorly on specific subsets of data. The performance inconsistency is further exacerbated when there's a significant disparity between these local data distributions.

FL comes into play as a potential panacea to this challenge. At its core, FL is about training on localized data while amalgamating insights globally. Instead of trying to find a one-size-fits-all model, FL respects the uniqueness of each data source, allowing models to learn from diverse and varied distributions. The resultant model is more holistic, capturing insights from a broad spectrum of data while avoiding the pitfalls of overfitting to any specific subset.

Moreover, FL's approach can augment linear models in unique ways. By training linear models on local devices and then aggregating the learned parameters centrally, it's possible to achieve a balance. Each local model can capture the nuances of its specific data, while the global aggregation ensures that a broader perspective is retained. This amalgamation of local insights and global perspective can lead to more robust and accurate linear models, especially in heterogeneous data environments.

In the grand mosaic of machine learning, linear models represent but one piece. As we delve deeper into the interplay between traditional machine learning and FL, we'll uncover how more complex and nuanced models fit into this evolving narrative and the unique challenges and opportunities they present in a FL context.

3.1.2 Tree Models

Tree models, comprising algorithms like decision trees and random forests, present a more dynamic approach to machine learning. Their structural design allows for easy interpretation, but they too face challenges in the FL setting, particularly when faced with heterogeneous data distributions.

Tree-based models recursively split the data based on certain criteria (like Gini impurity or entropy) until they reach a predefined stopping condition. The most common Tree models are:

- **Decision Trees:** The decision tree algorithm tries to solve the problem by making decisions based on asking multiple questions. For instance, for a dataset of animals, a question could be "Does it have feathers?". Based on the answer (Yes/No), the tree further splits and asks additional questions until it can make a prediction.
- **Random Forests:** Random forests build multiple decision trees and merge their outputs for a final decision, either by taking a majority vote (classification) or by averaging (regression). The randomness comes from two aspects: random subsamples of data for building each tree and random subsets of features considered for splitting.

It is important to note that while tree learning models are used in FL, there can be challenges in handling miscellaneous data distributions and point representations through bias. Forms of data preprocessing, point engineering, or adaptive learning strategies are employed to address these challenges and improve the performance of tree-based models in FL environments. Tree learning models, similar to decision trees and arbitrary timbers, can be used in FL to perform machine learning tasks on distributed data while retaining data leakages. FL helps in decoding. Previously in practice, clients can hide their data, and there is less chance of any breach. Zhao et al. (2018) assume that a system protected by data breaches is reinforced through tree models. Similarly, there is the possibility of a double group of tasks. However, they can associate your information without it leaking on the network, if there are more than two clients connected to a network. Automatic learning has not experienced certain advances in recent times and thanks to FL, clients can take advantage of its functions intimately.

Apart from this affirmation, there is an advance in security. Chang et al. (2021) state that perpendicular and vertical partition data will increase security. This momentum will allow the data to spread out in one dimension rather, multiple clients can weigh in on it. With similar information, there will be a comprehensive FL which, again in the future, would bolster the data operation effectively [68].

Decision Trees

Decision trees are one of the most intuitive machine learning algorithms. At their core, decision trees split the dataset based on feature values, essentially asking questions about the data, and making decisions based on the answers to those questions. Each internal node represents a "question" on an attribute (e.g., "Is feature A greater than a threshold?"), each branch represents the outcome of that question, and each leaf node represents a prediction. The primary advantage of decision trees is their simplicity and visual interpretability. They can easily handle both numerical and categorical data, making them versatile for a range of applications [68].

In a FL environment, individual devices or Edge Servers can train their local decision trees on their specific datasets. Zhao et al. (2018) introduced the Gradient Boosted Decision Tree (GBDT) in this setting. It is especially noteworthy for its enhanced security and capability to manage concurrent processing. By amalgamating regression trees from multiple data stakeholders, a synchronized FL system emerges. Each participant focuses on their local datasets, deriving predictions and bypassing the need for raw data exchange. These local models are then compiled by a central server to establish a globally coherent decision tree, seamlessly integrating diverse data insights [69].

Random Forests

Random forests are an ensemble method that harnesses multiple decision trees to bolster prediction accuracy and control overfitting. The underlying principle is simple: by leveraging the diversity of multiple trees, the model can capture intricate structures in the data more effectively than a single tree. Each tree in the forest is trained on a subset of the data, introducing randomness in two main ways - through bootstrapped datasets and random feature selection. The final prediction is an aggregation of predictions from all the trees, typically through majority voting or averaging [70].

When applied in FL, the distributed nature of random forests becomes even more relevant. Each participating entity, be it a device or an edge server, constructs a local decision tree using its inherent data . As highlighted by Haffer et al. (2023), these forests serve as an intermediary, playing a protective role against data intrusion [71]. The collective knowledge is compiled, ensuring the model's integrity remains intact. The central server manages the aggregation process, collating local decision trees into a global random forest model, preserving privacy at each juncture.

3.1.3 Neural Network Models

Neural networks have become the foundation for many advanced machine learning tasks due to their adaptability and ability to learn complex patterns from data. Depending on the type of data and task at hand, different architectures are utilized. Within the realm of FL, various neural network architectures are tailored to best suit the unique requirements of the decentralized data distribution.

3.1.3.1 Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs), since their inception, have radically redefined the landscape of computer vision. These neural structures, meticulously designed for data in matrix formats like images, have become the cornerstone for myriad applications. Whether it's the facial recognition system unlocking a smartphone, satellite image analyses for climate studies, or the algorithms helping self-driving cars interpret their surroundings, the genius of CNNs is palpable [72], [73].

Their groundbreaking impact is owed to their ability to naturally process visual data. While traditional machine learning models often saw images as flat arrays of pixels, CNNs interpret them in a more intuitive manner, recognizing spatial hierarchies and patterns. This recognition of spatial relationships, coupled with their ability to learn hierarchically from raw pixel data to abstract features, sets them apart in the realm of image-centric tasks [72].

CNNs, at their core, operate through a sophisticated symphony of layers, each playing its role in extracting and refining information from the input data. The convolutional layers, using sets of learnable filters, traverse the image, detecting spatial patterns. In the earlier stages of a CNN, these filters might pick up basic visual elements such as edges or color gradients. As we delve deeper into the network, the patterns recognized grow in complexity—from simple edges to shapes, and then from shapes to more complex structures like an eye or a wheel.

Complementing the convolutional layers are the pooling layers, specifically max-pooling, which downsample the spatial dimensions while preserving the most salient information. This is akin to compressing an image without losing its essence. Once the data passes through these stages, it encounters one or more fully connected layers, transforming the processed data into a format suitable for tasks at hand, be it classifying an image into categories or identifying objects within it [74].

The realm of FL, marked by distributed datasets, often finds itself in need of architectures that can process decentralized image data with efficiency and accuracy. CNNs naturally fit this bill. Emphasizing their efficacy, He et al. (2019) observed significant enhancements in model performance when CNNs were interwoven with FL protocols [75]. Their ability to adeptly rank data in such environments ensures that the core essence of the data is preserved and learned. Observing this potential, leading technological entities, like Google, have been integrating CNNs into their FL frameworks. This widespread adoption is a testament to the unparalleled capability of CNNs in decentralized data scenarios.

3.1.3.2 Recurrent Neural Networks (R.N.N.s)

Recurrent Neural Networks (RNNs) represent one of the most transformative advances in the domain of deep learning, particularly when dealing with sequential and time-dependent data. This special architecture finds its roots in the recognition of a simple yet profound idea: not all pieces of data are independent; some carry the weight and context of their predecessors. From comprehending the nuances of spoken language in speech recognition systems to predicting stock market trends based on historical data, RNNs have solidified their place as an indispensable tool for sequential data analysis.

Expanding on their applications, RNNs have been instrumental in scenarios where data is sequential or has temporal dependencies. For example, in natural language processing, understanding the context of a word often depends on its preceding words. Similarly, in financial predictions, the potential future value of a stock might be influenced by its past performances. RNNs, with their ability to 'remember' previous data in a sequence, aptly serve these requirements [76].

At the heart of the RNN architecture is its cyclic connectivity. Unlike traditional feed-forward neural networks, where data flows in a singular direction, RNNs allow for feedback connections. What this essentially means is that neurons can send their outputs back as inputs in a looped manner. This looped feedback mechanism bestows upon RNNs a form of memory, enabling them to maintain a historical context. For instance, when processing a sentence, an RNN can retain information about earlier words while interpreting the meaning of a current word, thereby allowing for a more holistic understanding.

Delving deeper into their structure, each neuron in an RNN processes an input while also considering its previously computed output. This amalgamation of current input and past output helps in creating a chain of information, seamlessly linking past and present data points. Such a mechanism ensures that as the network processes new data, it always carries forward a trace of the past, offering context and continuity to its operations [77].

In the realm of FL, where data distribution is inherently decentralized and often sequential, RNNs showcase their true potential. Consider mobile devices that capture user typing patterns or wearable devices that track health metrics over time. Each piece of data in such scenarios is a link in a chronological chain, and understanding one often requires the context of the others.

Highlighting their practical significance, Fekri et al. (2022) discussed the utility of RNNs in distributed setups, like managing power grids where understanding past power consumption patterns is crucial for future load predictions [78]. Similarly, in distributed fiscal planning systems, understanding past budgetary constraints and expenditures becomes pivotal for future financial decisions. Such applications underscore the adaptability and potency of RNNs when integrated into FL landscapes.

3.1.3.3 Long Short-Term Memory (LSTM) Networks

LSTM networks, standing as a testament to the continuous evolution of neural network designs, were crafted to overcome a fundamental limitation of their predecessor, the standard RNN: the notorious vanishing gradient problem. This limitation often handcuffed RNNs, preventing them from efficiently processing long sequences and retaining long-term dependencies. LSTMs, with their advanced architecture, are not just an incremental improvement but a radical solution, preserving the memory of sequences far longer than conventional RNNs.

LSTMs, as their name suggests, have the duality of retaining short-term intricacies while not losing sight of the overarching long-term narrative within the data. This makes them the gold standard for a plethora of sequence-based tasks, from language translation to predicting financial market movements [77], [79].

What differentiates LSTMs from standard RNNs is their cell structure, a meticulously designed mechanism brimming with gates. These gates - input, forget, and output - are not just passive pathways. They're decision-making entities. The input gate decides the influx of new information, the forget gate judiciously chooses what to let go from the cell's memory, and the output gate controls the output based on the cell's current state. These gates work in tandem, ensuring the network discerns which information is vital to retain over long sequences and which is transient, thereby averting the loss of crucial contextual cues [77].

By maintaining this delicate balance of remembering and forgetting, LSTMs excel in tasks where understanding the context over prolonged sequences is paramount. Their ability to connect distant events in a sequence makes them adept at capturing patterns that might be elusive to other architectures.

The distributed nature of FL, often characterized by data spanning across devices and temporal instances, demands architectures capable of understanding intricate patterns over extended sequences. LSTMs stand out in such scenarios. The ability of LSTMs to process and retain extensive sequential information aligns perfectly with the needs of federated ecosystems, making them indispensable tools in such settings.

3.1.3.4 Generative Adversarial Networks (GANs)

Generative Adversarial Networks, or GANs, represent a paradigm shift in the world of generative modeling. Before their arrival, generating high-fidelity, realistic data samples was a daunting challenge. GANs, however, turned this challenge on its head, opening doors to possibilities ranging from creating artwork to simulating real-world scenarios for training models.The genius of GANs lies in their adversarial framework—a relentless contest between two networks that pushes each other to perfection. This dynamic has transformed how we think about data generation, leading to results that often blur the lines between synthetic and real [80].

The architecture of GANs can be described as an interplay between two neural networks: the generator and the discriminator. The generator, commonly parallelized as an artist, crafts data samples from scratch. Simultaneously, the discriminator, playing the critic, assesses the authenticity of these samples, distinguishing between genuine and generated. As training progresses, the generator hones its skills, trying to create samples so realistic that the discriminator can't tell them apart from real ones. Conversely, the discriminator sharpens its discernment, attempting to catch the generator's “bluff”. This tug of war results in a feedback loop of continuous improvement, ultimately leading the generator to produce data that's almost indistinguishable from genuine samples [81].

FL, with its inherent distributed nature, often requires mechanisms to understand and represent global data distributions without accessing the actual data. Herein lies the value of GANs. Their ability to generate synthetic data that mirrors the underlying distribution of real-world, distributed datasets can be instrumental. Such synthetic data aids the central server in understanding and modeling the data landscape without violating privacy norms, ensuring the global model remains robust and representative [82].

On the same time, GAN poisoning attacks present a significant threat to FL systems, wherein malicious actors introduce manipulated data to deceive the global model. Such attacks can subtly modify the aggregated model, leading to degraded performance or misclassifications, thereby compromising the integrity of the entire learning process [58], [83].

3.1.3.5 Transformer Networks

The advent of transformer networks signified a great leap in the realm of natural language processing (NLP). With their efficiency and accuracy, they haven't just edged past their contemporaries—they've established a new benchmark. Whether it's large-scale language models, machine translation, or text summarization, transformers have become the centerpiece of state-of-the-art NLP solutions.

Transformers, with their innovative architecture, bring a fresh perspective to processing sequences. Instead of relying on recurrent structures, they harness parallel processing, enabling them to handle vast amounts of data efficiently, making them a force to be reckoned with in the vast seas of textual data.

The idea of transformers emanates from their "self-attention" mechanism. At its essence, this mechanism enables each element in the input data to focus on different parts of the sequence dynamically, assigning varied attention scores. This means that every word, or token, in a sentence can be influenced by any other word, regardless of their relative positions. Such dynamic interplay ensures that the output for a particular word is informed by the entire context, not just its immediate neighbors [84].

Furthermore, transformers often employ multiple such attention heads, each capturing different types of relationships within the data. This multi-pronged approach results in outputs that are not only contextually rich but also nuanced, capturing the intricate interplay of elements in sequences.

In FL environments, where data is scattered and often massive, the power and efficiency of transformers are becoming increasingly crucial. Their capacity to process vast amounts of textual data in parallel, coupled with their unparalleled contextual understanding, positions them as the preferred choice for federated tasks. Whether it's sentiment analysis, language modeling, or any text-centric application, transformers promise consistent and top-tier performance, even in the face of distributed and diverse data sources [85], [86].

3.1.4 Reinforcement Learning

Reinforcement Learning (RL) stands as a distinct paradigm in machine learning, enabling agents to learn optimal strategies through interactions with their environment. This dynamic method of learning has roots in behavioral psychology, suggesting the strengthening of certain behaviors via feedback [87]. RL diverges from traditional supervised and unsupervised learning techniques, placing agents in an environment where they actively learn optimal actions based on feedback.

At the onset, an agent's actions in an RL framework are often exploratory or random. As the agent interacts with its environment, it receives feedback in the form of rewards or penalties based on the quality of its decisions. This feedback loop, iterated over time, refines the agent's strategy. Ultimately, the agent seeks to develop a policy – a model predicting the most favorable action in a given context. The policy's goal is not just to react to immediate rewards, but to maximize cumulative rewards over time, ensuring long-term optimal strategies [88], [89] .

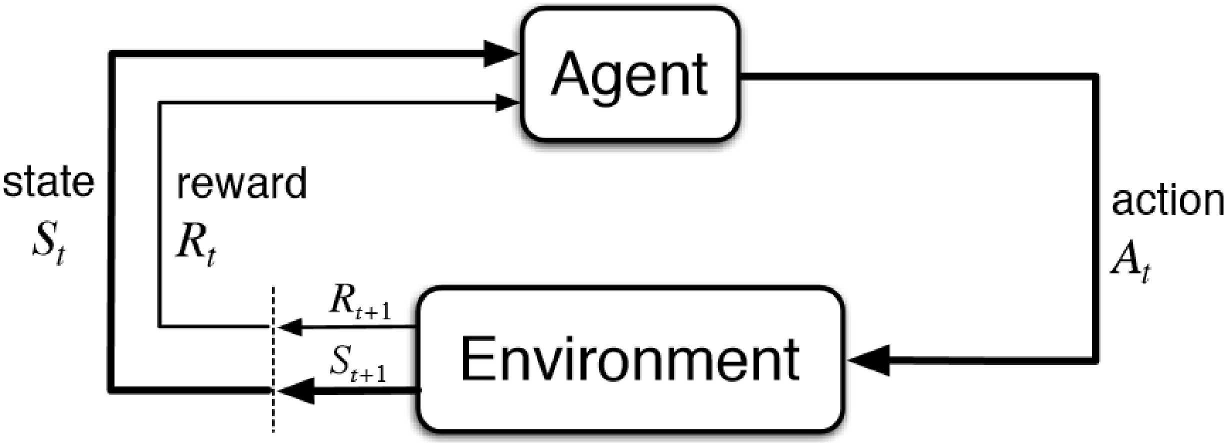


Figure 4 - A visual representation of Reinforcement learning, where t is the timestep, S_t is the state of the environment, R_t is the reward obtained for this state-action pair, A_t is the action of the RL agent, R_{t+1} and S_{t+1} are the next state and next reward which occurred from A_t action [88].

When integrated into federated settings, RL presents a suite of challenges and opportunities. RL in federated contexts, provies opportunities such as model searching, which aims to uncover the best neural architecture across a consortium of data sets while minimizing aggregate loss. Traditional FL models might overlook intricate model architectures, emphasizing predominantly centralized training.

For instance, in their paper, Shuai Yan and colleagues address the challenges arising from the rapid development of the Internet of Things (IoT) and edge computing technologies, specifically concerning privacy and security in heterogeneous device environments. Recognizing the potential of FL as a solution for privacy concerns in IoT edge computing, the authors introduce a novel node selection strategy anchored in deep reinforcement learning to enhance FL's effectiveness in these diverse environments. Furthermore, they devise a metric model to gauge the performance of various IoT devices. Their experimental findings indicate that their proposed method can elevate training accuracy by 30% within a heterogeneous IoT device setting [90].

Federated Reinforcement Learning (FRL) emerges as a synthesis of reinforcement learning (RL) and FL. Within the context of the Internet of Things (IoT), FRL harnesses the distributive nature of FL, allowing multiple learning agents to train a collective model without sharing their local datasets, thus ensuring privacy. This paradigm of learning is particularly crucial for the edge computing scenarios in IoT, where devices at the edge collect data and make decisions [91].

One specific model, termed FedMC, integrates reinforcement learning models from various edge devices into a cohesive model using a meta-learning approach. In this framework, each participating device uses a meta-value network (MVN) and task-actor encoder network (TAEN) to conduct meta-learning training on local task samples. The device then periodically uploads the weights of its local MVN and TAEN to a central server, which amalgamates them into a global model that boasts quick adaptability and applicability across different tasks. The overarching aim of FRL in IoT is to address intricate problems, spanning areas like security, efficiency, vehicular solutions, and industrial services, while preserving the privacy of individual datasets [92].

However, federated RL is not without challenges. As it will be further discussed later, data sets across different participants often deviate from being independent and identically distributed (i.i.d.), leading to possible overfitting or misalignment between participant models. Moreover, concurrently optimizing hyperparameters and model weights in such a distributed setting demands significant resources..

3.2 Types of FL

FL, as a frontier in the machine learning landscape, addresses challenges tied to data decentralization, privacy, and efficiency. Distinct scenarios demand specialized FL strategies, and hence we identify distinct categories. FL in literature is usually categorized in three primary categories, which we expand upon here, together with a novel one: Federated Reinforcement Learning (FRL). Each of these types of FL utilizes a variety of methods and techniques for model training, drawing upon the shared information sources available. The comprehension of these categories allows researchers to examine the unique characteristics and applications of each type, which in turn aids in the advancement of FL strategies [93].

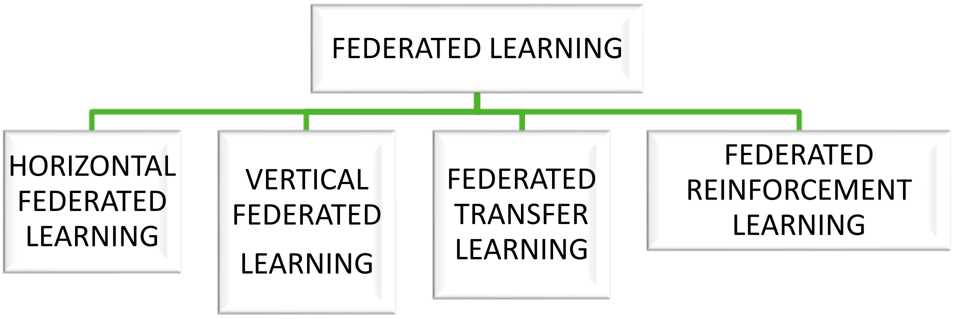

FEDERATED LEARNING
HORIZONTAL FEDERATED LEARNING
VERTICAL FEDERATED LEARNING
FEDERATED TRANSFER LEARNING
FEDERATED REINFORCEMENT LEARNING

Figure 5 – Types of FL

3.2.1 Horizontal FL (HFL)

Machine learning has traditionally relied upon the centralized training paradigm, wherein data from various sources are amalgamated into a single repository. This conventional methodology streamlines preprocessing, grants ease in data access, and enables a unified model evaluation procedure. The primary advantage of this approach lies in its efficacy—attributable to the homogeneity and volume of the data—which, in turn, fosters robust model performance and iterative improvements.

However, the increasing challenges of data distribution, coupled with mounting concerns over privacy and security, have paved the way for innovative methodologies that deviate from this centralized paradigm. Enter Horizontal FL (HFL) — a construct that serves as a beacon in such a landscape. Colloquially termed "sample-based" FL, HFL offers an cutting-edge perspective in the domain, uniquely suited to cater to decentralized data. It is particularly salient in scenarios where various entities—whether businesses or institutions—accumulate data that aligns in feature metrics but originates from a diverse set of user samples [94].

The foundational tenet of HFL is its consistent feature space maintained across all participating nodes or devices, while individual sample spaces or data records differ. This framework ensures that data integrity and privacy are sacrosanct. In essence, while the data features are consistent among participants, direct data sharing is obviated, allowing entities to collaboratively refine a shared model [93].

For a tangible insight into HFL's applicability, consider two geographically separated hospitals. Both institutions might capture analogous parameters for patients—age, weight, and blood pressure to name a few. The divergence arises from the distinct patient demographics each caters to. In this scenario, HFL acts as a nexus, enabling these hospitals to synergistically enhance their diagnostic models. A notable merit of this model is the implicit assurance that patient-specific information remains ensconced within its originating institution, underpinning the principle of data privacy, fairness and accuracy [95].

In this type, data samples from various devices are split horizontally, meaning that each device has the same labels for a subset of features. The objective is to prepare a model safeguarding the protection of information in a cooperative way. The usual way of handling AI, which includes concentrating information on a server, presents reasonable difficulties, for example, high mailing costs, exorbitant battery usage, and risks to the protection and security of customer information [93].

FL, presented by McMahan et al., it has gained critical consideration for its ability to build powerful models in a decentralized way without direct access to customer information, thus ensuring security [1]. Unlike conventional appropriate AI, blended learning tends to address the difficulties presented by non-IID (non-autonomous and indistinguishable circulation) data and imbalanced information experienced in genuine applications, for example, mobile phone applications and mode ID trip using non-IID GPS addresses. With the increasing complexity of information gathering and division among associations, especially when managing sensitive information, disconnected information repositories maintained by individual information owners have become prevalent. This requires the advancement of AI models that can be prepared without unifying all the information.

Yet, despite its promise, HFL is not without challenges. A foremost concern is the potential data imbalance among participants. A disparity in the volume or diversity of datasets between entities could precipitate a bias in model performance. This necessitates the implementation of strategies that ensure fair representation and learning, maintaining the integrity of the HFL methodology.

In summation, while traditional machine learning frameworks might revel in the advantages of data centralization, HFL stands testament to the potential of decentralized, collaborative models in an era marked by data privacy imperatives. This juxtaposition not only underscores the adaptability of machine learning but also heralds a future of evolving methodologies receptive to contemporary challenges.

3.2.2 Vertical FL (VFL)

Vertical FL (VFL) is a specialized variant of FL. In VFL, local parts often have diverse attributes or features of the same user cohort, instead of numerous records with identical feature space found in general FL systems. Within the VFL framework, two primary types of participants exist: one set (the active party) which initializes the training task and possesses the main label for the data samples, and the other (the passive participants) that contribute additional features to the same user set. An apt real-world example is in the financial sector. For instance, a bank might have limited transactional history features but can train a model to predict default risks and customer credit scores by leveraging VFL with another entity possessing complementary data [96], [97].

However, what differentiates VFL from its horizontal counterpart, Horizontal FL (HFL), is the nature of data it deals with. While HFL often concerns entities with data from different users but the same kind of information (often termed as 'data from A parts'), VFL grapples with situations where entities possess complementary datasets (or 'data from B parts'). Parties in a VFL setup target the same user set, but the features in their datasets differ significantly. Feng et al. (2018) further elaborated on this dichotomy, highlighting that one entity might have demographic data about users, while another might focus on their online behavior. The common identifier, such as user IDs, facilitates collaboration without compromising on data privacy. Throughout this entire process, feature values remain undisclosed, thereby ensuring data confidentiality [98].

Another significant distinction in VFL, in contrast to Horizontal FL (HFL), revolves around the nature of data. VFL emphasizes scenarios where participating entities have different features of the same user set, contrary to HFL where data is typically spread across numerous users. This cooperative approach in VFL aids in assembling AI models that use each participant's contribution, streamlining the model training process in settings where traditional averaging methods might fall short [99].

An Illustrative Real-world Application of VFL would be the following: Consider a scenario where a financial institution, abundant with transactional data, partners with an e-commerce platform, which meticulously tracks users' browsing and purchasing behaviors. Historically, insights from either of these entities would remain in isolated silos. However, VFL

introduces the potential for a transformative symbiotic relationship. Leveraging common user identifiers, these entities can collaboratively design predictive models, giving rise to insights such as predicting purchasing propensities based on an amalgamation of financial history and online behavior [96], [97] .
To further illustrate, consider a multi-party, multi-class VFL (MMVFL) system as proposed by Feng et al. 2018. This framework takes into account the distribution of labels across VFL participants while ensuring data privacy. It empowers multiple entities to collaborate and share labels, thereby enhancing the overall learning experience. A more streamlined approach, as suggested by Yang et al., reduces the complexity of the VFL system architecture and coordination requirements by eliminating the need for a central coordinator. Such advancements elevate the efficiency and adaptability of VFL frameworks [96].

VFL, despite its innovative nature, comes with its own set of intricacies. Challenges include the intricate alignment of data across involved parties, maintaining rigorous privacy standards, and seamlessly integrating models that are trained on diverse datasets. As Hu et al. emphasized, the objective is beyond mere data combination; it's about synthesizing the data for insightful and actionable outcomes.

To address the challenges arising from diverse and unreliable network connections among participants, asynchronous VFL architectures, have gained traction. These systems empower each party to update models asynchronously, eliminating the necessity to synchronize data sharing. Contrary to previous VFL systems that predominantly depended on cryptographic methods like homomorphic encryption and secure multi-party computation for secure and confidential learning, these modern strategies provide alternative techniques, aiming to strike a balance between protection, competition, and efficiency in VFL [94], [100].

Figure 6 offers a visual distinction between HFL and VFL frameworks. Notably, the VFL architecture often mandates an external coordinator, pivotal during the inference phase. This role, as illustrated in Figure 7, acts as a trusted intermediary, combining intermediate results from each entity to deduce the aggregated conclusions.

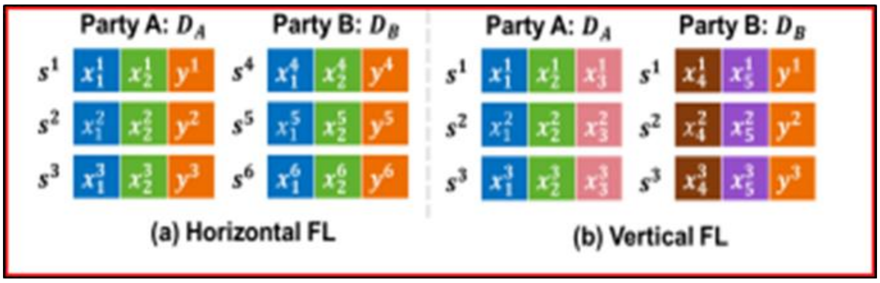

Party A: D_A
Party B: D_B
s^1 x_1^1 x_2^1 y^1
s^2 x_1^2 x_2^2 y^2
s^3 x_1^3 x_2^3 y^3
s^4 x_1^4 x_2^4 y^4
s^5 x_1^5 x_2^5 y^5
s^6 x_1^6 x_2^6 y^6
(a) Horizontal FL
Party A: D_A
Party B: D_B
s^1 x_1^1 x_2^1 x_3^1
s^2 x_1^2 x_2^2 x_3^2
s^3 x_1^3 x_2^3 x_3^3
s^1 x_4^1 x_5^1 y^1
s^2 x_4^2 x_5^2 y^2
s^3 x_4^3 x_5^3 y^3
(b) Vertical FL

Figure 6 - Comparison of HFL and VFL frameworks, where s i represents the i th record, x i j and y i represent the j th feature and the label of the i th record, respectively [98]

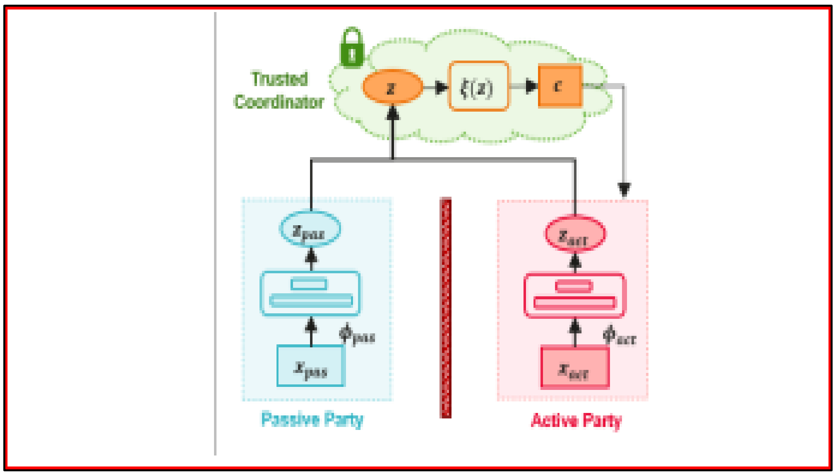


Figure 7 - Inference process of VFL systems with a coordinator [98]

While the centralized coordinator is pivotal in traditional VFL frameworks, acting as a protective buffer to fend off potential data breaches, the trend is shifting. The central role of an external coordinator, which collects intermediate results secretly and calculates aggregated conclusions, provides an additional layer of security. This mechanism acts as a deterrent, preventing potential attackers from directly accessing intermediate results. However, as technological landscapes evolve, newer methods advocate for the removal of this coordinator, directly linking the active and passive parties. It's imperative, given these advancements, to continuously weigh the benefits of efficiency against potential security vulnerabilities.

Recent breakthroughs, however, have postulated the removal of the coordinator role in VFL frameworks. Instead, as proposed in the hostless framework model, intermediate results from one entity (often the passive participant) are directly communicated to the active participant. While this configuration promises efficiency gains, it demands meticulous security scrutiny, to fortify against potential breaches and to ensure robustness against adversarial attacks.

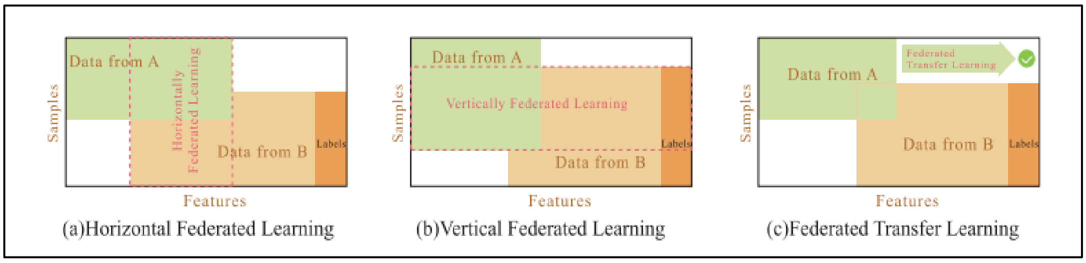
To conclude, VFL stands as a testament to the innovative endeavors responding to the demands of the contemporary data-driven epoch. Through facilitating collaboration between entities, it ensures data remains localized, yet the shared insights extracted are holistic. As its landscape evolves, continuous research is paramount, focusing especially on the intertwined challenges of efficiency, security, and privacy.

3.2.3 Federated Transfer Learning

In the realm of machine learning, the capacity to harness knowledge from one domain and apply it to another holds transformative potential. Federated Transfer Learning (FTL) emerges at this academic intersection, proposing a paradigm wherein models, once trained in a particular setting, can be deftly recalibrated to cater to a different, albeit related, task. This recalibration is performed with a keen sensitivity to the nuances in data distributions, presenting a marked advancement from traditional machine learning models that operate within more confined boundaries of fixed datasets [101], [102].

FTL astutely amalgamates the principles of transfer learning and FL, producing a hybrid methodology that is greater than the sum of its parts. Transfer learning, in its essence, facilitates the application of knowledge acquired in one domain to a separate but related domain. When FL is brought into the mix, the model can be fine-tuned across distributed datasets without direct data exchange, ensuring privacy and reducing computational overheads. This is a noteworthy departure from conventional models, which require centralization of data or operate within a limited scope of pre-defined tasks.

Figure 8 illustrates the distinct data distribution patterns in HFL, VFL, and FTL and Figure 5 providing clarity on their interplay and differences.



(a)Horizontal Federated Learning (b)Vertical Federated Learning (c)Federated Transfer Learning

Figure 8 - Different data partition of horizontal FL, vertical FL, and federated transfer learning [103]

To elucidate the potential of FTL, consider the domain of medical diagnostics, an area where data sensitivity and specificity are paramount. Let's postulate a pre-existing machine learning model adept at diagnosing skin diseases based on a European demographic dataset. Historically, to adapt this model for an Asian demographic, a complete retraining process would ensue, often necessitating the transfer of voluminous, sensitive data across geographies. However, with FTL, the model can be efficiently recalibrated using data from Asian hospitals, without the actual datasets ever leaving their respective institutions. This not only upholds data privacy but also capitalizes on the foundational knowledge embedded within the initial model like shown in [102] and [104] .

As with any advanced methodology, FTL is not exempt from inherent challenges. A salient concern in this domain is the concept of negative transfer. In traditional machine learning, the hazards of overfitting or improper training are well-documented. In the context of FTL, negative transfer can be perceived as an analogous predicament where the knowledge imported from the source domain counterproductively affects the model's performance in the target domain. Ensuring that the transferred knowledge is both relevant and constructive becomes crucial, demanding rigorous validation mechanisms and iterative refinements.

Traditional machine learning models, grounded in their specific datasets, often exhibit a lack of fluidity when addressing tasks beyond their training purview. Their performance, when confronted with subtle shifts in data distribution or task objectives, might be suboptimal. FTL, in contrast, offers a dynamic adaptability, ensuring models are not merely data-responsive but also data-proactive. By drawing upon previous learnings and adapting them to new contexts without the need for raw data exchange, FTL showcases an evolutionary stride in machine learning, rendering it more resilient and versatile.

FTL, while embodying a synthesis of transfer and FL, carves its niche in the ever-expanding tapestry of machine learning methodologies. Its capacity to bridge knowledge gaps across varying data distributions, all while upholding data privacy, underscores its promise. As research in this domain intensifies, refining its mechanisms and surmounting its challenges, FTL stands poised to redefine the boundaries of adaptive, distributed learning.

3.2.4 Federated Reinforcement Learning

Conventional RL paradigms are predicated upon the symbiotic relationship between an agent and its environment. Such agents continually refine decision-making paradigms through an iterative process informed by a sequence of actions and corresponding feedback. This cyclical interplay ensures a progressive honing of strategies, localized to the agent's immediate environment. Conversely, FRL introduces a more intricate schema, wherein multiple agents, each embedded within distinct environments, not only optimize based on local experiences but also integrate insights acquired from a broader network of peers. This multi-agent, decentralized approach infers that knowledge dissemination occurs across agents, transcending the confines of local environmental feedback.

To elucidate with greater specificity: envision a global matrix of autonomous vehicular entities. Under traditional RL frameworks, each vehicle, delineated as an agent, would adapt solely based on its localized traffic dynamics. FRL, however, postulates an enhanced paradigm wherein an agent (for instance, a vehicle navigating Parisian boulevards) assimilates shared knowledge from counterparts in disparate locales such as Tokyo or New York. Such a distributed learning framework posits that firsthand exposure to diverse scenarios is not a prerequisite; rather, vehicles can collate and operationalize shared experiential insights, thereby ensuring a more holistic navigational proficiency.

However, it is essential to acknowledge the nascent nature of FRL and the inherent complexities that arise therefrom. The methodology, though promising, is still crystallizing within the academic community. Pivotal challenges include the architecting of sophisticated communication protocols among heterogeneous agents and ensuring the fidelity of shared experiences in the face of diverse environmental stimuli. Addressing these challenges to attain a cohesive learning trajectory remains an active area of research.

In conclusion, while FRL represents a promising intersection of RL and FL, it remains embryonic in its academic exploration.

Chapter 4: Characteristics

Diving deep into the intricacies of FL, this section illuminates the unique attributes that define and distinguish it within the machine learning domain. Central to our exploration is the multifaceted realm of data privacy. Detailed examinations into differential privacy, secure multi-party computation, and homomorphic encryption highlight the importance and challenges of safeguarding data in a federated environment. Moving beyond privacy, we delve into the essence of personalization in FL, discussing how localized training and aggregated global models work in tandem. A significant portion is dedicated to the phenomenon of Non-IID Data, revealing its profound implications in this learning paradigm. Towards the end, an assessment of the overarching design space is presented, covering a spectrum from data distribution to pivotal considerations such as privacy, security, and robustness. Through this section, readers will gain a comprehensive understanding of the multifaceted aspects inherent to FL, setting the stage for subsequent discussions.

4.1 Data Privacy

FL places a heightened emphasis on data privacy. As highlighted by McMahan et al. (2017), with the mounting prevalence of sensitive data and the tightening of data privacy regulations, there's an undeniable push towards innovative solutions that respect data privacy. FL aptly responds to this call, facilitating the training of machine learning models without centralizing or revealing raw data [1].

4.1.1 Introduction to Data Privacy in FL

The linchpin binding FL's vast possibilities is its unwavering dedication to data privacy. Integrating pioneering techniques like SMC (Secure Multi-Party Computation) and Differential Privacy, FL endeavors to keep user data sacrosanct [44]. Yet, as the nexus of the digital realm expands, the challenges augment in tandem. Ponder upon the nascent dimensions of quantum computing or the intriguing domain of deepfakes – areas that beckon FL to fortify its privacy bastions further.

As we segue into communication, an indispensable facet of FL, especially when we fathom its implementation across a vast ensemble of devices, the insights of Almanifi et al. are actually enlighting. The authors accentuate the nuances of streamlined data conveyance and underline the significance of resource efficiency [51]. Compression techniques can mitigate data transmission volumes, thus fostering FL's feasibility even under bandwidth-limited scenarios .

Navigating the broader applicability spectrum of FL, one encounters the revelations of Bonawitz et al. [105], which underscore the immense potential awaiting industries contending with decentralized data frameworks. The narrative extends beyond training isolated models; envision the realms of ensemble models, intricate collaborative filtering mechanisms, and even more intricate architectures, all cultivated under the FL umbrella.

Central to the ethos of FL is its decentralized methodology. Unlike traditional models that often transfer data to a central repository, FL operates differently. Models are trained right at the source of the data, often referred to as 'edge devices' or 'nodes.' This approach ensures that only model updates or crucial, non-sensitive information are relayed back to the central model, drastically reducing risks associated with data exposure and breaches.

4.1.2 Motivations for Enhanced Data Privacy

The empowerment of original data is another significant aspect of FL [30], [40], [61]. Data remains in its place of origin, be it where it's generated or where it's stored. This decentralized model ensures data owners and creators maintain authoritative control over their sensitive information, empowering them and reducing risks. This very nature of FL, where data doesn't have to be centralized, not only minimizes potential data breaches but also diminishes legal and ethical liabilities tied to unauthorized data access.

Several mechanisms underline the privacy aspect of FL. Bonawitz et al. have thrown light on 'secure aggregation,' a technique that uses cryptographic protocols to amalgamate model updates from different nodes [106]. Such an aggregation ensures that the individual data contributions, despite being part of the broader model, remain shrouded in privacy during the entire process.

One technique that merits a deeper dive is differential privacy. This mechanism ensures that any output from a database, like query results, remains statistically indistinguishable whether a particular individual's information is included or not. The introduction of this randomness ensures that the model, even if trained thoroughly, cannot compromise or reveal specifics about individual data points. In essence, differential privacy provides a strong mathematical guarantee of privacy, allowing data to be used beneficially without jeopardizing individual data privacy.

Homomorphic encryption and Secure Multi-Party Computation (SMC) also stand as testaments to the emphasis on privacy in FL [107]. With sectors like healthcare and finance standing to benefit immensely, the motivations for enhanced data privacy in FL are not just technologically driven but also ethically and legally compelled.

FL places a heightened emphasis on data privacy. As highlighted by McMahan et al. (2017), with the mounting prevalence of sensitive data and the tightening of data privacy regulations, there's an undeniable push towards innovative solutions that respect data privacy. FL aptly responds to this call, facilitating the training of machine learning models without centralizing or revealing raw data.

4.1.3 Differential Privacy in FL

Differential privacy, as an emerging paradigm, is inextricably linked to the core principles of FL. FL, with its decentralized methodology, emphasizes training models directly at data sources, such as 'edge devices' or 'nodes', as opposed to the conventional transfer of data to a central repository. This intrinsic characteristic significantly reduces data exposure and breach risks, setting the stage for enhanced privacy methods like differential privacy. Differential privacy introduces a meticulously calibrated measure of randomness into the data or the model training process. The essence of this technique, as aptly pointed out in the description, is to ensure any output from a database remains statistically indistinguishable, irrespective of whether specific individual information is incorporated or not. The primary intent is to ensure that the model, even after rigorous training, remains unable to disclose or compromise specifics about individual data entries. [50], [56], [108]

Several mechanisms are in place to accentuate FL's commitment to privacy. For instance, Bonawitz et al. (2016) highlighted 'secure aggregation,' a method employing cryptographic protocols to combine model updates from diverse nodes [106]. Such a process, although involving individual data contributions, ensures these contributions remain confidential throughout. This amalgamation method is just one of many that leverage the potential of differential privacy. By introducing the said randomness, it's assured that outputs from a database—such as query results—are designed not to compromise individual data privacy. This mathematical robustness of differential privacy means data can be harnessed productively without endangering individual data privacy.

The advantages of integrating differential privacy in FL are manifold. Given FL's decentralized nature, where data remains at its origin, it already brings down potential data breaches and minimizes unauthorized data access liabilities. When combined with differential privacy, it offers a robust mathematical assurance of privacy, enabling beneficial data usage without compromising individual data confidentiality.

However, this commitment to data privacy doesn't come without its set of challenges. The introduction of randomness, while safeguarding individual data, can sometimes affect model accuracy. Striking a balance between ensuring data privacy and maintaining model accuracy becomes an intricate task. As the landscape of FL continues to evolve, researchers are continuously challenged to refine and perfect techniques that bolster data privacy while ensuring the efficiency and accuracy of models. This remains one of the pivotal considerations as FL seeks to conform to the rigorous data privacy standards set by contemporary regulations [12], [34], [57].

4.1.4 Secure Multi-Party Computation (SMPC or SMC)

An Overview of SMPC and its Importance in FL:

Secure Multi-Party Computation (SMPC or SMC) stands out as a pivotal concept in the realm of FL. With FL's decentralized approach, as depicted by its training of models directly at data sources (such as 'edge devices' or 'nodes'), there arises a need for advanced techniques that can effectively ensure data privacy without compromising on model efficiency. SMPC directly addresses this need. In essence, SMPC allows multiple participants, each with their private inputs, to collaboratively conduct computations and obtain a result without revealing their individual inputs to each other. In the context of FL, where data remains decentralized SMPC's importance cannot be understated. It allows for the collaborative training of models across multiple nodes without the necessity to expose individual datasets [109], [110].

How SMPC Works in a Federated Context:

In FL, models are trained at the data's source, ensuring minimal risk associated with data breaches and exposure. Within this framework, SMPC operates by facilitating collaborative computations across these decentralized nodes. Rather than sharing raw data, which could compromise privacy, each participant or node shares encrypted fragments of computations. These fragments, when combined, can provide a result (like a model update) without ever revealing individual data points.

While secure aggregation focuses on the amalgamation of updates, SMPC takes it a step further by allowing entire computations to be jointly performed, yet without the nodes having to reveal their proprietary datasets to one another.

Practical Applications and Limitations:

Given FL's application in data-sensitive domains like healthcare and finance, the utility of SMPC becomes even more pertinent. These sectors can leverage SMPC within the FL framework to derive insights collaboratively from multiple sources, all while ensuring that individual data remains confidential [111].

However, SMPC, despite its strengths, isn't without limitations. The complexity of collaborative computations across multiple nodes can introduce latency, especially as the number of participating nodes increases. Moreover, the cryptographic techniques underlying SMPC, while ensuring privacy, can sometimes be computationally intensive, demanding more resources and potentially affecting the efficiency of the FL process.

In conclusion, as FL's future continues to shine with ongoing research and innovations, techniques like SMPC are central to addressing the perennial challenge: balancing model accuracy and efficiency with the rigorous data privacy standards set by contemporary regulations.

4.1.5 Homomorphic Encryption in FL

An Introduction to the Concept and its Application in FL:

Homomorphic Encryption (HE) stands as a pioneering solution in the ever-evolving landscape of data security, and its intersection with FL marks a significant stride in preserving data privacy. At its core, HE is a form of encryption that permits computations to be performed directly on encrypted data without necessitating its decryption. In essence, it allows for the derivation of meaningful outcomes from computations on encrypted data, which, when decrypted, align with the results one would obtain from the original, unencrypted data (60,61).

Within the FL paradigm, where models are trained directly at the source of data (such as on 'edge devices' or 'nodes'), and the ethos leans heavily on decentralized methodology, HE emerges as a valuable tool. This ensures data owners retain control over their sensitive information. Herein, HE offers a dual advantage: allowing computations, including training of models, on data without exposing the raw data, and subsequently ensuring that any information relayed back to the central model remains encrypted and secure. This approach was further illuminated by Ma et al. (2020), highlighting the rising prominence of HE in the domain of FL [112].

Benefits of Using Homomorphic Encryption:

- **Enhanced Data Privacy:** HE ensures that raw data remains encrypted throughout the computation process. This means that data, even when being used for training or analysis, is never exposed in its raw form, thus significantly mitigating risks associated with data breaches or unauthorized access.

- **Flexibility in Computations:** Despite data being encrypted, HE allows for meaningful computations, ensuring that the insights or model updates derived from such computations are accurate and reflective of the original data.

- **Compatibility with FL:** Given the decentralized nature of FL, where data does not need to be centralized, HE fits seamlessly, ensuring that data privacy is maintained without hampering the training process.

Trade-offs, such as Computational Costs:

While HE offers a robust solution to many challenges posed by data privacy, it comes with its set of trade-offs:

- **Computational Intensity:** The very processes that allow computations on encrypted data can be computationally intensive, necessitating more robust computational resources and potentially elongating the training or analysis time.

- **Increased Latency:** Especially in real-time applications or scenarios where rapid response is crucial, the additional time required for HE computations can introduce delays.

- **Complexity:** Implementing and managing HE can be complex, requiring specialized knowledge and potentially complicating the deployment of FL solutions in certain environments.

In summation, while the future of FL continues to be bright, the integration of techniques like Homomorphic Encryption serves to bolster its promise. However, as with all technological advancements, it's crucial to weigh the benefits against the trade-offs to harness its full potential effectively.

4.1.6 Federated Identity and Access Management (FIAM)

Ensuring only authorized devices and nodes participate:

Federated Identity and Access Management stands as a beacon for cybersecurity within FL architectures. In an environment where vast quantities of sensitive data are processed across multiple devices or 'nodes,' the significance of ensuring only authorized participation cannot be overstated. FIAM builds upon traditional Identity and Access Management systems, adapting them to the challenges and intricacies of federated environments [114], [115].

Central to this is the concept of authentication. In FL, where models are trained right at the edge devices, the risk of data tampering or introduction of malicious data is heightened. By having a robust FIAM in place, the federated system can validate the credentials of each participating node, ensuring that they are recognized, trustworthy, and have the right to contribute.

Role-based access in federated networks:

Role-Based Access Control (RBAC) in FIAM goes beyond just determining which nodes can participate. It meticulously defines what each node, based on its assigned role, can and cannot do within the FL process. Roles are typically defined based on the function a node performs, its level of trust, or the nature of data it possesses [116], [117], [118].

For example, a node in a healthcare federated network with access to highly sensitive patient data might have a different role compared to a node with non-sensitive administrative data. The former might be restricted from sharing raw data but allowed to share aggregated insights. Implementing RBAC effectively can thus be a game-changer, offering fine-grained control over data sharing and processing activities within the federated network, safeguarding against potential breaches and misuse.

4.1.7 Privacy-Preserving Data Aggregation

In the contemporary landscape of data analytics and machine learning, there exists an imperative need to harness data from diverse nodes while maintaining stringent confidentiality measures for individual data points. The challenges posed by this requirement have necessitated the evolution of robust techniques to aggregate data without compromising its raw integrity.

As discussed earlier, one notable method employed to achieve this is Differential Privacy. Conceptually, differential privacy aims to perturb the true data values, introducing calibrated "noise" into the aggregation process. This noise infusion ensures that any statistical queries on the aggregated dataset do not inadvertently leak information about an individual data point's presence or absence. Its application, especially in FL paradigms, enhances the assurance that individual contributions remain indistinguishable within the aggregated dataset.

Another sophisticated methodology in this realm is k-Anonymity. This technique operates on the premise of data indistinguishability, wherein individual data records are rendered indiscernible from at least k-1 other records in the dataset. Essentially, even if an adversary possesses external knowledge of an individual's presence within the dataset, the exact data entry corresponding to that individual remains obfuscated, providing an additional layer of data privacy [118], [119], [120].

Further bolstering this arsenal is Data Masking. This approach strategically alters data values, producing a derivative dataset that retains the same structural attributes but possesses obfuscated true values. By transforming genuine data entries into structurally congruent, yet modified versions, data masking precludes the direct transmission of raw data, mitigating the risk of inadvertent or malicious exposure (67,69,70).

Transitioning from these data protection methodologies, it is crucial to underscore the importance of "Secure aggregation protocols." These protocols are not mere adjuncts; they form the bedrock of secure data transmission. A cornerstone of these protocols is the emphasis on Data Privacy. Leveraging advanced cryptographic methodologies, data is encrypted in a manner that permits the central server to decipher only the cumulative or aggregated insights, rendering individual data contributions unintelligible.

Complementing data privacy is the imperative for Integrity Checks. Beyond mere confidentiality, the authenticity and veracity of aggregated data are paramount. Secure aggregation protocols intrinsically incorporate mechanisms to detect anomalies or potential data adulteration, safeguarding the integrity of the collective data pool [27], [123].

In summation, as academic and industrial pursuits veer increasingly towards decentralized data analytics, the onus of safeguarding individual data integrity while leveraging its collective prowess has never been more pronounced. The aforementioned techniques and protocols epitomize the vanguard of our endeavors in this direction.

4.1.8 Privacy Challenges and Limitations in FL

One significant vulnerability that has garnered attention is the risk of model inversion attacks. Here, crafty adversaries, by merely analyzing the outputs of a trained model, attempt to reverse-engineer or "invert" specific data points that were likely utilized during the training phase. This modus operandi hinges on exploiting the model's outputs to deduce granular details about its training data, which could inadvertently lay bare sensitive user information. It's noteworthy that even though FL shields the raw data from exposure, the divulged model parameters—or their consequential outputs—might still be susceptible to these inversion strategies A comprehensive exploration of such attack paradigms and the mechanisms to thwart them will be detailed in Chapter 6.

Simultaneously, FL finds itself at the crossroads of conflicting needs. The sanctity of user privacy stands at one end of the spectrum, and at the other, the pressing need to derive models of impeccable accuracy. This dynamic presents multifaceted challenges. For instance, the very techniques crafted to bolster privacy, such as differential privacy [50], [107], introduce noise into the data. While this noise acts as a veil, safeguarding individual data points, it concurrently risks obfuscating genuine data patterns, potentially diminishing the model's predictive accuracy.

The conundrum doesn't end there. It begs the question: is there any merit in a supremely private model if its accuracy is too compromised for tangible real-world applications? And in the same breath, does a highly accurate model hold any value if it tramples over established privacy protocols? These questions become particularly poignant in critical sectors like healthcare. Here, the stakes are elevated, as the data is not only intensely personal but also pivotal for patient outcomes. In such contexts, the equilibrium between unerring predictions and unwavering privacy is not just desirable but imperative [33], [111].

Additionally, the very methods designed to ensure privacy can sometimes be a double-edged sword. Some of these, despite their efficacy, introduce computational and communication overheads. This can either decelerate the model training or escalate resource requisites. Such implications might pose hurdles, especially in environments where computational resources are at a premium [57].

4.1.9 Best Practices and Standards

To ensure privacy in FL deployments, it is imperative to adopt several guidelines. Firstly, FL primarily dictates that data should remain on the user's device, be it edge devices or nodes. As such, raw data should never be transmitted, curtailing potential data exposure risks. Secondly, it's crucial to adopt a data minimization approach. By ensuring that only the essential data required for model training is processed, the volume of data vulnerable to potential breaches is reduced, thereby minimizing the potential fallout. Regular model audits constitute the third guideline. By frequently validating and evaluating FL models, any potential privacy vulnerabilities can be promptly identified and mitigated. Fourthly, the application of differential privacy techniques can be invaluable. These techniques, which introduce calibrated noise into the data, mask individual contributions during the aggregation phase, bolstering privacy measures. Lastly, always make sure that data is encrypted during transmission. Techniques such as homomorphic encryption, which facilitates computation on encrypted data without the need for decryption, can be particularly valuable in this context [15].

Furthermore, the rising emphasis on data privacy in FL is leading to the emergence of standards and protocols. Notably, the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) have rolled out standards that specifically address the privacy nuances of cloud computing and distributed systems, making them particularly relevant to FL. Simultaneously, the National Institute of Standards and Technology (NIST) is offering robust guidelines and best practices, emphasizing the secure transmission and storage of data. In addition to these established institutions, community-driven initiatives are gaining traction. Open-source communities, in conjunction with academic consortiums, are proactively collaborating to delineate standards and protocols tailor-made for FL, reflecting its ascending significance in the realm of data privacy [124], [125], [126].

Bellow we summarize the best practices and standards in the two respective lists, for brevity and ease of use:

Guidelines to ensure privacy in FL deployments:

1. **Local Data Storage:** At its core, FL ensures that data remains on the user's device (like edge devices or nodes). This practice should be maintained, ensuring that raw data is never transmitted, reducing potential data exposure.
2. **Data Minimization:** Only the essential data required for model training should be processed. This reduces the volume of data at risk and lessens potential harm in the event of a breach.
3. **Regular Model Audits:** Frequent validation and evaluation of the FL models help in identifying and addressing any potential privacy vulnerabilities.
4. **Differential Privacy:** One should ensure the implementation of techniques like differential privacy to introduce calibrated noise into data, ensuring that individual contributions are masked during aggregation.
5. **Encryption:** One should always encrypt data during transmission. Techniques like homomorphic encryption, which allows computation on encrypted data without needing decryption, can be pivotal.

Emerging standards and protocols for data privacy in FL:

- **ISO/IEC Standards:** Various standards, particularly from the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC), address the privacy aspects of cloud computing and distributed systems, which are relevant to FL [125], [126].
- **NIST Guidelines:** The National Institute of Standards and Technology (NIST) offers guidelines and practices on securing data during transmission and storage [124].
- **Community-Driven Initiatives:** Open-source communities and academic consortiums are collaborating to define standards and protocols specific to FL, given its growing significance. [15], [127]

4.1.10 Future Directions

In the digital epoch, the notion of privacy is in constant flux, particularly in the context of FL. With an exponential rise in connected devices and an augmented dependence on AI, FL, though decentralized by nature, remains susceptible to potential security pitfalls. Yet, there's a silver lining: global momentum is shifting towards rigorous data protection regulations, coupled with the evolution of advanced privacy techniques tailored for FL. Notably, Zero-Knowledge Proofs are gaining traction. This cryptographic innovation is especially pertinent for FL, allowing for the verification of data's veracity without exposing its details, thus maintaining the essence of decentralized learning [128], [129] .

Moreover, the evolution of cryptography is lending a hand to FL. The next wave promises fortified secure aggregation protocols specifically for FL, which can deftly combine efficiency and security. These ensure the swift consolidation and training of decentralized models without jeopardizing user privacy.

The looming advent of quantum computing introduces a new dimension to FL's privacy landscape. The double-edged sword of quantum technologies presents a paradox. On one edge, once quantum computers achieve a certain prowess, they might shatter the existing cryptographic bulwarks that FL relies on. But in anticipation of this quantum challenge, the tech community is already pioneering quantum-resistant cryptographic solutions tailored for FL, ensuring the privacy of decentralized data even in a quantum era. On the other edge, quantum mechanics, which could be a threat, also offers salvation. Quantum Key Distribution (QKD), for instance, introduces a quantum-attuned encryption method, potentially elevating the security protocols in FL to an unprecedented zenith [130], [131].

To encapsulate, FL, despite being a beacon for decentralized and privacy-preserving machine learning, is still unfolding. As it navigates through the evolving technological challenges, it is imperative for FL enthusiasts and professionals to keep pace with emerging best practices and avant-garde solutions, ensuring that the promise of private, decentralized learning isn't just a dream but a tangible reality.

4.2 Personalization

In today's digital landscape, personalization stands as a beacon of enhanced user experience. With the burgeoning complexity of user preferences, the challenge is not just offering tailored services but doing so while upholding the principles of data privacy. FL, offers an avant-garde solution to this challenge, revolutionizing the traditional paradigms of personalization.

4.2.1 Localized Model Training

The cornerstone of FL's approach to personalization is localized model training. Models are sculpted directly on users' devices, imbibing the unique nuances, characteristics, and behavioral patterns of individual users. This on-device personalization, that services cater to users in ways that are most relevant to them, enhancing overall engagement and satisfaction [22].

Furthermore, the localized nature of the training allows for real-time adaptation. As users' behaviors and preferences shift, the models adapt almost instantaneously. This ensures that the predictive accuracy and relevance of the models are consistently high, leading to an enriched user experience.

Moreover, the efficiency of on-device training reduces the latency that can occur when data is sent back and forth between central servers. This increased efficiency means that updates can be rolled out rapidly, ensuring that users always experience the most current and optimal version of the model [21].

4.2.2 User Data Privacy

In contrast to traditional systems, where raw data is funneled to central repositories, FL stands apart. Personalization is achieved without the need to migrate raw user data to centralized servers. Instead, as outlined by Bonawitz et al. (2019), only aggregated model updates, refined from a multitude of devices, are communicated back to the central model, ensuring that the sanctity of individual data privacy remains inviolate [23].

By eliminating the need to centralize raw data, FL also diminishes the potential attack surface for malicious entities. Hackers and cyber attackers often target centralized databases; however, FL's decentralized nature diffuses this risk.

Additionally, with the growing concerns and stringent regulations surrounding data privacy worldwide, FL presents a forward-thinking solution. Organizations can now offer personalization without falling afoul of these regulations, thereby building trust with their user base.

4.2.3 Improved User Experiences

The benefits of personalization in FL manifest as tangible improvements in user experiences. Whether it's a predictive text application offering more accurate suggestions or a content platform curating a more relevant playlist, the models, tailored through localized learning, provide a seamless and intuitive user experience [132], [133].

These improvements extend beyond just digital services. Think about wearable health devices tailoring fitness suggestions based on a user's specific health metrics or a smart home adjusting itself based on a resident's preferences. The potential for hyper-personalization across a spectrum of applications is vast.

Importantly, improved user experiences often translate to increased brand loyalty and user retention. When users feel that a service "understands" them and caters to their unique needs, they're more likely to continue using that service.

4.2.4 Aggregated Global Model

FL's approach to personalization is twofold. While devices engage in localized learning, there's also an aggregated global model being refined in tandem. Insights from across devices are amalgamated, creating a model that captures broader patterns. This model is then disseminated back to individual devices, ensuring that even localized models can benefit from more expansive insights. This collective intelligence ensures a base layer of efficacy for all users. Even if an individual's device hasn't undergone extensive local training, they can still benefit from the broader knowledge amassed by the global model.

Moreover, the global model serves as a counterbalance. In instances where individual data might be too sparse or erratic, the global model provides stability, ensuring users still receive a coherent and effective personalized experience. Aggregated models are further discussed in Chapter 4.

4.2.5 Data Efficiency

FL showcases remarkable efficiency, especially when dealing with sparse or unique user data. By concentrating on local data, it harnesses the power of even limited datasets, ensuring that even outliers or users with distinct preferences experience top-tier personalization.

This approach reduces the computational overhead typically associated with big data. Instead of sifting through vast and sometimes redundant data sets, FL focuses on what's immediately relevant. Furthermore, such efficiency proves invaluable in scenarios where data transmission is costly or bandwidth is limited. Local processing ensures that only essential updates are communicated, preserving resources.

4.2.6 Collaborative Personalization

Beyond individualized learning, FL also facilitates collaborative personalization. Here, devices or users with overlapping preferences or behaviors might collaboratively refine a shared model. While each user's unique data remains private, collective insights bolster the model's effectiveness [134].

This shared learning can expedite the refinement process. When multiple users encounter a similar challenge or express a common preference, the model can adapt more swiftly. Additionally, this collaborative approach engenders a sense of community. Users indirectly benefit from the collective wisdom of their peers, fostering a more interconnected and enriched user experience [135] FL with Personalization Layers.

4.2.7 Dynamic Adaptation

The dynamic nature of FL ensures that models are not static. They evolve, adapting to the changing moods, preferences, and habits of users. This ensures that personalization remains a constant, even in the face of fluctuating user behaviors.

This adaptability is essential in today's rapidly changing digital landscape. As trends evolve and user preferences shift, models that can't keep pace become quickly outdated. Moreover, such dynamism allows for real-time feedback loops. As users interact with models, their responses can be immediately factored into ongoing refinements, creating a truly responsive personalization framework.

4.2.8 Challenges and Trade-offs

However, as with all innovations, FL's approach to personalization isn't devoid of challenges. Striking the right balance between personalizing a model and ensuring its broader applicability remains a focal point. Over-personalization might lead to models being too niche, risking reduced effectiveness when user behaviors shift.

It's also worth noting the computational challenges. While FL distributes the computation across devices, this requires a certain baseline capability on user devices. Older or less powerful devices might struggle with complex on-device calculations. Furthermore, ensuring that the global and local models integrate seamlessly can sometimes be a balancing act. There's a potential for conflicts or inconsistencies, which necessitates sophisticated model management techniques. [7], [22]

The promise of FL in personalization is profound. From healthcare, where personalized models can catalyze tailored treatment regimens while upholding patient privacy, to e-commerce, where individualized recommendations can enhance user engagement without compromising transactional data – the applications are boundless. As research continues to expand, the confluence of personalization and data privacy through FL is poised to redefine the digital realm.

4.3 Non-IID Data in FL

In the emerging field of FL, data distribution remains central to the effectiveness of the resulting models. Particularly, the challenge posed by non-IID (Non-Independent and Identically Distributed) data, which presents unique complications in a decentralized training environment, demands attention.

4.3.1 Understanding Non-IID Data

In a FL framework, non-IID data is characteristic of situations where participating devices, be they clients or nodes, possess data distributions that aren't a balanced representation of the broader population. Several factors can lead to this imbalance. For instance, different devices might adopt varying data collection procedures, reflecting the diversity and variability of devices. Further complicating matters are user-specific behaviors, which can create unique data environments. Challenges like device-specific characteristics and interactions that could further skew the data [53].

4.3.2 The Implications and Challenges of Non-IID Data

The inherent challenges of non-IID data in FL are multi-faceted:

Model Convergence Issues

The path to model convergence becomes rocky with non-IID data. Wang et al. (2020) provided insights into this phenomenon, noting that an imbalance in data distribution across nodes results in protracted convergence rates, thereby extending the training period and potentially reducing model efficacy [54].

Performance Degradation

A model's real-world effectiveness is compromised when it is dominantly trained on non-representative data. In scenarios where such data is prominent, the model, when exposed to varied real-world scenarios, might underperform or yield unpredictable results.

Biased Outcomes

A skewed data source, if dominant during training, can lead to models that are biased towards particular user groups or scenarios. Such a bias not only reduces the model's broad application but also raises ethical concerns. As Briggs et al. (2020) elaborated, data biases can inadvertently promote certain groups while marginalizing others, thus compromising the fairness and ethicality of the model [136].

4.3.3 Non-IID: Potential Strategies

Addressing non-IID challenges also requires a multifaceted approach:

Data Preprocessing

A foundational step towards countering data imbalances is through preprocessing measures. Implementing techniques like data augmentation, stratified sampling, and synthetic data generation can equalize data distribution across nodes, ensuring a more balanced training environment.

Model Aggregation Approaches

Aggregation techniques bear significance in a non-IID environment. Recent methodologies, such as those echoing the principles of FedAvg and FedProx, show promise. These methods, as explored by Briggs et al. (2020), aim to counteract the challenges posed by skewed data, enhancing both model performance and convergence [136].

4.3.4 Real-world Implications across Sectors

The practicality of non-IID data settings in FL transcends numerous sectors. First he decentralized nature of FL, combined with non-IID data considerations, permits hospitals to collaboratively train models without compromising patient confidentiality.

Also, in the finance sector, FL offers a mechanism to analyze distributed financial datasets securely, ensuring proprietary and confidential information remains within respective entities. Last but not least, IoT, with its vast array of heterogeneous devices, produces diverse datasets. FL becomes indispensable in such settings, allowing for collaborative model training that respects the unique data origins.

4.3.5 Synchronous vs. Asynchronous Updates

Synchronous and asynchronous mechanisms form the core of FL model updates. The synchronous mechanism involves a central server that waits for updates from every node before proceeding with model aggregation. Yet, challenges emerge when certain devices lag – a phenomenon termed as "stale updates." Such staleness, often attributed to device heterogeneity and network reliability, becomes a significant concern. [137]

4.3.6 Privacy and Security in Non-IID Settings

While FL inherently prioritizes data privacy and efficient communication, the specter of security breaches looms large. Concerns, ranging from user privacy violations to threats against model integrity and data attributes, punctuate the discourse. However, the ongoing challenge remains: ensuring unwavering security in scenarios where data isn't uniformly distributed across nodes.

The intricate challenge posed by non-IID data in FL calls for nuanced understanding and tailored solutions. Only by comprehensively grappling with these challenges can FL truly fulfill its promise across diverse domains.

4.4 Design Space

In the emergent domain of FL, the design space plays an instrumental role in determining the viability, efficacy, and security of deployed systems. Comprising an array of choices such as data distribution, model architecture, communication methodologies, and privacy safeguards, this design space is vast. This section provides a detailed dissection of these pivotal design components, undergirded by seminal research findings, illuminating the path for FL endeavors.

4.4.1 Data Distribution and Management

In the realm of FL, where decentralized data sources are the norm, the challenges associated with data management are manifold. One primary concern arises from Data Heterogeneity. Navigating diverse data types, structures, and distributions across distinct nodes is imperative. Such heterogeneity often stems from the varied origins of data sources, demanding bespoke approaches to data harmonization.

Additionally, the Volume of Data held by participating clients stands as a considerable concern. A harmonious FL system ensures a balanced data representation, warding off the risks of over or under-representation of specific client groups. Such balance ensures a generalized model that isn't biased towards any particular data subset.

Lastly, the lifecycle management of data in a federated setting poses unique challenges. Decisions pertaining to updates, deletions, and archiving processes need meticulous planning, ensuring the sanctity and relevance of data throughout the learning phase [138], [139] .

4.4.2 Model Architectures

Model choices significantly steer the training dynamics and eventual outcomes in federated contexts. A fundamental decision point is between Homogeneous and Heterogeneous Models. Such customization potentially optimizes the learning process by catering to the unique characteristics of data subsets [140], [141] .

Furthermore, the Size of the Model directly interacts with communication costs. Larger models, while potentially more accurate, can strain the communication bandwidth. Yet, their capacity to capture intricate patterns might justify the communication overhead, marking the importance of striking a judicious balance. Employing pre-trained models to initiate federated training, particularly in scenarios with limited data, offers a promising avenue for achieving superior results without exhaustive training.

4.4.3 Communication Strategies

FL's iterative nature mandates optimized communication strategies. The debate between Synchronous and Asynchronous Updates hinges on the trade-offs between system latency and efficiency. While synchronous methods ensure coordinated updates, asynchronous strategies might offer flexibility at the potential expense of consistency [137].

Furthermore, given the frequent exchange of model parameters and updates, strategies must account for Bandwidth Constraints. Limiting the amount of transferred data becomes paramount, especially in environments with network limitations [142].

Complementing this is the role of Compression Techniques. Employing mechanisms to condense model updates prior to transmission can yield tangible bandwidth savings, thereby optimizing the communication process [143].

4.4.4 Privacy and Security

In FL, concerns of privacy and security aren't mere afterthoughts but foundational requirements. A cornerstone principle is On-Device Computation. It ensures that confidential data remains localized, transmitting only model updates, thereby obviating the risks associated with data exposure.

Parallelly, Secure Aggregation mechanisms ensure that while model updates from clients are consolidated, individual client data points remain concealed, fostering trust in the federated ecosystem [144].

Ensuring system integrity through Authentication Protocols is equally pivotal. By validating genuine client participation, these protocols shield the FL process from potential adversarial intrusions [145].

4.4.5 Scalability and Robustness

As FL systems potentially integrate an expansive array of clients, emphasizing scalability and robustness is paramount. Managing Client Dropouts, as elucidated by Wang and Xu (2022), becomes an operational necessity. Formulating strategies to accommodate clients who may exit or rejoin the training process is crucial to maintain consistent learning trajectories [146].

Moreover, implementing Adaptive Learning Rates offers an elegant solution to the variability in client contributions. By modulating learning rates in alignment with the quality and volume of client updates, the system can optimize its learning curve.

Lastly, building a resilient federated system that can persevere despite potential node failures or disruptions ensures the uninterrupted continuation of the learning process [147].

Concluding, the design space in FL is intricate, demanding an integrative understanding of domain requisites, technological constraints, and application subtleties. With informed and discerning design choices, FL can be sculpted to suit diverse applications, delivering performance, resilience, and stringent privacy.

Chapter 5: Algorithms & Frameworks

FL has emerged as a groundbreaking advancement in the machine learning landscape. This decentralized approach allows models to be trained across a myriad of devices, ensuring user privacy while delivering personalized outcomes. This chapter provides a detailed exploration into the intricate algorithms and frameworks that anchor FL, setting the stage for subsequent research, enhancements, and practical applications.

However, it is important to note that this chapter, with its in-depth examination of current FL algorithms, merely scratches the surface. The ever-increasing demand for decentralized and privacy-centric machine learning solutions foretells an evolution in the algorithms driving them, promising even more resilient, efficient, and adaptable solutions for the future.

5.1 FL Algorithms

5.1.1 Introduction

In the ever-evolving domain of machine learning, the development and deployment of algorithms form the bedrock of any successful implementation. This sentiment holds even truer for FL, a paradigm that brings with it a set of unique challenges and intricacies. While FL, as a concept, is crucial in preserving data privacy and minimizing data transfers, it is the algorithms that breathe life into this framework, ensuring optimal performance even in decentralized settings.

FL algorithms are a breed apart from their traditional machine learning counterparts. The classical algorithms in machine learning primarily operate under the assumption of direct access to centralized datasets, making computations relatively straightforward. They function under the premise of homogeneity, with consistent data distribution and an ample number of training samples for each client. In stark contrast, FL algorithms grapple with decentralized datasets, often skewed and non-identically distributed among various client devices. This necessitates crafting algorithms that can not only handle such data discrepancies but also optimize communication efficiency, given that slow or unreliable networks can often be the bottleneck.

Vehicles producing vast amounts of data, as a fitting example, underscore the pressing need for FL algorithms. With traditional data transfer methods becoming untenable due to the sheer data magnitude and security concerns, federated algorithms emerge as the beacon, ensuring that while raw data remains localized, insights derived are globally relevant. Such algorithms prioritize transmitting compressed model parameters over raw data, thereby reducing bandwidth needs and enhancing overall efficiency.

As we delve deeper into this chapter, our focal point will be on exploring the nuances of various FL algorithms. The algorithms, ranging from the traditional federated averaging (FedAvg) and Federated Stochastic Gradient (FedSDG), to stochastic variance reduced gradient (FedSVRG) and “CO-OP”, will be meticulously examined. By juxtaposing the latter with traditional algorithms, we aim to highlight their distinctiveness and underscore their significance in the broader machine learning landscape [148].

5.1.2 Federated Averaging (FedAvg)

5.1.2.1 Introduction and Motivation Behind FedAvg

Federated Averaging, commonly known as FedAvg, emerges from the pressing need to converge decentralized machine learning in a harmonized manner [28]. In a world where data is increasingly distributed across devices, transferring all of it to a central location for training is both impractical and infeasible due to bandwidth constraints and privacy concerns. FedAvg presents a middle ground by ensuring that while the model training occurs locally on each device, the learning is coordinated through a global shared model, represented as wt, where t denotes the communication round. Essentially, while individual clients use traditional methods like Stochastic Gradient Descent (SGD) for local training, the central idea behind FedAvg is to average these local updates at the server to refine the global model iteratively [69].

Recent benchmarking on FL algorithms offers insights into the performance of FedAvg. Nilsson et al. deployed the MNIST dataset, a popular benchmark for evaluating digit classification algorithms first introduced by LeCun (1998) , for their baseline experiments across a client-server setup [148], [149] . They compared the performance of multiple algorithms, focusing on aspects like communication efficiency and convergence (refer to Table X for an exhaustive comparison). Notably, FedAvg showcased impressive results when the total client communication was capped at 10,000, especially under i.i.d data management. Their findings, grounded in empirical data, attest to FedAvg's robustness in certain FL scenarios.

5.1.2.2 Mathematical Formulation of the Algorithm

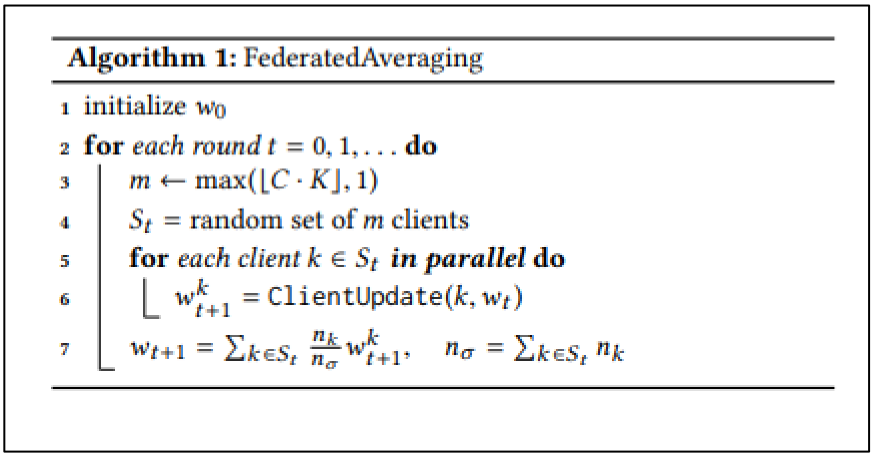
At the heart of FedAvg lies its mathematical structure, which is articulated through a series of steps and hyperparameters. FedAvg incorporates five principal hyperparameters: C, the fraction of clients chosen for an update; B, the mini-batch size; E, the number of local epochs; η, the learning rate; and potentially a learning rate decay, λ. The initial global model, w_0 , starts with a random introduction. For each communication round in FedAvg:

1. The server selects a subset of clients, S_t , with $|S_t|=C \cdot K \geq 1$.
2. The current global model w_t is then sent to all clients in S_t.
3. Every client updates its local model w_{kt} to be in coherence with the global model, $w_{kt} \leftarrow w_t$.
4. The local data is segmented into batches of size B, followed by local updates using SGD epochs.
5. After training, clients send their updated models w_{kt+1} back to the server.

6. The server aggregates these models into a new global model, w_{t+1} utilizing a weighted sum based on the number of local training samples.

This procedure is succinctly outlined in Algorithm 1, as seen in Table 1.

Table 1- Algorithm 1 Federated averaging [148]



```
Algorithm 1: FederatedAveraging
initialize $w_0$
for each round $t = 0, 1, \ldots$ do
    $m \leftarrow \max(\lfloor C \cdot K \rfloor, 1)$
    $S_t$ = random set of $m$ clients
    for each client $k \in S_t$ in parallel do
        $w^k_{t+1}$ = ClientUpdate($k, w_t$)
    $w_{t+1} = \sum_{k \in S_t} \frac{n_k}{n_\sigma} w^k_{t+1}, \quad n_\sigma = \sum_{k \in S_t} n_k$
```

5.1.2.3 Potential Benefits and Use Cases of FedAvg

FedAvg, from its inception, showcased a compelling blend of theoretical elegance and practical utility. The framework allows decentralized devices to learn locally while still contributing to a global understanding, representing a paradigm shift from traditional centralized models.

With FedAvg's local training approach, the overwhelming task of transferring vast amounts of data to a central server becomes redundant. This data efficiency not only preserves bandwidth but also ensures that models converge faster, optimizing the learning process in resource-constrained environments.

In an era where privacy breaches dominate headlines, FedAvg emerges as a beacon of user-centric data handling. Since raw data remains on the user's device and only model updates or aggregates are shared, the user's privacy is inherently safeguarded. This feature stands in stark contrast to traditional models that necessitate data pooling on central servers.

FedAvg's architecture finds resonance in many real-world scenarios: Given their proliferation and the vast amounts of data they generate, IoT devices can utilize FedAvg to learn from their environments without constantly uploading data. Modern smartphones, equipped with powerful processors, can engage in local learning, refining models based on user behavior.

Meanwhile, medical devices storing sensitive health data can benefit immensely from FedAvg, ensuring patient confidentiality while still contributing to broader medical research.

5.1.2.4 Limitations and Challenges of FedAvg

In decentralized computational frameworks, the FedAvg algorithm presents a notable approach, anchored in the collaborative input of multiple devices, each aiming to refine a comprehensive model. One inherent challenge in this setup is the disparate data distributions among these devices. Often, individual devices house datasets that don't necessarily align, embodying non-IID characteristics. This divergence can inadvertently affect the representativeness of the global model, potentially leading to a lack of comprehensive data capture and the introduction of unintended biases.

Further complicating the landscape is the asynchronous nature of model updates. Due to the vastness and variability of the network, devices don't always synchronize their updates. Some might be integrating the latest insights from the overarching model, while others could be basing their updates on prior versions. Such asynchronous operations introduce complexities to the data integration process, emphasizing the need for robust synchronization protocols.

Central to the performance of the FedAvg algorithm are its hyperparameters. These crucial settings, pivotal in guiding the learning trajectory, necessitate careful calibration. Misconfigurations can either cause the model to become overly tailored to specific local datasets or prevent it from recognizing subtle data patterns, both of which hamper its wider applicability.

Moreover, the distributed architecture inherent to FedAvg poses security challenges. Sharing model updates across a vast network creates potential vulnerabilities. There's a risk of adversarial entities exploiting the system, either to compromise the model's core functionality or to introduce biases. Ensuring robust security measures and maintaining the integrity of model updates are thus paramount for the algorithm's reliable performance [150].

5.1.3 Federated Stochastic Gradient

In the evolving landscape of decentralized learning, the Federated Stochastic Gradient (FSG) emerges as a pivotal technique. Situated at the crossroads of FL and optimization algorithms, FSG harnesses the power of stochastic gradient methods within a federated framework. This union promises enhanced computational efficiencies and robust model training across distributed devices. As we delve into section 5.1.3, we aim to unravel the intricacies of Federated Stochastic Gradient, exploring its foundational principles, potential advantages, and challenges in the broader context of decentralized computational systems.

5.1.3.1 Introduction and Reasoning for FedSGD:

Federated Stochastic Gradient Descent (FedSGD) is a decentralized version of the classical Stochastic Gradient Descent (SGD) adapted to the context of FL. Recognizing the privacy concerns and data distribution challenges in modern machine learning applications, FedSGD provides a mechanism to train machine learning models across multiple devices or nodes without the necessity to centralize data [151].

SGD, as the name implies, uses a stochastic approximation of the gradient of the objective function to update model parameters. In the federated context, each client (or device) computes the SGD locally using its data and then communicates the updates to a centralized server. The server aggregates these updates to improve the global model [151], [152].

5.1.3.2 Algorithmic Steps and Mathematical Background:

The algorithmic procedure can be described by natural language as the following steps [151], [152], [153]

1. **Initialization:** A global model is initialized, and its parameters are shared with all participating clients.
2. **Local Computation:** Each client computes the gradient of the model on its local data using SGD.
3. **Model Aggregation:** Clients send their computed gradients or model parameters to the central server.
4. **Global Model Update:** The central server aggregates these updates (typically by averaging) to adjust the global model.
5. **Distribution:** The updated global model is shared back with all clients.
6. **Iterations:** Steps 2-5 are repeated until convergence or for a predefined number of communication rounds.

While the mechanism may seem straightforward, several nuances, like handling communication inefficiencies, stragglers, and ensuring privacy, play a critical role in the actual implementation.

5.1.3.3 Challenges

The FedSGD, as an state-of-the-art technique in decentralized machine learning, carries a spectrum of prospects. Nevertheless, transitioning from its conceptual brilliance to real-world application is not without its trials. Central to the tenets of FL is the ceaseless exchange of model gradients between an array of clients and a centralized server. Yet, the diverse nature of network conditions often throws a spanner in the works, leading to potential delays or inconsistencies in these transmissions. Such challenges tend to magnify when we consider more extensive models or a large client base, with the cadence of these communications further straining bandwidth resources.

Within this vast federation, there's a lack of homogeneity in the operational tempo of nodes. Certain clients, constrained perhaps by their computational limits, network responsiveness, or intermittent availability, often find themselves lagging. These participants, colloquially termed 'stragglers', can inadvertently decelerate the updating of the global model. The central server, in anticipation of their input, might find itself in a holding pattern. Innovative solutions, such as imposing deadlines for updates or the formulation of adaptive algorithms, might hold the key to navigating this challenge.

But beyond operational challenges lies a more profound concern: privacy. The genesis of FL was significantly influenced by the imperative to uphold data confidentiality. However, this commitment to privacy can be paradoxical. Even in the absence of direct data sharing, the dissemination of model updates can inadvertently reveal sensitive data facets. Crafty adversaries, with the right tools and techniques, might discern or even recreate elements of individual datasets. Techniques like differential privacy or encrypted computations present a beacon of hope in this regard. Yet, seamlessly melding these with FedSGD, ensuring no compromise on efficiency or accuracy, remains an intricate endeavor [154].

In essence, while the allure of FedSGD as a beacon in decentralized machine learning is palpable, its full potential can only be harnessed by judiciously navigating these challenges, ensuring the methodology remains both adept and secure.

5.1.3.4 Comparison with FedAvg:

While both FedSGD and FedAvg are techniques in FL aiming to aggregate local model updates, the primary distinction is the method of aggregation. FedSGD focuses more on the gradient updates, sending these to the central server for aggregation. In contrast, FedAvg sends the model parameters themselves after local training. Essentially, while FedSGD is gradient-centric, FedAvg is model-centric in its aggregation approach [148], [155].

5.1.4 Model Aggregation Methods

In the realm of FL, model aggregation is the linchpin that consolidates the local updates from disparate devices or nodes into a unified global model. The quintessence of FL is the decentralization of model training across multiple devices, followed by the centralization of model aggregation. This section delves into the most important model aggregation techniques, highlighting their advantages, challenges, and appropriate application contexts.

5.1.4.1 Simple Averaging

The most straightforward method of aggregation is simple averaging, prominently used in the Federated Averaging (FedAvg) algorithm [156]. In this method, updates from each client (e.g., gradient updates or model parameters) are averaged to produce the global update. Mathematically, for N clients:

$$\theta_{global} = \frac{1}{N}\sum\nolimits_{i=1}^{N} \theta_i$$

Where θi represents the model parameters from the i^{th} client.

Advantages:

- Computational simplicity.
- Scalability to a large number of clients

Challenges:

- Assumes IID data across clients, which is often not the case.
- Susceptibility to adversarial attacks, as malicious clients can skew the average.

5.1.4.2 Weighted Averaging

An evolution of the simple averaging method is weighted averaging, where each client's updates are weighted by the number of samples it possesses [156]. This approach gives more importance to clients with more data, which can be beneficial in non-IID settings.

$$\theta_{global} = \frac{\sum_{i=1}^{N} w_i \theta_i}{\sum_{i=1}^{N} w_i}$$

Where w_i is the weight (often the number of samples) of the i^{th} client.

Advantages:

- More robust to non-IID data scenarios.
- Reduces the impact of stragglers or clients with fewer data samples.

Challenges:

- Still vulnerable to adversarial attacks if a malicious client possesses a significant amount of data.

5.1.4.3 Geometric Median Aggregation

Instead of averaging, some aggregation techniques aim to find the geometric median of the model updates . The geometric median offers robustness against adversarial or Byzantine attacks, as it's less susceptible to outliers.

Advantages:

- Robustness against adversarial updates.

- Suitable for scenarios where security and integrity are paramount.

Challenges:

- Computationally more intensive than averaging methods.
- Determining the geometric median in high-dimensional spaces can be challenging.

5.1.4.4 Personalized Aggregation

Recent advances have proposed methods for personalized model aggregation, where the global model is fine-tuned or adjusted based on individual client characteristics. This approach acknowledges the heterogeneity of data and user behaviors across clients.

Advantages:

- Better model performance on individual clients.
- Addresses the diverse nature of federated data sources.

Challenges:

- Can lead to overfitting if not properly regulated.
- More complex than traditional aggregation techniques**.**

5.1.4.5 Other Aggregation Methods

The tapestry of FL is rich and varied, with model aggregation techniques acting as the threads that weave individual learning experiences into a coherent global understanding. Beyond the foundational techniques of simple and weighted averaging, there exists a plethora of aggregation methodologies tailored for specific challenges and goals, as described in [156].

The **Average Aggregation** serves as the bedrock, a method that straightforwardly computes the mean of updates received from client nodes. This foundational approach sets the stage for more nuanced techniques. Venturing a step further, Clipped Average Aggregation introduces an element of refinement to the averaging process. By constraining model updates within a predefined range prior to aggregation, this method mitigates the potential distortions introduced by outliers or malicious entities.

In a world that prioritizes data security, **Secure Aggregation** emerges as a beacon, integrating cryptographic techniques like homomorphic encryption and secure multi-party computation. The result is a fortified aggregation process where client data remains shielded throughout. Building upon the principles of data privacy, the **Differential Privacy Average Aggregation** infuses the aggregation with an additional layer of confidentiality. By blending client updates with carefully calibrated noise, this method strikes a balance between data privacy and model fidelity.

Recognizing the iterative nature of learning, the **Momentum Aggregation** captures the historical trajectory of model changes. By appending a momentum term, indicative of past model shifts, to new updates, this method aims to expedite convergence. The realm of **Bayesian Aggregation** offers another perspective, viewing model updates through the lens of Bayesian inference, thereby accommodating uncertainties in model parameters.

As the landscape of FL becomes more intricate, the need to safeguard against adversarial entities becomes paramount. **Adversarial Aggregation** rises to this challenge, employing a gamut of techniques to discern and neutralize malicious updates. Techniques such as outlier detection and model-based anomaly recognition bolster the defenses.

On the practical front, **Quantization Aggregation** addresses the logistical challenges of data transmission. By distilling model updates into a more compact form before transmission, this method optimizes communication efficiency. **Hierarchical Aggregation** takes a macroscopic view, orchestrating the aggregation in a tiered manner, mirroring hierarchical structures. This multi-level approach curtails communication overhead, enabling efficient local aggregations before the synthesis at higher echelons.

Model aggregation is pivotal in FL, determining the efficacy, robustness, and resilience of the global model. The choice of aggregation method should be contingent on the nature of the federated data, the underlying network architecture, and the specific challenges posed by the deployment scenario. As FL continues to mature, novel aggregation techniques that address its inherent challenges will be instrumental in its widespread adoption.

5.1.5 Communication-Efficient Algorithms

In the evolving landscape of FL, the spectrum of its deployment spans an impressive range, from data-intensive centers to modest edge devices. This broad implementation accentuates a central challenge: ensuring communication efficiency. Within such distributed systems, communication frequently establishes itself not merely as a component but as a potential bottleneck. Addressing this bottleneck is critical, not just for the optimal performance but also for the broader scalability and applicability of FL in diverse real-world settings (1,2) .

While model training remains at the heart of FL, the success of this training is intertwined with the quality of communication between the nodes and the central server. Especially in expansive federated networks, the volume of data to be transmitted can be daunting. This challenge amplifies in constrained environments, such as those characterized by mobile networks or satellite connections. In these settings, the premium is on bandwidth availability, and latency can be a persistent concern. Optimizing communication in these contexts promises a trifecta of benefits. Firstly, there's a direct financial implication: efficient data transmission can lead to notable cost savings, particularly in scenarios tethered to bandwidth costs. Secondly, from a process perspective, reduced synchronization delays can expedite the overall training, fostering quicker model convergence. Lastly, there's the dimension of energy. For battery-operated devices, efficient communication translates to power conservation, extending their operational longevity within the federated framework.

Delving into the strategies aimed at enhancing communication efficiency in FL, several nuanced approaches emerge. **Sparsification**, for instance, hinges on the principle of selective communication. By focusing on transmitting only the most pivotal updates, this method seeks to streamline data flow. Techniques under sparsification might involve threshold-based criteria or the selective relay of top-ranking parameters. However, a challenge that surfaces here is the calibration of these selection criteria, ensuring that communication reduction doesn't compromise model integrity [122], [157].

Quantization offers another avenue, emphasizing data compression. Advanced incarnations of this method encompass techniques like scalar quantization, which maps parameters to discrete scalar values, and gradient bucketing, which quantizes grouped gradients. While promising in terms of data size reduction, they introduce the challenge of accurate data reconstruction post-quantization [158].

Then there's the domain of coding techniques, which marries FL with principles from information theory. By integrating methods like error-correcting codes or gradient coding, the objective is to enhance the robustness of data relay. Such techniques infuse the system with resilience against transmission errors or computational lags, albeit at the cost of potential computational overheads and intricate code design complexities.

In conclusion, the endeavor to architect communication-efficient algorithms in FL is a complex tapestry of system optimization, model fidelity, and practical constraints. As FL continues its expansion, adapting to a myriad of environments and challenges, the iterative refinement of these communication strategies will be pivotal. The horizon beckons for agile algorithms, ones that resonate with varying network dynamics, ever-evolving model prerequisites, and the kaleidoscope of device capabilities, reinforcing the stature of FL as a cornerstone in the edifice of machine learning.

5.1.6 Adaptive Learning Rates in Federated Contexts

In the expansive arena of machine learning, the learning rate stands as a pivotal hyperparameter, determining the steps' size that an optimization algorithm takes in search of a minima. Too large, and it risks overshooting; too small, and it may get trapped in local minima or converge painstakingly slowly. In FL contexts, these challenges are accentuated due to the distributed nature of the data and the need to aggregate diverse model updates from various devices. This section delves into the realm of adaptive learning rates in federated settings, exploring their significance, the algorithms tailored for them, and the associated benefits and challenges.

5.1.6.1 Federated Adaptive Algorithms and the Importance of Adaptive Learning in a Federated Setting

Traditional centralized learning benefits from uniform and often IID (independently and identically distributed) data. However, FL grapples with non-IID data, sourced from diverse devices with possibly differing distributions. This heterogeneity can lead to disparate model updates, demanding a more nuanced approach to learning rates. Instead of a uniform rate, the need arises for adaptive mechanisms that can cater to each client's unique data distribution and model progression. Adaptive learning rates provide the flexibility needed to accommodate such diverse conditions and lead to faster and more stable convergence in federated settings [159], [160].

Several adaptive algorithms have been proposed and adapted for the FL paradigm. The following two can be considered as the most important:

Federated Adagrad: Building upon the Adagrad algorithm, which adapts the learning rate based on the historical gradient information, Federated Adagrad customizes this for the decentralized data in FL. It adjusts learning rates for each client based on their unique data gradients, ensuring personalized optimization [161].

Federated Adam: Inspired by the Adam optimizer, which combines the advantages of Adagrad and RMSprop, Federated Adam brings this adaptive moment estimation to federated contexts. It maintains running averages of both gradients and their squares for each client, allowing a more balanced and informed learning rate adaptation.

Both Federated Adagrad and Federated Adam introduce dynamic and adaptive learning rates in FL, taking into account the specific gradient characteristics at each client. While they promise improved convergence and stability, considerations about communication overheads and the heterogeneity of client data need to be intricately managed [147].

Currently, the importance of adaptive learning rates is becoming increasingly evident. Such rates offer a plethora of benefits. For instance, by customizing learning rates based on the gradient profiles of each client, adaptive algorithms can achieve faster convergence to optimal solutions. This adaptability also lends itself to improved stability by curbing oscillations in the loss landscape, ensuring a smoother path to optimization. Given the inherent heterogeneity in client data distributions and model states in federated contexts, adaptive learning rates rise to the occasion by effectively catering to these diverse scenarios. Additionally, they lessen the reliance on initial learning rate settings, providing a level of resilience against potentially suboptimal hyperparameter choices.

However, while adaptive learning rates promise numerous advantages, they come with their own set of challenges, especially in a federated setting. For one, there's the computational overhead. Devices, especially those with limited resources, might find it taxing to maintain historical gradient information or moment estimates. Furthermore, when there's a need to synchronize such data with a central server, the communication overhead can become a significant concern. Moreover, in situations characterized by highly non-IID data, there's a risk that aggressive learning rate adjustments could lead to divergence if not judiciously monitored. And, even with their inherent adaptability, these algorithms aren't entirely free from the need for hyperparameter tuning. For instance, parameters like the beta values in the Adam optimizer might still require adjustments to suit specific federated contexts [147].

This discussion is further enriched by findings from a study presented in [161]. The research therein delves deep into communication-efficient strategies such as sparsification and quantization. These methods, while holding immense potential, grapple with challenges such as ensuring model integrity post-quantization and the accurate reconstruction of data. The study also introduces coding techniques derived from information theory, aiming to bolster data transmission. Yet, these techniques are not exempt from challenges, with design complexities and computational overheads being notable concerns. An extensive analysis of various optimizers on datasets like COIL-100, Caltech-101, and MNIST underscored the preeminence of the Adagrad optimizer, which consistently outperformed its counterparts. In sum, while Gradient Descent remains a cornerstone for neural network training, there's an increasing recognition of the Adagrad optimizer's efficacy, particularly for diverse image datasets.

As the landscape of FL continues to evolve, striking a balance between the dynamism of adaptive learning rates and addressing their inherent challenges will be paramount. The future of FL will undoubtedly see algorithms that further refine adaptability, judiciously manage computational and communication overheads, and navigate the complexities of distributed data scenarios.

5.1.6.2 Federated Adagrad

Introduction and Context:

Adagrad (Adaptive Gradient Algorithm) is an algorithm specifically designed to improve the convergence performance by adapting learning rates to the parameters, employing larger updates for infrequent and smaller updates for frequent parameters. Federated Adagrad brings this adaptive learning rate mechanism to the decentralized realm of FL.

Mathematical Framework:

Given a gradient g t for a parameter at time step t, Adagrad modifies the general learning rate η at each time step for every parameter based on the past gradients:

$$G_t = G_{t-1} + g_t^2$$
$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{G_{t+\epsilon}}} \cdot g_t$$

- Where: G_t is the sum of squares of past gradients up to time step *t*.
- ϵ is a smoothing term to prevent division by zero (typically a very small value, e.g., *1e−8*).
- θ_t is the parameter at time step *t*.

In a federated context, each client would compute its own G_t based on its local data and then send the necessary information to the server, which would then aggregate these updates [162].

Challenges in Federated Settings:

- The potential for more communication overhead as both gradients and squared gradient accumulations might need to be communicated.
- Differences in the magnitude and direction of client updates, due to non-IID data, can cause issues in global aggregation.

5.1.6.3 Federated Adam

Introduction and Context:

Adam (Adaptive Moment Estimation) is an optimization algorithm that computes adaptive learning rates for each parameter by considering the first and second moments of the gradients.

The federated version of Adam incorporates this adaptive methodology in distributed settings [163].

Mathematical Framework:

Adam maintains two running averages for each parameter:

- The first moment (mean) of the gradient, denoted as m_t.
- The second raw moment (uncentered variance) of the gradient, denoted as v_t.

The running averages are computed as:

$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$
$v_t = \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$

- Where: β_1 and β_2 are hyperparameters that control the exponential decay rates of the running averages. Typically, values are set close to 1 (e.g., $\beta_1 = 0.9$, $\beta_2 = 0.999$).
- g_t is the gradient at time step t.

Adam then applies bias correction to these moments:

$$\widehat{m_t} = \frac{m_t}{1 - \beta_1^t}$$

$$\widehat{v_t} = \frac{v_t}{1 - \beta_2^t}$$

The parameter update is then given by:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\widehat{v_t}} + \epsilon}$$

In the federated context, each client calculates its local updates and running averages, which are then aggregated at the server level.

Challenges in Federated Settings:

- Increased communication overhead as clients might need to transmit more information (both moments) to the server.
- Similar to Federated Adagrad, the non-IID nature of data can cause aggregation discrepancies at the server level [161].

5.1.7 Asynchronous Methods and Algorithms

FL inherently stands on the promise of harnessing data from myriad devices distributed across geographies and functionalities, and it is imperative that these devices need not always operate in synchrony. This calls for methods that allow devices or clients to update the global model in an asynchronous fashion. This section dives into the nuances of asynchronous methods in

FL, their motivation, detailed exposition, comparison to synchronous counterparts, and the associated advantages and challenges.

The diverse nature of devices participating in FL necessitates mechanisms that can accommodate varied network conditions, computational capacities, and availability. Asynchronous methods offer flexibility in this realm, allowing for a more organic flow of updates without a rigid structure. Here we dive deeper into the intricacies of asynchronous strategies in F.

5.1.7.1 Motivation for Asynchronous Updates in FL

The complexities of real-world distributed systems present unique challenges that asynchronous methods can address:

Network Disparities: In environments where devices span a wide geographical area, network speeds and stability vary drastically. A smartphone in an urban setting with 5G connectivity contrasts starkly with an IoT sensor in a remote location reliant on low-bandwidth connections. Asynchronous updates respect these disparities, allowing each device to communicate based on its optimal conditions [2].

Computational Constraints: Devices participating in FL range from powerful servers to resource-constrained sensors. Expecting them to process and deliver updates concurrently becomes unreasonable. Asynchronous methods provide an avenue for each device to contribute based on its computational pace [2].

Intermittent Availability: Devices might have sporadic connectivity or might be set to participate only at specific intervals (e.g., during low-usage times). Asynchronous updates accommodate such varied participation patterns, ensuring that every device, no matter its schedule, contributes to the global model.

5.1.7.2 Explanation of Asynchronous Algorithms

Asynchronous approaches, in their essence, are about flexibility and adaptability [2]:

- **Staleness:** In an asynchronous setting, updates from slower clients might become outdated or "stale" by the time they reach the central server. Understanding and managing this staleness is critical. For instance, if an update is based on a much older global model, should it be given the same weight as a more recent update?

- **Decentralized Aggregation:** Traditional FL aggregates client updates in batches, typically after receiving updates from all (or a majority of) clients. Asynchronous methods differ in that they don't wait. The server continually updates the model, integrating new data as it arrives.

5.1.7.3 Comparison with Synchronous Updates

While both methods have their merits, a nuanced understanding aids in their appropriate application [2], [122] :

- **Latency:** Asynchronous methods, by design, aim to reduce latency. Instead of waiting for every participant, they capitalize on available data immediately, promoting a more fluid model evolution.

- **Communication Efficiency:** Waiting for straggling devices in synchronous updates can be resource-intensive. Asynchronous updates eliminate this waiting period, leading to potentially fewer communication rounds and faster overall learning.

- **Convergence:** The predictable nature of synchronous updates often leads to more stable convergence patterns. In contrast, the dynamic and unpredictable flow of asynchronous updates can sometimes pose challenges in ensuring stable and consistent model convergence.

- **Scalability:** With the rise in the number of devices, synchronous methods may become impractical. Asynchronous methods inherently support large-scale deployments, seamlessly incorporating updates from an ever-growing participant pool.

5.1.7.4 Benefits and Challenges

Asynchronous methods come with a distinct set of pros and cons:

Benefits:

- **Real-time Learning:** The global model can adapt almost instantaneously to changes, making it highly responsive and potentially more accurate.

- **Efficiency:** The continuous flow of updates ensures optimal utilization of server resources, reducing idle times.

- **Scalability:** As networks grow, asynchronous methods naturally adapt, handling more clients without necessitating structural changes.

Challenges:

- **Staleness Management:** Ensuring that older updates don't adversely affect the model requires intricate management strategies, potentially complicating the learning process.

- **Overhead Complexity:** Continuously integrating diverse updates might lead to overheads in terms of computational resources and complexity in aggregation logic.

- **Inconsistency:** Without careful management, the global model might reflect transient states more more than a consistent representation of the entire network's knowledge.

Embracing the chaotic yet organic nature of asynchronous updates can greatly benefit FL environments, especially those spanning vast and varied networks. However, the challenges they introduce necessitate robust strategies and architectures that can ensure consistent, efficient, and accurate global model evolution.

5.1.8 State-of-The-Art FL Algorithms: An Indicative Comparative Benchmark against FedAvg

5.1.8.1 Introduction

Within this subsection, we delve deep into two such promising algorithms - CO-OP and Federated Stochastic Variance Reduced Gradient (FSVRG). By introducing these algorithms, we aim to furnish the reader with a comprehensive understanding of their foundational principles, mechanisms, and unique features, and showcase optimized versions of the classic algorithms discussed before.

However, understanding their theoretical underpinnings is only a part of the larger puzzle. Practical implementations and performance metrics speak volumes about the actual utility and efficiency of any given algorithm [164]. With this in mind, we extensively study, and further enrich this section, by discussing in detail a selected indicative benchmarking analysis from literature. In this comparison, CO-OP and FSVRG will be pitted against the renowned FedAvg, a baseline algorithm in the FL domain.

After understanding the basic essence of FSVRG and CO-OP, and by then examining their benchmark against FedAVG, readers will hopefully gain insights into potential performance improvements, the scalability of these algorithms, and their suitability for various applications. The goal is to illuminate the strengths and weaknesses of FedAVG and each algorithm in a real-world context.

5.1.8.2 Federated Stochastic Variance Reduced Gradient (FSVRG)

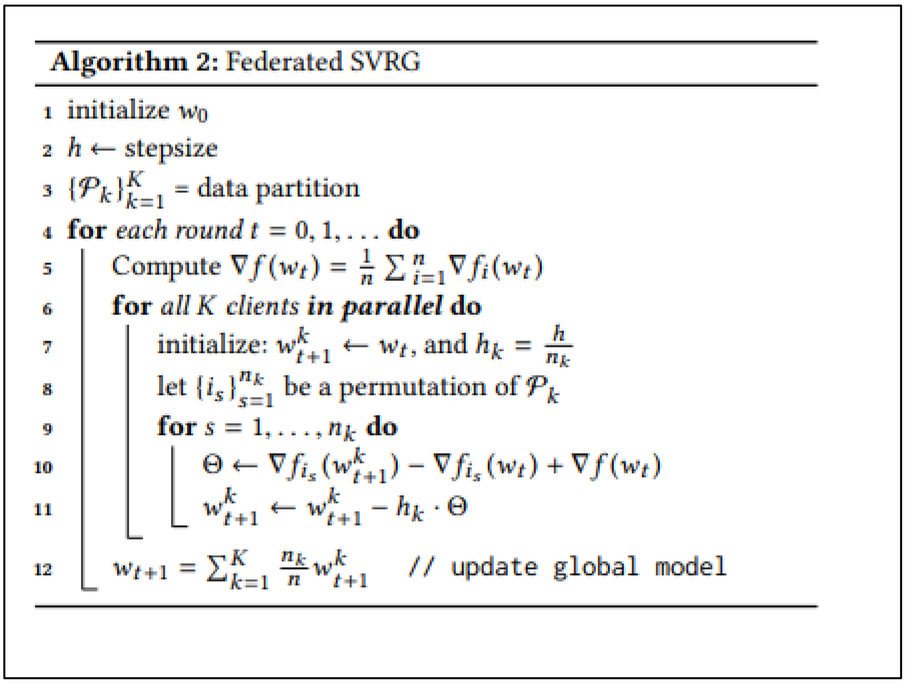
FSVRG, introduced by Johnshon & Zhang (2013), and inspired by the Unified Stochastic Fluctuation Reduced Slope, aims to address the inherent challenges of FL, particularly the distributed nature of data [165]. The core philosophy behind FSVRG is the amalgamation of an initial in-depth gradient evaluation with subsequent iterative stochastic updates on each participating client. By doing so, it captures the global data structure while factoring in specific intricacies of each client's dataset, offering a balance between generalization and specialization.

Algorithmic Steps and Mathematical Background:

FSVRG is predicated on a balance between computational efficacy and data accuracy. The process commences with clients procuring the current model, w_t. Using this model as a reference, they compute gradients based on their localized data. This locally computed data is then channeled to a central server for aggregation, resulting in a holistic gradient computation, symbolized as $f(w_t)$. This global perspective is then relayed back to the clients. They, in turn, adjust their local models, denoted as w_{kt}, in accorprocess with this aggregated gradient. The subsequent phase involves the clients undergoing several Stochastic Variance Gradient

Gradient (SVGG) iterations. Collectively, these steps culminate in the formulation of an updated global model, represented as *wt+1*, resonating with the FedAvg paradigm.

Table 2 - SVRG Algorithm [148]



Algorithm 2: Federated SVRG

```
initialize $w_0$
$h \leftarrow$ stepsize
$\{\mathcal{P}_k\}_{k=1}^{K}$ = data partition
for each round $t = 0, 1, \ldots$ do
    Compute $\nabla f(w_t) = \frac{1}{n}\sum_{i=1}^{n}\nabla f_i(w_t)$
    for all $K$ clients in parallel do
        initialize: $w_{t+1}^{k} \leftarrow w_t$, and $h_k = \frac{h}{n_k}$
        let $\{i_s\}_{s=1}^{n_k}$ be a permutation of $\mathcal{P}_k$
        for $s = 1, \ldots, n_k$ do
            $\Theta \leftarrow \nabla f_{i_s}(w_{t+1}^{k}) - \nabla f_{i_s}(w_t) + \nabla f(w_t)$
            $w_{t+1}^{k} \leftarrow w_{t+1}^{k} - h_k \cdot \Theta$
    $w_{t+1} = \sum_{k=1}^{K}\frac{n_k}{n}w_{t+1}^{k}$    // update global model
```

Both FedSGD and FSVRG share the overarching goal of refining a global model through local client computations. However, the nuances of their methodologies offer different perspectives. While FedSGD focuses on synchronizing local updates to shape a centralized model, FSVRG integrates a preliminary variance reduction phase via comprehensive gradient calculations. This strategy equips FSVRG to handle datasets characterized by pronounced variability. Nonetheless, this additional computational layer implies that FSVRG could demand more computational resources than its FedSGD counterpart, especially during the initial gradient assessment and the ensuing stochastic rounds.

Benchmarking studies by Nilsson et al. (2013) underscore the efficiency of FSVRG in federated environments. The results showed that FSVRG eventually outperformed FedAvg in terms of execution, highlighting its adaptability and capability. This empirical evidence demonstrates the strengths of FSVRG in particular contexts. Their extensive results, as illustrated in Fig. 10 and Fig. 11, demonstrate how FSVRG eventually gains an edge over FedAvg in certain conditions. Such empirical insights spotlight the distinct advantages of FSVRG in specific settings [148].

5.1.8.3 CO-OP Algorithm

Introduction and Framework Overview:

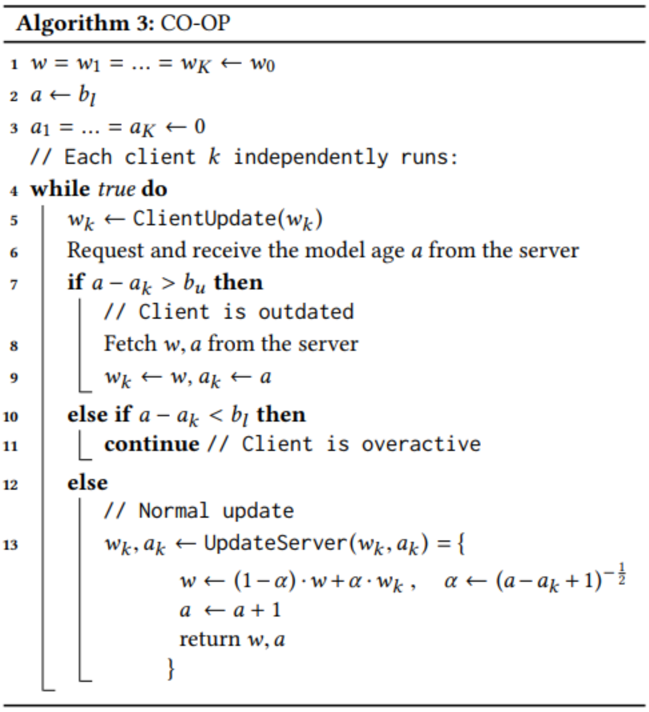
The asynchronous CO-OP protocol emerges as a novel approach to handling the complexities of distributed machine learning, particularly in dynamic environments. Instead of relying on static datasets, as is common with many existing FL frameworks, CO-OP is adeptly designed to adapt to dynamically generated data. As users interact with their mobile devices, the individual datasets of the users grow in real-time, introducing an added layer of complexity.

Consider a system where K mobile clients are actively participating in a cooperative learning process. Each client, governed by the unique interaction patterns of its user, gathers its own set of data. Once a pre-defined threshold (B samples, known as the local batch size) is reached, the client begins the process of refining its local model, employing gradient descent techniques based on this recently accumulated data.

Traditional FL frameworks often employ a more centralized approach, wherein the server periodically prompts specific clients to contribute to model updates. CO-OP, however, champions decentralization by empowering clients to initiate model updates asynchronously. This autonomy allows clients to choose the best time and environment for local training, such as during optimal network conditions. After refining their local model, clients can then liaise with the central server, merging their updates and subsequently fetching the revised global model.

Model and Evolution:

Table 3 – CO-OP Algorithm [148]



```
Algorithm 3: CO-OP
$w = w_1 = \ldots = w_K \leftarrow w_0$
$a \leftarrow b_l$
$a_1 = \ldots = a_K \leftarrow 0$
// Each client $k$ independently runs:
while true do
    $w_k \leftarrow$ ClientUpdate($w_k$)
    Request and receive the model age $a$ from the server
    if $a - a_k > b_u$ then
        // Client is outdated
        Fetch $w, a$ from the server
        $w_k \leftarrow w, a_k \leftarrow a$
    else if $a - a_k < b_l$ then
        continue // Client is overactive
    else
        // Normal update
        $w_k, a_k \leftarrow$ UpdateServer($w_k, a_k$) = {
                $w \leftarrow (1-\alpha) \cdot w + \alpha \cdot w_k$,    $\alpha \leftarrow (a - a_k + 1)^{-\frac{1}{2}}$
                $a \leftarrow a + 1$
                return $w, a$
            }
```

The global model parameters on the server are denoted as w, with each client's parameters being represented as wk . One of the pivotal metrics within the CO-OP framework is the 'age' of the global model, signifying how often it has integrated updates from various clients. Parallelly, each client monitors its own 'model age', which gets recalibrated according to the CO-OP protocol.

Commencing with a universally adopted model w, clients within the CO-OP ecosystem continuously accumulate data, finetune their models, and when conditions are ripe, synchronize with the server. The protocols hallmark is its 'age filter', ensuring that client model integrations are timely and relevant, eliminating outdated or redundant contributions.

As FL advances, the foray into newer aggregation methods, like CO-OP, is a testament to the evolving needs and challenges of distributed learning. Distinct from traditional algorithms, CO-OP is poised to mitigate discrepancies between server and client data, catering to both i.i.d. and non-i.i.d. data setups.

With empirical studies and benchmarking, the effectiveness of CO-OP comes into focus. Using standardized datasets like MNIST, researchers have critically assessed CO-OP against mainstays like FedAvg and FSVRG. Preliminary results indicate that while CO-OP offers a fresh approach to aggregation, it encounters robust competition, especially from entrenched algorithms.

In some studies., despite CO-OP's innovative mechanics, it couldn’t consistently surpass the performance of algorithms like FSVRG. Such findings underscore the inherent complexities in FL, highlighting the perpetual quest for improved algorithms.

In the panorama of FL, the CO-OP aggregation method illuminates the path forward with its novel strategies. Drawing from empirical studies and evaluations using established datasets, CO-OP stands as a testament to the ongoing evolution of decentralized machine learning. Although promising, its journey towards becoming the go-to choice in diverse federated scenarios is still unfolding.

5.1.8.4 Benchmarking Case Results

MNIST dataset, is a collection of handwritten digits, which has served as a foundational testing set for neural networks over the years. Nilsson et al. utilized the MNIST dataset for their baseline experiments [148]. The authors approached the distribution of this data amongst clients in two ways: Independent and Identically Distributed (i.i.d.) and non-i.i.d. This distinction is illustrated in Figure 9, displaying a side-by-side comparison of both data distributions from one of the client cases.

In the benchmarking method of Nilsson et al., a total of 100 clients, spread over three machines, were deployed. One device functioned as the central server. Notably, the performance of 'CO-OP', a primary algorithm, shifts based on how these clients are distributed across the machines. Clients situated on the same machine as the server generally experience more rapid upload speeds because of the lack of Ethernet communication. Such a dynamic can impact the performance of the 'CO-OP' algorithm, potentially causing some clients to seem unusually active and leading to inactive clients on different machines.

Throughout their benchmarking, the authors ensured a consistent setup where machines harbored comparable client simulations. While algorithms like FedAvg and FSVRG saw their global models evaluated at the culmination of every matching round, for CO-OP, Nilsson et al. chose to evaluate the global model only after every 10th update. This choice is grounded on the relatively minor progress made by a single client update in CO-OP in contrast to a full round of synchronous communication.

Data Distribution and Evaluation Approach

The difference between i.i.d. and non-i.i.d. data distributions becomes apparent when examining the manner in which data is allocated among clients. Employing the non-i.i.d. method renders a more lifelike scenario, dividing data into uniform batches and distributing segments arbitrarily among clients. These differences are depicted in Figure 9.

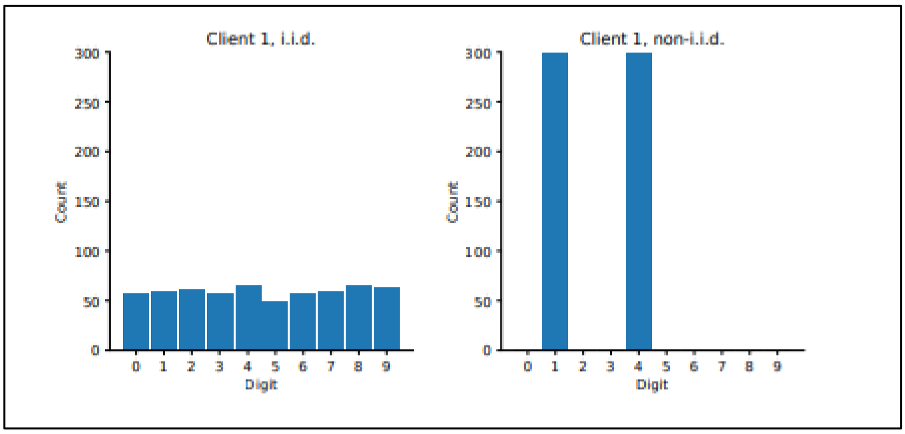


Figure 9 - Side-by-side comparison of an I.I.D. and non-I.I.D. distribution taken from one of the clients [148]

The authors' evaluation method included both i.i.d. and non-i.i.d. data distributions for their experimental framework. For the i.i.d. variant, the entire MNIST dataset, inclusive of the test set, was randomized and segmented into five equal portions, ensuring every client obtained an even assortment of images. For the non-i.i.d. version, the data handling diverged, emphasizing particular values or standards.

Performance Insights and Observations

Figure 10 exhibits the nuanced performance variances of 'CO-OP', particularly when juxtaposed with other classifiers. Even with its pioneering aggregation mechanisms, 'CO-OP' encounters stiff rivalry from more established algorithms.

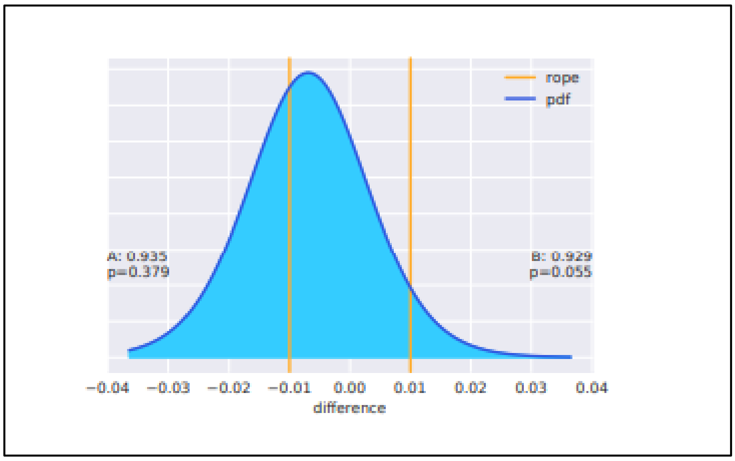
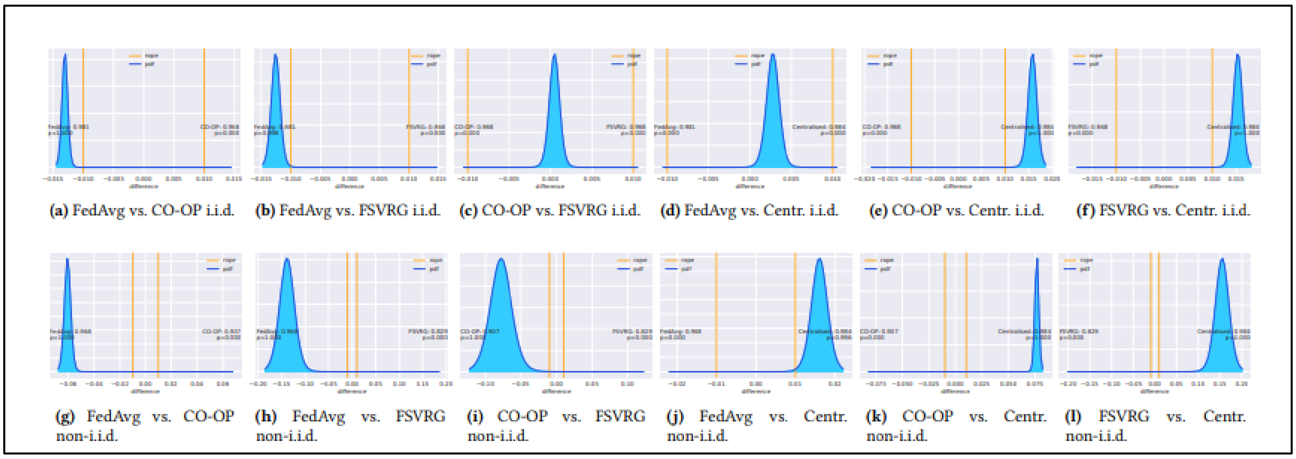


Figure 10 - Posterior probability distribution of a correlated Bayesian t-test between classifiers A and B. It defines a region of practical equivalence where the mean difference in accuracy is no more than ±1%. [148]

Nevertheless, as vividly portrayed in Figure 11, a primary observation from the study of Nilsson et al. is that 'CO-OP', in spite of its avant-garde methodology, fell short of FSVRG in their experimental framework. This underlines the intricacies and obstacles present in FL and accentuates the persistent demand for advancements in algorithms.



(a) FedAvg vs. CO-OP i.i.d. **(b)** FedAvg vs. FSVRG i.i.d. **(c)** CO-OP vs. FSVRG i.i.d. **(d)** FedAvg vs. Centr. i.i.d. **(e)** CO-OP vs. Centr. i.i.d. **(f)** FSVRG vs. Centr. i.i.d.

(g) FedAvg vs. CO-OP non-i.i.d. **(h)** FedAvg vs. FSVRG non-i.i.d. **(i)** CO-OP vs. FSVRG non-i.i.d. **(j)** FedAvg vs. Centr. non-i.i.d. **(k)** CO-OP vs. Centr. non-i.i.d. **(l)** FSVRG vs. Centr. non-i.i.d.

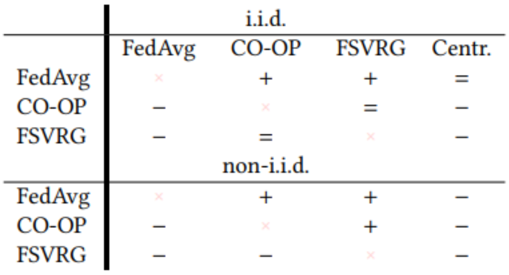
Figure 11 - Comparisons between federated optimization algorithms and centralized learning in the form of posterior distributions on MNIST I.I.D. and non-I.I.D. Each algorithm ran three iterations of 5-fold cross-validation. Note that the x-axes have different scales [148]

Conclusive Remarks

The "CO-OP" aggregation technique introduces an innovative perspective in the realm of FL, strengthened by empirical investigations and revered datasets like MNIST. However, its relative performance denotes an ongoing journey of enhancement and progress before achieving universal acclaim as the best solution across diverse federated scenarios.

Lastly, as detailed in Table 1, a synopsis of algorithm comparisons sheds light on the respective pros and cons of each, suggesting avenues for further inquiry and refinement within the FL sphere.

Table 4 - Summary of algorithm comparisons, showing if the algorithm in a row is better (+), worse (−), or practically equivalent (=) compared to the algorithm in a column [148]



	i.i.d.			
	FedAvg	CO-OP	FSVRG	Centr.
FedAvg	×	+	+	=
CO-OP	−	×	=	−
FSVRG	−	=	×	−
	non-i.i.d.			
FedAvg	×	+	+	−
CO-OP	−	×	+	−
FSVRG	−	−	×	−

5.2 Frameworks and Tools

The implementation and deployment of FL have seen significant enhancements with the introduction of dedicated frameworks and tools. These tools, tailored to address the intricacies of federated architectures, are not only essential for streamlining the process but are also pivotal for ensuring robustness and security. In the sections below, we explore three prominent frameworks that have been instrumental in shaping the landscape of FL: TensorFlow Federated, PySyft, and FATE.

5.2.1 TensorFlow Federated (TFF)

TensorFlow Federated emerges from the illustrious lineage of the TensorFlow framework. As a specialized extension, TFF brings the strengths of TensorFlow into the domain of FL. The framework boasts a comprehensive ecosystem, which means that developers have access to a vast library of functions and tools specifically crafted for federated scenarios. One of the standout features of TFF is its local simulation environment. This environment is invaluable for developers, providing them with a sandbox to iteratively refine and test their federated models without the hassles of a full-fledged deployment.

The advantages of TFF are manifold. Its seamless integration with TensorFlow means that developers familiar with TensorFlow can transition into the federated realm with minimal friction. Furthermore, owing to the widespread adoption of TensorFlow in the machine learning community, TFF benefits from extensive documentation, community-driven content, and a plethora of tutorials. However, every tool has its challenges. The richness of TFF can also be its Achilles' heel, especially for newcomers. The vastness of the TensorFlow ecosystem can sometimes result in a steep learning curve. That said, the rewards for scaling this learning curve are plentiful l [166], [167].

5.2.2 PySyft

PySyft marks its distinct space in the federated world by offering tools not just for FL but also for encrypted and private machine learning. This framework is an extension to popular deep learning platforms like PyTorch and TensorFlow, thus enabling multi-party computations. What sets PySyft apart is its focus on decentralized deep learning. Its flexible API is crafted with an emphasis on differential privacy and encrypted computations, thus ensuring that the privacy concerns intrinsic to FL are well-addressed [168].

The versatility of PySyft is one of its shining attributes. Given its capabilities that extend beyond FL, it presents itself as a swiss-army knife for any practitioner interested in privacy-preserving machine learning. Its integration with PyTorch, a favorite among the deep learning community, offers a familiar setting, which is always a boon. However, PySyft is relatively nascent compared to TensorFlow, and this youth can sometimes show in the form of features that might be in beta or underdeveloped [169]. Despite this, the community behind PySyft, led by OpenMined, is vibrant, passionate, and constantly pushing the boundaries.

5.2.3 FATE (Federated AI Technology Enabler)

FATE stands as a comprehensive solution in the FL landscape. Designed to be an end-to-end solution, it spans the entire lifecycle of FL, right from data preprocessing to model deployment. One of FATE's unique propositions is its focus on cross-platform support. Recognizing the heterogeneous nature of data sources in federated settings, it is built from the ground up to support varied data sources and diverse computing environments [170], [171].

The scalability of FATE is commendable. Whether it's large-scale datasets or complex models, FATE is constructed to handle them with finesse. An emphasis on secure exchange protocols ensures that data security, especially during inter-party exchanges, is never compromised. However, as with any sophisticated tool, FATE comes with its set of challenges. The broad range of features it offers can sometimes be daunting for beginners, translating to a steeper learning curve. Yet, as the community around FATE grows, and as more practitioners adopt it, it's expected that the collective knowledge will make the adoption of FATE smoother.

In summation, the FL space is enriched by the availability of these frameworks, each with its unique strengths and challenges. The onus is on practitioners to align their specific needs with the capabilities of these frameworks, ensuring that the chosen tool not only addresses the immediate requirements but is also poised for future challenges and expansions in the realm of FL.

Chapter 6: Attack Strategies

6.1 Introduction

In the age of increasing interconnectedness and digital collaboration, FL emerges as a groundbreaking approach that allows model training across multiple devices while keeping data localized. FL emerged as a beacon of hope for a world teetering on the balance of data utility and privacy. By promising localized data processing and decentralized training, FL appeared poised to resolve the enduring privacy concerns associated with centralized machine learning models. However, like all technologies, FL is not impervious to vulnerabilities, and model inversion attacks represent one of its critical challenges.

However, with these advancements come vulnerabilities, some specific to federated architectures, some inherited from the broader realm of machine learning. An understanding of these vulnerabilities and the potential attacks that exploit them is pivotal for the secure deployment and scalability of FL systems. In this Chapter we will cover a vast spectrum of the most well-known attack strategies, presenting in brief the idea behind them, the dangers they introduce in the context of FL, as well as mitigation techniques and policies that can reinforce FL systems against them.

6.1.1 Contextualizing Attacks in FL

FL has revolutionized the machine learning landscape, introducing a decentralized model that champions data privacy and reduces the need for massive data transfers. By keeping the data localized at its source and merely sharing model updates, FL promises a world where insights can be collectively harnessed without compromising the sanctity of individual data points. Yet, every silver lining has a cloud, and for FL, the challenges introduced by its decentralized architecture are considerable and demand our attention.

Historically, centralized machine learning models were susceptible to adversarial threats, where malicious actors would manipulate data or the model itself, striving to compromise the system's integrity or gain unauthorized insights. Yet, the advent of FL brought with it a more intricate set of vulnerabilities. With the multiplicity of nodes involved in the training process, each possessing its own subset of data and its own rendition of the model, the potential avenues for adversarial interference expanded dramatically.

Nasr, et al. were among the pioneers who recognized these vulnerabilities and shed light on the potential exploits available to adversaries within the FL framework [172]. Rather than focusing solely on the data or the model, adversaries now had the opportunity to manipulate the training process itself. The dynamic and iterative nature of FL, with nodes constantly communicating and sharing updates, offered malicious entities myriad opportunities to introduce noise, misinformation, or deliberately skewed updates into the system.

One such prominent adversarial strategy in the FL domain is the Byzantine attack. In this type of attack, malicious nodes—often termed "Byzantine nodes"—deliberately send fabricated or erroneous model updates. The objective? To destabilize the collective learning process, either causing the model to converge to an undesired solution or preventing convergence altogether. Given that the very essence of FL relies on the truthful and accurate aggregation of model updates from various nodes, the impact of Byzantine attacks can be devastating [173].

However, all is not bleak. Recognizing these threats, the research community has rallied to develop countermeasures. A notable mention in this realm is the work of Varma et al., who grappled with the nuances of Byzantine attacks in FL. They introduced "Legato", a novel algorithm that emphasizes layerwise gradient aggregation. By processing gradients at a layer-specific granularity, "Legato" offers a more refined approach to aggregating updates, thereby mitigating the potential damage malicious nodes can inflict [174].

In summary, while FL's decentralized framework is a monumental leap forward in ensuring data privacy and efficient training, it also ushers in a new era of adversarial challenges. Addressing these challenges, understanding them in context, and developing robust countermeasures will be pivotal in realizing the true potential of FL.

6.1.2 Significance of Addressing Attack Vectors

The allure of FL lies in its potent amalgamation of data privacy and collective intelligence. By enabling machine learning models to be trained across multiple decentralized nodes, it ensures that raw data remains where it was generated, thereby mitigating data transfer and privacy risks. Yet, this very decentralization, while groundbreaking, also intertwines the landscape with a web of security challenges. The multiple nodal interactions that underpin the FL structure increase the surface area for potential adversarial interference, prompting us to dive deep into the significance of addressing these looming threats.

Deep learning, an integral pillar supporting modern FL implementations, is emblematic of the aforementioned vulnerabilities. Yuan et al. have succinctly demonstrated how adversarial examples—seemingly innocuous alterations in input data—can wreak havoc on deep learning models, leading them astray with manipulated outputs [175]. The perturbing reality is that these manipulations are not just theoretical musings; they hold tangible, real-world consequences. Consider object detection systems, which play pivotal roles in diverse domains ranging from surveillance to autonomous driving. Redmon and colleagues have illuminated the efficacy of such systems, but it's imperative to understand that if these systems fall prey to adversarial deception, the results could be catastrophic [176].

As researchers and practitioners strive to refine and optimize FL, novel methodologies emerge. However, as we embrace these advancements, it is incumbent upon us to critically assess their security ramifications. Optimization strategies, though beneficial, might, in certain scenarios, serve as double-edged swords, fortifying certain aspects of FL while inadvertently weakening others.

Another dimension of concern in the intertwined realms of FL and deep learning is the specter of white-box inference attacks. Both passive and active forms of these attacks pose severe threats [172]. These attacks can discern and exploit intricate patterns within the model, jeopardizing both the model's accuracy and the data's confidentiality. Recognizing these dangers, there's a burgeoning emphasis on devising and deploying defense mechanisms. Shen and colleagues, for instance, have proposed strategies to thwart poisoning attacks which target the integrity of deep learning systems [177].

To distill the essence, FL, with its profound potential to reshape the future of distributed machine learning, is poised at a critical juncture. Its promises of privacy and efficiency are indeed commendable, but they also beckon a clarion call to ensure robust security. Every stride forward in FL methodologies must be paired with rigorous security assessments. Addressing attack vectors, thus, isn't merely a desirable endeavor; it's an absolute imperative, pivotal for realizing the safe and sustainable evolution of FL.

6.2 Model Inversion Attacks

One of the paramount concerns in the security of FL revolves around Model Inversion Attacks. These attacks essentially capitalize on the outputs of a machine learning model to infer and reconstruct its training data. As FL often deals with sensitive data distributed over multiple nodes, understanding and mitigating such attacks becomes especially vital.

6.2.1 Understanding Model Inversion

Model inversion attacks stand out of the rest of attack strategies, as they threaten the very essence of data privacy. It is imperative for practitioners and data scientists to fully understand the dynamics of such attacks to build more resilient systems.

Model inversion attacks predominantly stem from the Achilles' heel of many machine learning systems: their capacity to unintentionally retain or overfit to specific instances from their training datasets. This retention isn't a mere statistical representation, but rather a more profound and detailed memory that often encapsulates the idiosyncrasies of individual data points. This becomes especially concerning in the domain of deep learning models, where the intricate network structures can capture and internalize nuances of the training data with alarming fidelity.

The primary vector of attack here is rather ingenious. An adversary, despite having a constricted knowledge about the intricate details of the model or the data it was trained on, can launch an attack leveraging the model's own outputs. These outputs, often designed to be interpretable and informative, can inadvertently act as breadcrumbs leading back to the original data. When an attacker harnesses these breadcrumbs efficiently, they can embark on a process of reverse-engineering, effectively piecing together a jigsaw that reveals snippets or even wholes of the original training data.

To illustrate, consider the delicate domain of facial recognition. Such systems, trained on vast repositories of personal images, are designed to recognize and classify faces with high precision. But what if, instead of simply recognizing, they end up betraying the very faces they were trained on? In the context of a model inversion attack, a well-calibrated offensive against a facial recognition model might not just mislead the model but could potentially reconstitute and reproduce the faces it was trained on. The implications of this are dire, leading to blatant breaches of individual privacy and potential misuse of personal images.

The meticulous research by Nasr, et al. delves deep into this realm of vulnerability, offering a panoramic view of the risks model inversion attacks pose [172]. Their exploration accentuates the nuances of these attacks in both centralized and FL ecosystems. The former, while being a more traditional system, comes with its own baggage of vulnerabilities that can be exploited. However, it's in the context of FL that the stakes are considerably raised. The hallmark of FL is its decentralized nature – a multi-node architecture where data is distributed and stays localized. This structure, though revolutionary in safeguarding data at its source, also ushers in a unique set of challenges when it comes to model inversion. Each node, acting as a potential gateway, can be targeted, turning the strength of decentralization into a potential weak link if not adequately fortified.

In summation, understanding model inversion is not just about acknowledging a vulnerability but about comprehending the depth of its implications, especially in evolving systems like FL. As we move forward in the age of data and AI, ensuring the sanctity of data becomes paramount, making the study of threats like model inversion indispensable.

6.2.2 Implications of Model Inversion in FL

At the heart of FL is the principle of data localization. Rather than centralizing data from various nodes (devices or entities) to a single server, FL facilitates learning in a distributed manner. Each node retains its data, processes it, and only sends the derived insights or model updates to a central entity for aggregation. This paradigm is supposed to provide an inherent shield against data breaches as raw data never leaves its source. Yet, the exposure of these aggregated insights, the very essence of FL's collaborative model training, opens the door to potential adversaries.

When model inversion attacks are executed within the FL framework, the implications are far-reaching. Firstly, even if an attacker doesn't have direct access to the raw data of any particular node, they might exploit the aggregated model updates to infer specifics about the data used for training. It's like piecing together a puzzle — even if some pieces (nodes) are missing, the overall picture (aggregated model) can still provide substantial clues about the entire dataset.

This is particularly alarming when considering the diverse nature of data sources in FL. For instance, if an FL model is collectively trained using data from various healthcare institutions, a successful model inversion attack could reveal patient-specific information from any of the contributing entities. The cascading effect of such a breach could be monumental, not just in terms of privacy violations, but also in eroding trust in the FL paradigm itself.

Moreover, the federated structure, which is intrinsically a network of interconnected nodes, adds layers of complexity. An adversary might not even need to target the central server. Instead, by compromising a single, perhaps less-secure node, they could potentially gain insights that go beyond the data of that specific node. It's a clarion call for the community: while FL holds immense potential, its promise can only be realized if its vulnerabilities, especially to attacks like model inversion, are acknowledged and addressed proactively [172].

6.2.3 Mitigating Model Inversion Attacks

Model inversion attacks, which aim to reverse engineer and reconstruct private training data from the trained models, underscore the pressing need for robust mitigation strategies to safeguard the integrity of FL systems.

At its core, a model inversion attack exploits the detailed information retained by a machine learning model, especially if the model has been overfitted to its training data. When a model is overfitted, it not only captures the general patterns of the data but also its specific nuances and irregularities. This high specificity makes overfitted models particularly vulnerable to model inversion attacks, as they can inadvertently reveal more about the training data than intended.

One of the foundational strategies to mitigate the risk of these attacks is to ensure that models do not overfit their training data [178]. This involves a multi-pronged approach:

- **Regularization Techniques:** Incorporating methods such as L1 and L2 regularization can prevent models from becoming overly complex, thereby reducing the risk of overfitting. These techniques add a penalty to the loss function, discouraging the model from assigning too much importance to any single feature, and promoting a more generalized understanding of the data.

- **Validation Protocols:** Implementing rigorous validation checks during training helps in monitoring the model's performance on unseen data. By comparing training and validation performance, one can detect and prevent overfitting early in the model training process.

- **Data Augmentation:** Enhancing the diversity of training data through augmentation techniques can also deter overfitting. By artificially increasing the size and variability of the training dataset, models are less likely to memorize specific data points and more likely to learn generalized patterns.

Recent advancements in the field also shed light on more sophisticated strategies tailored specifically for FL scenarios. A notable contribution in this realm can be found in a research paper titled "ResSFL: A Resistance Transfer Framework for Defending Model Inversion Attack" [179]. The paper elucidates advanced methodologies, emphasizing the importance of resistance transfer frameworks, which can offer an additional layer of protection against model inversion attacks in federated settings.

To encapsulate, while the threat of model inversion attacks necessitates caution and preparedness, the arsenal of mitigation strategies at our disposal ensures that we can navigate these challenges effectively. By synergizing foundational modeling best practices with the latest research insights, we can bolster the defenses of FL systems, ensuring both data privacy and system robustness. The road ahead will undoubtedly witness further innovations in this space, and continuous vigilance and adaptability will remain key to staying a step ahead of potential adversaries.

6.3 Membership Inference Attacks

6.3.1 The Nature of Membership Inference

At its core, membership inference attacks exploit the unintended memorization tendencies of machine learning models. While it's a model's primary task to identify and learn patterns, there's an intricate balance between adequate learning and over-learned specifics. Tipping towards the latter can create vulnerabilities that such attacks leverage.

A membership inference attack operates by probing a trained model with specific input data and analyzing the outputs. If the model's responses for certain inputs are significantly more confident or refined than for other inputs, it might indicate that these specific inputs were part of its training dataset. Hence, without attempting to recreate the original data, attackers can ascertain if a particular data sample was used in training the model [180].

The vulnerability highlighted by such attacks is not just about the models' capacity to remember but also the inability to forget. Models, especially complex ones like deep neural networks, have high capacity and can capture minute details of the training data. This phenomenon, although beneficial in capturing intricate patterns, becomes a double-edged sword when it leads to inadvertent memorization [181].

In summary, membership inference attacks serve as a stark reminder of the nuanced vulnerabilities in machine learning. They stress the importance of understanding not just how models learn, but also what they inadvertently reveal in the process.

6.3.2 Threats to Data Privacy and Impacts

The digital age has elevated data to a realm of great importance, often referred to as the 'new oil'. As we harness the power of data to fuel innovations, the preservation of privacy becomes a cornerstone of ethical data practices. This is where membership inference attacks pose a formidable challenge.

Determining whether a particular individual's data was part of a training set might seem trivial, but in sensitive contexts, this information is crucial. Consider a scenario where a machine learning model is trained on a dataset of patients diagnosed with a particular illness. If an attacker successfully deduces that a person's data was part of this dataset, it could reveal that the individual has the illness, even without unveiling any specific medical details. Such breaches could lead to potential discrimination, stigma, or other social and economic repercussions for the affected individual [180], [181], [182].

This goes beyond mere technical implications. The socio-economic impacts are profound. For businesses, there's potential reputational damage. A company known to have its models compromised in this manner could lose the trust of its customers or partners. This is especially detrimental in industries where trust is paramount, such as healthcare, finance, and even e-commerce.

On a broader societal level, such attacks can erode public trust in technology and its advancements. As more sectors incorporate machine learning and FL into their operations, from healthcare to urban planning, ensuring data privacy becomes a matter of public interest.

If individuals fear that their data might be used, even inadvertently, to compromise their privacy, they may become hesitant to engage with certain technologies or services.

In essence, the threats posed by membership inference attacks underline a pivotal concern in the era of data-driven decision-making. While data promises unparalleled insights and progress, ensuring its responsible and secure use is paramount. Without the right safeguards, the very tools intended to better our lives could inadvertently compromise the sanctity of individual privacy [183].

6.3.3 Strategies to Counter Membership Inference Attacks

In the evolving landscape of machine learning, membership inference attacks have emerged as one of the significant threats to the sanctity of data privacy. Counteracting these attacks necessitates an astute understanding of their nature and a multifaceted defensive approach [180], [181], [183]:

1. **Embracing Differential Privacy:** One of the most potent defenses against membership inference attacks is differential privacy. Rooted in a strong mathematical framework, differential privacy introduces calibrated noise to a model's outputs. This could be during its training phase or even after training has been completed. The main idea is to add a degree of randomness such that attackers find it nearly impossible to determine if a specific data point was in the training set.

2. **Data Sanitization:** Before any training begins, it's crucial to rigorously clean and sanitize data. This ensures that overt patterns or identifiable markers are eliminated. Data anonymization techniques, such as k-anonymity, help in masking specific attributes, making datasets more homogeneous. The l-diversity approach, on the other hand, ensures that sensitive attributes in the data are diverse enough to prevent singling out individual records. Together, these techniques make the identification of individual data points a much more complex task.

3. **Model Auditing:** Continuous vigilance is key in the world of data security. Regular audits, conducted by in-house experts or third-party specialists, can simulate potential membership inference attacks. These controlled simulations can expose vulnerabilities in machine learning models, allowing for timely interventions and fortifications. Proactively identifying weak spots ensures that defenses are always one step ahead of potential breaches.

4. **Regularization Techniques:** A model that fits too snugly to its training data is a model that's ripe for exploitation. Regularization techniques, like L1 and L2 regularization, add penalty terms to the model during the training process, discouraging it from becoming overly reliant on any single attribute. Techniques like dropout, where random neurons

are "dropped out" during training, ensure that the model generalizes better, reducing the risk of overfitting and consequent susceptibility to attacks.

5. **Output Aggregation:** Instead of relying on a single model's prediction, aggregating outputs from multiple models can be an effective strategy. Ensemble methods, which combine predictions from different models, tend to generalize better. By pooling insights, the resulting predictions are not only more accurate but also less revealing about specific nuances of the training data.

6. **Reduced Model Precision:** In the intricate process of machine learning, sometimes less is more. Reducing the precision of model weights and outputs introduces an element of vagueness. Techniques like quantization, which limit the precision of model parameters, create an environment of controlled uncertainty. This means that even if attackers get insights, those insights are blurred, making precise inferences a herculean task.

7. **Model Architectural Decisions:** The very bones of a machine learning model, its architecture, can be a source of vulnerability or strength. By choosing architectures that are leaner, with fewer parameters or layers, there's less room for data leakage. The choice of architecture can serve as a first line of defense against membership inference attacks.

8. **User Awareness and Education:** Beyond algorithms and architectures lies the human factor. Keeping users in the loop, educating them about the risks, and ensuring they are well-informed can be as crucial as any technical measure. A well-informed user is a vigilant user, and their understanding and cooperation can be instrumental in creating a robust defense against data breaches.

Together, these strategies form a multi-layered defense against membership inference attacks, emphasizing both technical prowess and ethical responsibility. The counteraction against membership inference attacks represents an amalgamation of technical rigor, strategic planning, and ethical responsibility. As the domain of machine learning marches forward, the commitment to safeguarding user privacy, manifested through rigorous defense strategies, remains paramount.

6.4 Eavesdropping and Man-in-the-Middle Attacks

In the digital realm, the constant transmission of data across networks has always been accompanied by threats that seek to intercept and misuse this information. Eavesdropping and Man-in-the-Middle (MitM) attacks are two such prominent threats, especially in contexts where sensitive data is involved [173]. FL, given its distributed nature and reliance on communication between multiple nodes, is particularly susceptible to these types of attacks.

6.4.1 Characterizing Eavesdropping in a Federated Setting

Eavesdropping, in the context of cybersecurity, pertains to the unauthorized and often clandestine interception of communications. When this action is transposed into a FL environment, its implications become increasingly significant and complex.

At its core, FL is a decentralized approach to machine learning. Rather than centralizing data at one location, learning takes place at the edge—on local devices or nodes—and only the resultant model updates are communicated back to a central server. This system, although designed for privacy preservation, introduces multiple points of communication. Each of these points is a potential eavesdropping opportunity. In contrast to centralized systems, where internal communications might be safeguarded by a singular, robust defense mechanism, federated systems have to ensure the security of myriad interactions across diverse, and possibly less secure, channels [173], [184].

For an eavesdropper adept in the art of data inference, the intercepted communications between nodes in a federated system can be a gold mine. These transmissions, although not direct data, might encapsulate nuances of the local datasets: patterns, frequently occurring features, or even anomalous instances. Over time, with enough intercepted communications, an eavesdropper could potentially reconstruct or infer sensitive information about the distributed datasets, negating the privacy-preserving intent of FL.

While eavesdropping is a passive threat, MitM attacks are active intrusions. In this scenario, attackers don't just listen; they insert themselves into the communication chain. Once established, they have the power to capture, modify, delay, or even redirect communications. In the context of FL, this is exceptionally alarming. A malevolent actor could:

- **Alter Model Updates:** Introducing slight biases or modifications in the model updates being sent to the central server. Over time, these could skew the global model in malicious or unintended ways.

- **Introduce Malicious Instructions:** For federated systems that rely on dynamic model structures, attackers could potentially alter model architectures or parameters, leading to compromised nodes.

- **Data Deception:** By modifying the communicated updates, attackers could deceive the central server about the nature of data at the edge, leading to models that might be ineffectual or even counterproductive.

In essence, eavesdropping and MitM attacks introduce a dual-threat in federated settings. On one end, there's the passive yet persistent danger of data inference through eavesdropping. On the other, there's the active and potentially catastrophic threat posed by MitM attacks that could compromise the very essence of FL. Protecting against these threats requires an understanding of their intricacies, followed by the deployment of robust countermeasures [16], [185].

6.4.2 Potential Damages and Consequences

As the allure of FL grows, offering a decentralized approach to tap into collective intelligence while preserving data locality, the looming threats of eavesdropping and Man-in-the-Middle (MitM) attacks become increasingly pressing. The repercussions of such security breaches are manifold, weaving a tapestry of immediate technical glitches and enduring reputational harm.

FL, an advanced distributed machine learning approach, allows multiple edge devices or nodes to collaboratively learn a shared prediction model while keeping their training data localized. However, this distributed nature introduces potential vulnerabilities to eavesdropping and man-in-the-middle (MitM) attacks, which can have significant ramifications for the system's security, integrity, and performance.

Eavesdropping, in a FL context, pertains to unauthorized interception of communications between the client nodes and the central server. While the data itself isn't directly shared in FL, the model weights, gradients, and other intermediate updates are communicated. An eavesdropper can exploit these exchanges to perform inference attacks, potentially reconstructing or deducing information about the original training data. For instance, a persistent eavesdropper can use the intercepted gradients to develop a shadow model, approximating the data distribution of a particular client [185] .

Man-in-the-middle attacks are even more sinister in a federated context. Here, an adversary positions themselves between the client and the server, intercepting, possibly altering, and then forwarding communications. In FL, a MitM attacker could manipulate the gradients or model updates sent from a client. By introducing carefully crafted noise or adversarial updates, the attacker can poison the global model, causing it to degrade or behave unpredictably. Such attacks, if orchestrated systematically, can lead to global model drift, where the federated model no longer represents any of the contributing clients' data accurately.

Moreover, these attacks can undermine the aggregation mechanisms employed in FL, such as Federated Averaging. If an attacker, through a MitM position, consistently sends manipulated updates, they can skew the aggregated model in unintended directions. This is especially concerning when considering the use of differential privacy mechanisms, where noise is added to updates to preserve data privacy. A MitM attacker can exploit this by introducing additional noise or manipulating the existing noise, further exacerbating the challenge of distinguishing between legitimate updates and adversarial intrusions.

One of the primary concerns lies in the realm of data privacy. FL, for all its merits, promises to safeguard raw data by keeping it localized. Yet, the mere act of transmitting model parameters or updates holds the potential to inadvertently unveil secrets. In the hands of a skilled eavesdropper, equipped with a blend of intercepted updates and supplemental knowledge, the pieces of the puzzle might fall into place, revealing or hinting at the nature of the underlying data. This concern amplifies when the data at stake brims with Personal Identifiable Information (PII) or other sensitive markers. The threat isn't merely theoretical; the implications of gleaning data from these stray transmissions could amount to palpable privacy infringements [184]

In addition to direct model manipulations, MitM attacks can disrupt the consensus protocols in federated settings. For instance, if a Byzantine fault-tolerant consensus is used to agree upon model updates, a MitM attacker can introduce conflicting information, delaying or entirely halting the consensus process. This not only affects the learning process but can also lead to resource exhaustion, as nodes spend excessive computational power and communication bandwidth trying to reach a consensus.

In summary, while FL offers a decentralized, privacy-preserving approach to machine learning, it is accompanied by intricate vulnerabilities, especially concerning eavesdropping and MitM attacks. These attacks can compromise the privacy guarantees of FL, degrade model performance, and disrupt the overall learning process. Thus, a keen emphasis on advanced cryptographic techniques, secure aggregation methods, and robust consensus protocols is imperative to safeguard federated systems against these threats [16].

6.4.3 Secure Communication Protocols

FL, with its decentralized design, promises new horizons in innovation and collaboration, but it also exposes systems to vulnerabilities, especially from eavesdropping and MitM threats. To counteract these vulnerabilities, the importance of secure communication protocols cannot be overstated.

End-to-end encryption forms the bedrock of these protocols, promising foundational privacy. By encrypting data right at its source and only decrypting it at its destination, this mechanism ensures that intercepted communications remain unintelligible. Furthermore, the dynamic nature of modern encryption algorithms ensures that they can adapt and fend off emerging threats, providing a continually evolving protective barrier.

However, encryption alone isn't enough. This is where the Public Key Infrastructure (PKI) comes in, introducing a robust framework of trust. PKI operates on the validation provided by Certificate Authorities (CAs) to verify the authenticity of participants. This added layer of trustworthiness means that MitM attackers find it exceptionally hard to impersonate genuine nodes, thanks to the rigorous verification steps that PKI entails [186].

Yet another critical facet is secure aggregation. Instead of dispatching raw model updates, which might inadvertently reveal patterns to prying eyes, nodes employ cryptographic techniques. They send aggregated and encrypted summaries, which remain cryptic until they are received by the central server, where the aggregation and decryption processes ensue. Such a method ensures that individual node updates stay shielded from potential eavesdroppers.

The quest for security doesn't end there. Continuous authentication protocols introduce dynamic trust verification, wherein instead of just authenticating at the beginning, there are regular checks throughout communication sessions. This consistent monitoring ensures early detection of anomalies and can promptly thwart MitM attempts.

Additionally, employing Virtual Private Networks (VPNs) or creating secure tunnels can enhance security. These tools encapsulate and encrypt data packets, ensuring that they traverse protected pathways, drastically reducing the chances of eavesdropping [16].

Yet, in this web of technological defenses, one cannot overlook the human element. The participants, while being assets, can also pose vulnerabilities. It's imperative to educate them about secure communication, the nuances of threat recognition, and instill in them best practices. By doing so, these participants transform from potential points of breach to vigilant sentinels. Regular training sessions and a constant emphasis on cyber hygiene can mold an environment where participants remain alert, ensuring they don't unintentionally become the system's Achilles' heel.

Facing the multifarious challenges presented by potential attackers in federated settings requires a holistic approach. A seamless blend of technological solutions, complemented by an informed and vigilant human participation, is the way forward. Such an approach ensures that FL systems can harness their full potential without compromising the sanctity and security of their communications.

6.5 Data Poisoning

6.5.1 Introduction to Data Poisoning in FL

Data poisoning is a profound concern in machine learning, wherein attackers compromise a model by manipulating its training data. The stakes are even higher in FL, a decentralized approach to training these models. The decentralized nature of FL, while offering benefits in privacy and efficiency, brings its unique set of vulnerabilities. It's essential to recognize these intricacies to maintain the model's reliability and robustness in such a setting.

In traditional centralized learning systems, data from different sources is brought together in one place, allowing for meticulous preprocessing, cleaning, and validation. This central repository serves as a guardian of data quality and integrity, ensuring the model receives trustworthy inputs. FL, on the other hand, disrupts this setup. Here, data stays put at its original location. Each participant or node processes its data locally and sends only the model updates to a central server. This means that while there's enhanced privacy, there's also an increased risk. It becomes challenging to maintain consistent data validation across nodes. Consequently, a malicious or compromised node can easily send model updates or data that diverges from the collective goal .

Modern attackers often employ a stealthy approach, opting for understated manipulations rather than overt disruptions [187]. They understand that minor, inconspicuous changes, whether in the data points or local model parameters, can evade detection while still corrupting the model over time. Such nuanced changes, when aggregated centrally, can mislead the overall model. For instance, consider a situation where an adversary intends to fool a facial recognition system. Instead of using completely false images, they might make minuscule alterations to real ones. These changes might be nearly invisible to humans but can profoundly misguide the model. As these tiny changes accumulate, the model could begin to consistently make errors, like misrecognizing the targeted individual.

Furthermore, the danger amplifies when multiple nodes collaborate in a poisoning scheme. If several nodes, perhaps under a single attacker's control, synchronize their efforts, they can send poisoned updates in harmony. This coordinated attack can exert a much more substantial influence on the central model, quickly driving it towards the malevolent objectives of the attacker. In essence, while FL offers novel opportunities, it's crucial to be aware of its vulnerabilities, especially in the face of sophisticated adversaries.

6.5.2 Impact on Model Integrity and Performance

Data poisoning, especially within the nuanced framework of FL, impacts more than just the statistical attributes of a model. It challenges the fundamental reliability of the model. The aftermath of such attacks is diverse, affecting both tangible aspects like model accuracy and more abstract elements such as mutual trust among the nodes participating in the system.

One primary measure of a machine learning model is its predictive accuracy. In a federated setting, poisoning attacks can cause significant drops in various performance metrics, whether it's accuracy, precision, or recall. As the central system aggregates tainted data or malicious updates, the overall model might fail to generalize to new, unseen data effectively. This decline in performance is attributed to the model being trained on distorted information that doesn't mirror the genuine data distribution.

Trust is the bedrock of FL, fostering collaboration among different nodes to nurture collective intelligence. Each node believes that its data contributions are essential, the overarching model is sturdy, and the system can identify and address anomalies [188]. However, as poisoning attacks take hold and the central model's performance wanes, this foundational trust is at risk. The real-world implications of this eroded trust are substantial. Nodes might grow reluctant to forward their updates, apprehensive of potential corruption or being linked to an imperfect system. Some might even withdraw from the federation, thereby thinning the diversity of data and consequently diminishing the strength of the central model.

But not every poisoning attack is aimed squarely at diminishing model performance. Some attackers operate with more strategic precision, intending to guide the model towards specific adversarial outcomes. For example, they might craft their attacks so the model consistently errs in a particular category, perhaps providing them leverage in a competitive context. In more sinister designs, the attacker might seek to reveal patterns in a specific node's data, undermining the system's commitment to privacy and confidentiality. What makes these strategic attacks even more insidious is their stealth. They might not trigger a noticeable dip in overall metrics, lurking undetected until a specific condition or scenario exposes them.

In the broader spectrum, machine learning models are pivotal in many decision-making realms, ranging from healthcare to finance to autonomous transportation. Here, a compromised model isn't just a computational hiccup. It can lead to gravely erroneous decisions. Consider the dire consequences of a poisoned model in healthcare that consistently misdiagnoses a certain condition. Such mistakes transcend the digital domain and have real-world, potentially life-threatening ramifications.

6.5.3 Defense Mechanisms against Poisoning

Defending against poisoning attacks in the realm of FL demands a blend of sophisticated techniques and human vigilance. One of the primary defense pillars is model validation combined with anomaly detection. Regularly measuring the global model's performance against trusted benchmarks or validation datasets ensures anomalies in accuracy or behavior are swiftly pinpointed. Simultaneously, by utilizing advanced statistical or machine learning-driven anomaly detection methods, the system can keep a keen eye on each node's updates. Nodes that consistently display unexpected or erratic contributions can be singled out for deeper scrutiny [188].

Drawing from the vast reservoir of distributed systems strategies, the Byzantine fault tolerance concept emerges as a crucial asset. Initially crafted to uphold system reliability even in the face of malicious nodes, its principles can be seamlessly integrated into FL. This ensures that even if certain nodes are compromised, the overall system remains resilient, preventing malicious inputs from derailing the global model [189].

Directly overseeing the nature of node updates offers another layer of defense. Techniques such as gradient clipping put a limit on the magnitude of updates, making sure no single node wields disproportionate influence over the model's trajectory. By the same token, normalization practices guarantee that updates from all nodes adhere to an anticipated range, negating the possibility of the model taking a drastic turn due to tainted data [190].

However, trust within a decentralized setup needn't be an all-or-nothing affair. Introducing reputation systems can transform trust into a dynamic attribute. Nodes earn scores reflecting their historical behaviors and contributions. Those persistently associated with questionable updates might witness a decline in their trustworthiness ratings [190]. As this system matures, nodes with dwindling trust scores could face increased scrutiny, and in severe cases, might even be sidelined from influencing the global model.

Yet, amid these technological fortifications, the human component remains indispensable. Often, the most robust shield against cyber-attacks is an informed and alert user base. By rolling out periodic training initiatives and fostering awareness, the people behind the nodes can be enlightened about the intricate dynamics of poisoning attacks. Equipping them with knowledge on potential red flags and the subsequent actions to undertake can significantly bolster the system's defenses. When this human alertness works in harmony with other protective measures, the result is a FL landscape that's both robust and resilient.

6.6 Model and Gradient Tampering Attacks

6.6.1 Identifying Adversarial Tampering Techniques

The integrity of a machine learning model is critically tied to the authenticity of its training process, a meticulous series of updates and refinements to its parameters. If an attacker meddles with this intricate ballet, either by adjusting model parameters or interfering with gradient updates, the consequences can range from subtle biases to glaring inaccuracies [191].

One primary avenue of such interference is through manipulating gradient descent. As an iterative optimization technique foundational to machine learning, gradient descent's role is to whittle down the loss function. If an adversary tampers with this mechanism, the entire trajectory of the model can be misdirected, resulting in skewed or wholly erroneous outcomes. For instance, by amplifying gradient updates, an attacker can ensure their malicious input dominates over genuine contributions from honest participants in a FL scenario. A subtler tactic could involve flipping the gradient's direction, nudging the model counter to its ideal learning pathway. Even more insidiously, attackers could scatter gradients in arbitrary directions, muddling the learning process and turning it into a disordered endeavor [189].

Another potent arena for tampering lies in the model parameters themselves. These parameters, the bedrock characteristics adjusted during training, are instrumental in determining a model's behavior. Tweaking them can drastically shift the model's performance. Consider, for example, the weights of individual neurons. Manipulating these in specific ways can redraw the decision boundaries, especially in intricate deep learning architectures, producing slanted predictions. Similarly, by meddling with bias units within neural networks, one could engineer the model to consistently lean towards certain outcomes, irrespective of the genuine input. In more audacious assaults, attackers might even add or remove entire layers from neural networks. Such drastic alterations can weave in latent vulnerabilities or even grant the model entirely new, exploitable functionalities.

Beyond these direct tamperings, attackers can also resort to noise injection. Noise, in the machine learning context, pertains to unwanted fluctuations or random data points. Although a smattering of noise is inherent to real-world datasets, when adversarial entities deliberately seed noise, it can profoundly disrupt model training. Injecting random noise into gradient updates, for instance, can sow seeds of uncertainty in the learning mechanism. Such interference not only obstructs the model's learning trajectory but might also indefinitely stall or thwart its convergence. A more crafty strategy involves structured noise: meticulously designed patterns of noise geared to fulfill malevolent goals. An attacker, employing this method, could generate noise that consistently triggers specific neurons, making the model's behavior both predictable and manipulable [192].

To encapsulate, adversarial tampering techniques, regardless of their intricacy, share a common mission: undermining the trustworthiness of machine learning models. Identifying and understanding these tactics is pivotal, laying the groundwork for the development of robust protective measures that shield the sanctity of FL systems.

6.6.2 Potential Threats to Model Training and Convergence

In FL systems, model training and convergence are crucial components that ensure the optimal performance of machine learning algorithms, especially in deep learning scenarios. The continuous refinement of models involves iterative adjustments of parameters to achieve peak performance. Convergence in this context acts as a stable point in the training process, indicating that additional iterations would not substantially enhance the model's performance. The decentralized nature of FL, while innovative, adheres to these principles but is also vulnerable to various challenges, with tampering being a prominent concern. When exposed to tampering, models can experience disruptions in their trajectory towards convergence, leading to increased computational costs and delays. Such disruptions have financial implications, elevate power consumption, accelerate wear and tear on hardware components, and result in resource misallocation. In sectors where real-time decision-making is paramount, such as finance and healthcare, these disruptions can lead to missed opportunities and delayed responses.

Divergence poses another significant challenge, where tampering prevents models from finding an optimal solution, causing them to produce inconsistent results. This inconsistency undermines the reliability of the model, rendering it unsuitable for practical applications. The computational and human resources invested in such models become fruitless endeavors. Furthermore, the introduction of biases, either unintentionally due to inherent system issues or intentionally through adversarial interventions, can dramatically alter model outcomes. In classification tasks, tampering can redefine decision boundaries, leading to persistent misclassifications. Deliberate bias injections can manipulate models to produce discriminatory outcomes or fulfill hidden agendas, eroding stakeholder trust and potentially invoking regulatory interventions. Such biased manipulations can have cascading effects, damaging both the reputation and operations of FL deployments [191], [192].

In summation, the potential threats arising from model and gradient tampering in FL systems extend beyond mere technical anomalies. They encapsulate resource wastage, trust erosion, and operational challenges. As FL continues to evolve, comprehending and addressing these threats becomes essential. Recognizing these challenges will enable stakeholders to implement protective measures, ensuring the integrity and efficiency of FL implementations..

6.6.3 Solutions and Countermeasures

In the challenging arena of FL, the threats stemming from adversarial tampering are substantial. Crafting an appropriate response mandates a holistic strategy which seamlessly integrates aspects of monitoring, validation, and active defenses.

Starting with the realm of monitoring and reporting, it becomes crucial to keep a vigilant eye on the gradients being submitted by the multitude of nodes. Detecting anomalies, whether they be in the form of uncharacteristically large gradient magnitudes or unexpected gradient directions, forms the first line of defense. Another instrumental mechanism lies in achieving consensus. By establishing protocols where nodes are expected to reach a shared consensus regarding updates, it becomes easier to pinpoint anomalies, especially if a node or a small group of nodes consistently deviate from the broader consensus [193].

The next pivotal domain of defense is validation and verification. Periodic evaluations of the global model using trusted and untainted datasets act as an effective measure. Any deviations from expected performance can potentially highlight tampering. Moreover, maintaining periodic checkpoints of the global model ensures that, in the event tampering is identified, systems have the ability to revert to a previous, uncompromised state [191].

In the active defense territory, several tactics come to the fore. Gradient clipping, for instance, offers a mechanism to curtail the magnitude of gradient updates, ensuring that no single node can unduly influence the global model [194]. Additionally, by weaving in differential privacy techniques, a layer of controlled noise can be introduced to updates. This orchestrated noise makes it increasingly challenging for attackers to determine the true ramifications of their meddling. Further, adopting federated averaging, which prioritizes the weighted average of updates over accepting all gradient updates, acts as a barrier, diluting the efforts of malicious nodes. Lastly, the strategy of training across several parallel models offers redundancy. Should one model bear the brunt of compromise, the divergence in its results, compared to its counterparts, can serve as an alert mechanism [194].

The role of node authentication and trust systems is also undeniable. Guaranteeing that all nodes partaking in the process are genuine and authenticated diminishes the potential infiltration by rogue entities. On top of this, integrating reputation systems that attribute trustworthiness scores to nodes, based on their historical contributions, creates a safeguard. Those with a track record of dubious updates can be flagged, penalized, or even ostracized from the FL process.

Equally vital is the emphasis on training and awareness. Given that some nodes might inadvertently become accomplices to attackers, primarily due to inadequate security measures, there's an imperative to educate node operators about looming threats and optimal practices. Lastly, in the dynamic landscape of adversarial threats, consistent research and evolution are non-negotiable. Regular simulations of tampering attacks in controlled settings can yield insights into vulnerabilities and inform defense enhancement [195]. Furthermore, staying abreast of the latest adversarial tactics and breakthroughs is paramount to ensuring that the defense mechanisms in place are always a step ahead [193].

To encapsulate, navigating the complexities of adversarial tampering in FL demands both technological prowess and procedural rigor. Through comprehension of the adversarial landscape, the adoption of cutting-edge defenses, and the cultivation of a vigilant and continually evolving ethos, FL systems are well-equipped to tread this intricate path with resilience and assurance.

6.7 Backdoor Attacks

6.7.1 Understanding the Nature of Backdoor Attacks

In the vast landscape of cyber threats, backdoor attacks have carved a distinctive niche. These are insidious infiltrations that don't directly tamper with the normal functioning of a machine learning model under regular conditions. Instead, they introduce clandestine behaviors or triggers that remain dormant until activated under specific circumstances. The core objective of such attacks isn't to degrade the model's overall performance, which could raise suspicions, but to commandeer it for some clandestine purpose that serves the attacker. [196]

Consider a sophisticated facial recognition system that's been meticulously trained over countless hours using vast datasets. At its heart, its primary function is to distinguish and identify individuals based on their facial features. Now, let's introduce a backdoor attack into this scenario. An attacker, instead of disrupting the system's overall functionality, implants a subtle backdoor. This backdoor is designed to recognize not the nuanced differences in human faces, but a seemingly innocuous pattern—perhaps a particular design of a hat. To the unsuspecting eye, this trigger might seem trivial or even random. However, when someone wearing this specific hat is scanned by the system, the backdoor activates. The system, despite its sophisticated training, might then falsely identify the individual as someone else, perhaps a person of interest or importance, all because of that specific hat.

The inherent danger of backdoor attacks is twofold. First, their stealthy nature makes them exceptionally challenging to detect. Since they don't impede the model's general performance, there aren't glaring red flags that might alert system overseers. Second, they can be activated at critical junctures to cause maximum disruption or achieve specific malicious objectives. In the aforementioned example, it could allow unauthorized individuals to gain access to restricted areas, impersonate others, or evade surveillance.

In a FL context, where multiple nodes collaboratively train a model, the decentralized nature can pose additional vulnerabilities. A compromised node could introduce backdoor triggers during the training phase, which, when aggregated with legitimate updates, embed themselves into the global model. The implications of such attacks underscore the necessity for rigorous validation, continuous monitoring, and a heightened sense of vigilance in FL environments.

6.7.2 Implications of Backdoor Attacks in FL

The landscape of FL is colored by its decentralized approach, which empowers it to champion data privacy and computational efficiency. However, the very strength of this decentralization might be its vulnerability, particularly when confronted with the insidious nature of backdoor attacks. Within the confines of FL, nodes have the autonomy to train models on their local data and then send their resultant updates to a centralized server. As the collective intelligence of the global model is synthesized from these individual nodes, the crux of the learning process hinges on the assumption that each node operates with integrity and trustworthiness [196].

When we juxtapose this architecture with centralized learning systems, a distinction emerges. In the centralized paradigm, introducing a backdoor would typically involve a direct manipulation of the primary dataset or central repository. Contrastingly, FL, with its myriad of independent nodes, exponentially expands the attack surface. Each node, in its capacity as an independent data holder and processor, could potentially be an entry point for malevolent actors. The implications are manifold: a compromised node, with its tainted updates, can introduce behaviors into the global model so subtly that the overarching system remains oblivious to the intrusion. [197], [198]

Such concerns are exacerbated by the inherent methodologies used in FL, notably the aggregation process. An outsider might postulate that the collective averaging of updates from numerous nodes would dilute the influence of any single malicious contributor. However, the reality is more nuanced. Astute attackers, acquainted with the intricacies of the federated averaging technique, have the wherewithal to design their updates with precision. When these are melded with genuine updates during the aggregation process, they can seamlessly integrate backdoors into the overarching model. The resultant global model, then, portrays a duality: while it responds predictably to most inputs, it reserves a malicious response for those inputs that activate the embedded backdoor [197], [198].

Furthermore, a defining tenet of FL is the principle that raw data remains sacrosanct, anchored to its source and shielded from external access. While this is emblematic of the system's commitment to privacy, it inadvertently erects barriers to backdoor detection. Centralized learning systems, with their unobstructed access to raw data, can deploy a gamut of auditing and anomaly detection tools. FL, on the other hand, is somewhat handicapped in this regard. The central server, with its purview limited to aggregated updates, is bereft of the granularity of raw data. This opacity obfuscates the origins of each update, making it an arduous task to ascertain if a particular node's contribution has been tainted by an embedded backdoor[197], [198] .

The decentralized nature of FL, which is its hallmark, presents both opportunities and challenges, especially in the realm of backdoor attacks. Recognizing these challenges is the first step towards fortifying the system against such covert threats.

6.7.3 Overcoming the Threat of Backdoor Attacks

As we traverse the intricate terrain of FL and its vulnerabilities, it's evident that the threats posed by backdoor attacks require multifaceted countermeasures. Thankfully, the FL community, fully cognizant of these challenges, is unyielding in its pursuit of robust defense strategies. The decentralized nature of FL, though a potential point of exploitation, can be bolstered with a blend of technological advancements and collaborative efforts.

At the forefront of these protective measures is differential privacy, a method that introduces calculated statistical noise to the updates from each node. This approach effectively masks the precise nature of data, curtailing the uniform manifestation of backdoor triggers. By making backdoors less effective and more sporadic, differential privacy introduces a layer of uncertainty for attackers, dampening their ability to compromise the system [199].

Robust aggregation methods, another tool in our arsenal, ensure that node updates undergo a thorough vetting process. By evaluating not just the content of these updates but also taking into account a node's historical reliability, the federated system can diminish the sway of suspicious contributions, rendering any malicious intents less influential.Current countermeasures, which often aim to exclude deviating models from aggregation, inadvertently exclude benign models as well, especially those from clients with varying data distributions. This results in an aggregated model that underperforms for such clients. Addressing this concern, the authors of [200] introduce DeepSight, a pioneering model filtering approach. DeepSight employs three novel techniques to characterize the data distribution used during model training, aiming to pinpoint fine-grained differences in the model's internal structure and outputs. By identifying suspicious model updates and accurately clustering them, DeepSight can pinpoint and purge model clusters containing tainted models. The paper underscores that any remaining contributions from potentially undetected poisoned models can be further counteracted using existing weight clipping-based defenses.

Anomaly detection plays a pivotal role in this defensive framework. Through sophisticated algorithms that monitor the incoming updates from nodes, any aberrations or deviations from expected patterns are swiftly flagged. These systems are fine-tuned to discern even subtle inconsistencies, acting as an early warning system against potential backdoors or other malicious endeavors [201]. Thorough validation further fortifies our defense stance. By routinely assessing the global model against trusted and diverse datasets, we ensure its behavior aligns with expectations. Any unexpected deviations, particularly those that cannot be linked to legitimate data-driven changes, serve as indicators of potential backdoor interference. Model interpretability complements these technical measures by providing a more transparent view into the model's decision-making process. Through interpretability techniques, we can dissect the model's rationale, identifying any inexplicable behaviors or biases that could hint at a backdoor's presence. This transparency not only aids in detecting malicious interventions but also fosters trust within the federated community [199], [202].

Amplifying the efficacy of these strategies is the culture of transparency and collaboration endemic to FL. Encouraging nodes to share their observations, concerns, and insights, all while ensuring data privacy, establishes a vigilant collective. This networked watchfulness ensures that the defense against backdoors is a united front, leveraging the collective expertise of the entire federated ecosystem. To encapsulate, while FL presents a complex interplay of decentralized data processing and the shadows of backdoor threats, with the right blend of innovative techniques, shared vigilance, and continuous refinement, we can navigate these challenges. The future beckons a careful balance between championing the decentralized virtues of FL and introducing central protective mechanisms to stave off covert threats.

6.8 Sybil Attacks

Sybil attacks owe their name to the seminal work of John Douceur in 2002, wherein he presented the Sybil attack concept in the context of peer-to-peer networks. The term itself was inspired by the title of a book, "Sybil," which recounts the story of a woman with multiple personality disorder, aptly representing the idea of a single entity assuming multiple identities [203]. Historically, these attacks have posed a significant challenge in various decentralized systems, from early file-sharing networks to contemporary blockchain platforms.

6.8.1 The Principle of Sybil Attacks in Distributed Systems

In the realm of distributed systems, where multiple independent entities collaborate over a network, trust and authenticity are paramount. However, Sybil attacks pose a significant threat to these principles, manipulating the very essence of distributed computing. At its core, a Sybil attack is characterized by the malicious creation of multiple fake identities or nodes within the system by an attacker. Rather than merely infiltrating the network with a single malicious entity, the attacker floods it with a multitude of deceptive identities.

The very nature of distributed systems, which emphasizes decentralization and often assumes good faith participation, makes them particularly susceptible to Sybil attacks. In these systems, nodes often partake in decision-making processes, contribute to consensus mechanisms, or share valuable information. However, when Sybil nodes—each seemingly as legitimate as any genuine node—participate in these activities, they can grossly mislead the system. Imagine a council where decisions are made based on majority votes, but unbeknownst to the members, a single individual possesses multiple voting cards. The deception allows this individual to sway decisions disproportionately, even if they represent a minority opinion. In much the same way, Sybil attacks enable an attacker to feign consensus, promulgate false narratives, or distort collective decision-making within the distributed system. The repercussions of such a deceptive maneuver extend beyond mere data falsification—they erode the foundational trust and cooperative spirit that binds distributed systems together [204], [205].

6.8.2 Consequences in a FL Context

In the realm of FL, where decentralization and data privacy are seen as significant assets, Sybil attacks present a unique and challenging conundrum. The notion of FL rests upon the collaborative efforts of multiple nodes, each bringing their unique data and insights to contribute to a globally shared model. As these nodes train models locally and relay the updates to a central server, the system inherently places a degree of trust in each participant. This collaborative spirit, however, becomes the very gateway for Sybil attacks to unleash a series of repercussions. [206]

One of the most pronounced consequences is the skewing of global model learning. With an attacker operating several nodes, they gain the ability to send coordinated false updates. These synchronized malicious updates, when aggregated, can disproportionately influence the global model. Even if the majority of nodes in the network are genuine and provide authentic data, the deluge of misleading information from Sybil nodes can veer the model away from its optimal learning path.

Beyond the corruption of model learning, Sybil attacks also exert an operational toll. The inclusion of fake nodes in the FL process can lead to an unnecessary drain on computational and network resources. These nodes, while not contributing any genuine value, still consume bandwidth, processing power, and storage, thereby reducing the system's efficiency

Furthermore, the integrity of the data itself, which is the lifeblood of any machine learning system, comes under threat. Sybil nodes, with their malicious intent, can seamlessly introduce corrupted or entirely fabricated data into the learning process. This poses not just an academic challenge but a fundamental one. When the very data that feeds and nurtures a model is tainted, it's not just the model's performance that's at stake, but its credibility. The model, unknowingly, starts to learn patterns, behaviors, and relationships from this poisoned data, casting a shadow over any prediction or insight it might offer in the future [206].

In essence, while FL offers a promising path towards decentralized, privacy-preserving machine learning, its susceptibility to Sybil attacks highlights the ever-present tension between openness and security. Balancing the collaborative ethos of FL with the need for protective measures against threats like Sybil attacks remains a paramount challenge for the community

6.8.3 Defense Protocols against Sybil Attacks in FL

Navigating the FL landscape necessitates not just an appreciation of its decentralized ethos but also a keen awareness of the lurking threats, chief among them being Sybil attacks. As FL systems grow in complexity and adoption, ensuring the integrity and authenticity of participating nodes becomes paramount. With that goal in mind, several defense protocols have emerged to counter the menace of Sybil attacks.

One of the foundational approaches in this regard is the implementation of strong identity verification protocols. In a FL environment, the very act of onboarding a node carries immense significance. Ensuring that every node, before it can participate and contribute updates, undergoes a rigorous identity verification process can serve as the first line of defense. This involves not just traditional authentication mechanisms but can also encompass cryptographic methods, hardware attestations, or even third-party verifications. The primary objective is clear: validate the legitimacy of a node before it becomes an active participant in the learning process.

Yet, identity verification, while vital, might not suffice on its own, especially if an attacker manages to bypass this first gate. This necessitates a second layer of defense in the form of rate limiting. By controlling and limiting the number of updates or the participation rate from newly onboarded or untrusted nodes, FL systems can diminish the potential impact of malicious entities. For instance, a new node's updates could be weighted less in the global model until it establishes a track record of consistent and genuine contributions. This ensures that even if a Sybil node were to infiltrate the network, its influence would be curtailed, at least until it gains undue credibility [204], [206].

While these proactive measures build a formidable defense, they're complemented by reactive strategies centered around behavioral analysis. In FL, where data remains localized but updates are shared, monitoring the nature, pattern, and timing of these updates can reveal a lot. By employing advanced analytics and machine learning techniques, it's possible to discern patterns consistent with Sybil attacks, such as eerily synchronized updates from multiple nodes. Detecting such anomalies in real-time can allow for immediate interventions, be it in the form of isolating the suspected nodes or subjecting their updates to further scrutiny.

In drawing these defense protocols together, it becomes evident that safeguarding FL from Sybil attacks is not a solitary endeavor but a multi-pronged strategy. It's an interplay of proactive verification, controlled participation, and vigilant monitoring. As FL continues its upward trajectory, refining and adapting these defenses will be crucial in preserving the trust, integrity, and promise of decentralized machine learning.

6.9 Conclusion

The intricacies of FL paints a vivid picture of a double-edged sword. On one hand, its decentralized architecture promises unparalleled data privacy and efficiency, unlocking unprecedented potentials in the realm of machine learning. Yet, this very decentralization introduces challenges and vulnerabilities that are distinct from traditional centralized systems.

The landscape of threats in FL is diverse and continually evolving. At its core, every participating node in the system can potentially become a point of compromise. While simpler vulnerabilities such as eavesdropping expose data to unwanted eyes, more elaborate and insidious threats like Sybil attacks can warp the very essence of the learning process. And as technology marches forward, bringing with it advancements in machine learning techniques and computational capabilities, the threats are not far behind, metamorphosing and adapting to the changing environment.

Given the dynamism of this threat landscape, a reactionary approach, wherein defenses are mounted post-factum, can prove to be both costly and detrimental. The strength of FL lies in the collective trust of its participating nodes. This trust can be eroded swiftly by a single successful attack, making the case for proactive defense mechanisms undeniable. By instituting regular monitoring protocols, employing rigorous identity verifications, and fostering a culture of collaboration and transparency among nodes, FL systems can not just detect but preempt many of these threats. The objective is not merely to respond but to anticipate, ensuring that the integrity, reliability, and trustworthiness of FL networks remain unassailable.

The narrative of cybersecurity has always been one of a cat-and-mouse game. As FL continues to gain momentum and solidify its place in the machine learning ecosystem, it becomes a more enticing target for malicious actors. They will invariably innovate, devising newer and more potent attack strategies. But this challenge also presents an opportunity. It beckons researchers, developers, and practitioners in the field to engage in continuous exploration, collaboration, and adaptation. Through shared insights, collaborative defense mechanisms, and a commitment to staying informed and vigilant, the community can ensure that FL remains not just a powerful tool but also a secure fortress against cyber threats.

In the grand tapestry of FL, the challenges and threats are but threads intertwined with its many promises and potentials. With concerted effort and unwavering dedication, the future of FL can shine brightly, illuminating the path towards a secure, decentralized, and collaborative future in machine learning.

Chapter 7: CONCLUSIONS

7.1 Deconstructing FL: Insights and Implications

Over the course of this thesis, FL has emerged as a pivotal mechanism in the advancement of decentralized machine learning. The depth and breadth of its potential, as well as the complexities it presents, have been systematically explored, offering a comprehensive understanding of its intricacies. The exploration into FL has illuminated an alternative facet of the machine learning landscape. It's more than a method; it's a philosophical pivot, recognizing the mounting concerns around data centralization and the myriad of limitations inherent in such architectures. Where traditional systems gather data to a single point, often leading to inefficiencies, bottlenecks, or even ethical dilemmas, FL turns this approach on its head.

From the foundational perspectives discussed, it's evident that FL represents a significant departure from traditional data paradigms. Unlike conventional approaches that centralize data, FL operates on the principle of decentralized learning, offering opportunities to process information at the source without compromising on data sovereignty. The real-world applications of FL, discussed in Chapter 2, further emphasize its transformative potential. From ubiquitous devices like smartphones to expansive organizational networks and the interconnected web of the IoT, FL finds relevance and resonance. However, the same chapter also underscores the challenges that come with implementing FL, particularly regarding scalability, communication overhead, and model consistency.

Chapter 3's exploration of FL's compatibility with different machine learning models demonstrates its versatility. Whether it's the straightforward linear models, the hierarchical structure of tree models, or the intricate neural network architectures, FL can be adapted accordingly. The categorizations into Horizontal, Vertical, and other types of FL further exemplify how FL can be tailored to address unique data distribution and privacy constraints.

Then, data privacy, a cornerstone of FL as outlined in Chapter 4, is among its most compelling features. Advanced techniques such as Differential Privacy, SMPC, and Homomorphic Encryption delineate the robust measures being adopted to safeguard data. These measures emphasize data's stationary nature in FL, mitigating risks associated with data transfer. Furthermore, as gleaned from Section 4.3, addressing non-IID data distribution remains a pressing concern. Ensuring equitable and accurate outcomes in situations with data skewness will be paramount. However, there's also recognition of the inherent challenges: achieving practical and efficient computations using Homomorphic Encryption remains resource-intensive, and perfecting data privacy techniques to adapt to dynamic real-world scenarios is still an ongoing effort.

The algorithms and frameworks that underpin FL, as detailed in Chapter 5, are the mathematical and technical heart of this approach. Algorithms like Federated Averaging offer efficient ways to aggregate learnings from decentralized data sources. Yet, the chapter also highlights the need for more research, especially in areas like asynchronous learning methods, to optimize performance and reduce communication overheads.

More specifically, the introduction to FL algorithms (Section 5.1.1) demystifies the concept, illustrating that FL is more than just a theoretical novelty. It is underpinned by powerful algorithms that guarantee its effectiveness. As we journey through Sections 5.1.2 to 5.1.7, it becomes evident that FL is buoyed by a vast array of algorithms. Whether it's FedAvg or asynchronous techniques, each is tailored to meet specific needs and operational constraints. Collectively, they form the robust infrastructure of FL. Adding to this is the comparative benchmark in Section 5.1.8, which showcases the pinnacle of current FL algorithms, setting a benchmark for excellence in the field.

Looking ahead, the potential directions for FL are multifaceted. There is a palpable need for refining the existing algorithms, particularly with innovations like adaptive learning rates that can cater to dynamic data distributions. The recent CO-OP algorithm emerges as a beacon of promise in this direction. Another frontier is the realm of communication efficiency. As the field leans towards communication-efficient algorithms, there's an impetus to tackle challenges of sparse communication, potentially merging with the principles of edge computing. However, algorithms alone don't suffice. The practical deployment and management of FL require a suite of dedicated frameworks and tools, as discussed in Section 5.2. These tools serve as the bedrock, facilitating the smooth execution of the aforementioned algorithms.

Also, tools such as TensorFlow Federated and PySyft signal the growing ecosystem supporting FL, but they also represent the infancy of tools designed explicitly for federated contexts, hinting at the vast developmental trajectory ahead.

However, the optimism surrounding FL is tempered by the vulnerabilities it's susceptible to, as discussed in Chapter 6. While FL's decentralized nature offers unique privacy advantages, it's not exempt from potential attacks. Whether it's the threat of data poisoning impacting model integrity or backdoor attacks that compromise model outputs, the security landscape of FL is complex. The ongoing research in this space, aimed at countering such adversarial actions, underscores the importance of holistic security frameworks for FL.

In essence, FL, as dissected through the chapters, stands at the intersection of innovation and challenge. It embodies the next frontier of machine learning, weaving together the narratives of decentralization, privacy, adaptability, and security. It embodies the next frontier of machine learning, weaving together the narratives of decentralization, privacy, adaptability, and security. As the domain continues to evolve, ongoing research and collaboration will be pivotal in navigating the intricacies and realizing FL's full potential.

7.2 Future Implications and Directions

As the world gradually transitions into an era where the convergence of machine learning and data privacy takes center stage, the prominence of FL becomes unmistakably clear. It's no longer a question of "if" but "how" FL will revolutionize this nexus. Delving deper into the horizon that awaits, several pivotal aspects demand our unwavering attention.

Collaboration and Standardization:
Delving into Chapter 3's discourse on best practices, standards, and prospective trajectories for data privacy, one can't help but emphasize the need for concerted collaboration. With the

fragmented nature of the present FL landscape, disparate systems often operate in silos, leading to potential inefficiencies or misalignments. By fostering a culture of collaboration, not only can shared challenges be addressed more holistically, but knowledge can be disseminated more evenly. Alongside, the clamor for standardized protocols grows louder. A cohesive framework, universally accepted and adopted, can streamline efforts, reduce redundancies, and establish a foundation upon which further innovations can confidently be built. This standardization also promises a level of interoperability, ensuring that various FL systems can communicate and collaborate seamlessly.

Balance between Personalization and Privacy:
At the heart of FL lies a dual quest: to craft personalized experiences while fiercely guarding data privacy. This equilibrium is delicate and often elusive. As algorithms become more sophisticated in their personalization endeavors, they inadvertently risk breaching the sanctum of privacy. For instance, an algorithm that knows too much can inadvertently reveal sensitive information. Hence, the challenge lies not just in refining techniques but in deepening our understanding of the interplay between personalization and privacy. There's a need to develop mechanisms that can intuitively gauge the boundary where optimal personalization doesn't compromise data sanctity. This dynamic balance might very well be the defining challenge for FL in the coming years.

Broader Application Areas:
While the present discourse has spotlighted several application areas for FL, one must acknowledge that we've merely scratched the surface. The adaptability and versatility of FL hint at its potential applicability across sectors we haven't even considered yet. From healthcare to urban planning, from agriculture to space research, the decentralized nature of FL holds the promise of transforming data analysis and application. Intensive research initiatives and exploratory projects can help identify these novel domains, ensuring that FL's benefits are reaped far and wide.

Continual Vigilance against Attacks:
The evolutionary nature of technology is a double-edged sword. As FL techniques become more refined, adversaries too arm themselves with sophisticated tools and strategies. The landscape of cyber threats is not static; it's a relentless flux where new vulnerabilities can emerge from the slightest of oversights. The threats identified, like Model Inversion or Sybil Attacks, are just the tip of the iceberg. A proactive stance, rooted in continuous research, monitoring, and system upgrades, is imperative. Defense mechanisms need to be agile, adaptive, and ever-evolving, mirroring the very nature of the threats they seek to thwart.

In summary, as the curtain rises on the FL-dominated era, it beckons stakeholders to embrace both its promises and challenges with equal fervor. It's a journey replete with opportunities, but only a synergistic blend of collaboration, understanding, exploration, and vigilance can truly unlock FL's transformative potential.

7.3 Concluding Note

The voyage through the realm of FL has been both enlightening and challenging. This thesis serves as a testament to the potential, challenges, and dynamic nature of FL. As we move forward, the lessons learned will undoubtedly guide researchers, practitioners, and enthusiasts in harnessing the full power of this decentralized learning paradigm.

Bibliography

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, ‘Communication-Efficient Learning of Deep Networks from Decentralized Data’, in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, PMLR, Apr. 2017, pp. 1273–1282. Accessed: Aug. 28, 2023. [Online]. Available: https://proceedings.mlr.press/v54/mcmahan17a.html

[2] J. C. Jiang, B. Kantarci, S. Oktug, and T. Soyata, ‘Federated Learning in Smart City Sensing: Challenges and Opportunities’, *Sensors*, vol. 20, no. 21, p. 6230, Oct. 2020, doi: 10.3390/s20216230.

[3] H. S. Sikandar, H. Waheed, S. Tahir, S. U. R. Malik, and W. Rafique, ‘A Detailed Survey on Federated Learning Attacks and Defenses’, *Electronics*, vol. 12, no. 2, Art. no. 2, Jan. 2023, doi: 10.3390/electronics12020260.

[4] Y. Liu, L. Zhang, N. Ge, and G. Li, ‘A Systematic Literature Review on Federated Learning: From A Model Quality Perspective’. arXiv, Dec. 01, 2020. doi: 10.48550/arXiv.2012.01973.

[5] Flower Framework 1.5.0, ‘What is Federated Learning?’ Accessed: Aug. 28, 2023. [Online]. Available: https://flower.dev/docs/framework/tutorial-what-is-federated-learning.html

[6] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, ‘Communication-Efficient Learning of Deep Networks from Decentralized Data’. arXiv, Jan. 26, 2023. doi: 10.48550/arXiv.1602.05629.

[7] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, ‘Federated Learning: Challenges, Methods, and Future Directions’, *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, Feb. 2020, doi: 10.1109/MSP.2020.2975749.

[8] C. Sammut and G. I. Webb, Eds., *Encyclopedia of Machine Learning*. Boston, MA: Springer US, 2010. doi: 10.1007/978-0-387-30164-8.

[9] C. Sammut and G. I. Webb, Eds., *Encyclopedia of Machine Learning and Data Mining*. Boston, MA: Springer US, 2017. doi: 10.1007/978-1-4899-7687-1.

[10] Open Risk Manual, ‘Federated Learning Glossary’. Accessed: Aug. 28, 2023. [Online]. Available: https://www.openriskmanual.org/wiki/Federated_Learning_Glossary

[11] P. Mammen, ‘Federated Learning: Opportunities and Challenges’, *ArXiv*, Jan. 2021, Accessed: Aug. 28, 2023. [Online]. Available: https://www.semanticscholar.org/paper/Federated-Learning%3A-Opportunities-and-Challenges-Mammen/012f7bbd7ce4eedb105d79c49388b49d0ae7c728

[12] T. Zhang, L. Gao, C. He, M. Zhang, B. Krishnamachari, and A. S. Avestimehr, ‘Federated Learning for the Internet of Things: Applications, Challenges, and Opportunities’, *IEEE Internet Things Mag.*, vol. 5, no. 1, pp. 24–29, Mar. 2022, doi: 10.1109/IOTM.004.2100182.

[13] P. Kairouz *et al.*, ‘Advances and open problems in federated learning’, *Found. Trends Mach. Learn.*, vol. 14, no. 1–2, pp. 1–210, Jun. 2021, doi: 10.1561/2200000083.

[14] A. Rehman, I. Razzak, and G. Xu, ‘Federated Learning for Privacy Preservation of Healthcare Data From Smartphone-Based Side-Channel Attacks’, *IEEE J. Biomed. Health Inform.*, vol. 27, no. 2, Art. no. 2, Feb. 2023, doi: 10.1109/JBHI.2022.3171852.

[15] N. Truong, K. Sun, S. Wang, F. Guitton, and Y. Guo, ‘Privacy preservation in federated learning: An insightful survey from the GDPR perspective’, *Comput. Secur.*, vol. 110, p. 102402, Nov. 2021, doi: 10.1016/j.cose.2021.102402.

[16] J.-P. A. Yaacoub, H. N. Noura, and O. Salman, ‘Security of federated learning with IoT systems: Issues, limitations, challenges, and solutions’, *Internet Things Cyber-Phys. Syst.*, vol. 3, pp. 155–179, Jan. 2023, doi: 10.1016/j.iotcps.2023.04.001.

[17] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, ‘A survey on federated learning’, *Knowl.-Based Syst.*, vol. 216, p. 106775, Nov. 2021, doi: 10.1016/j.knosys.2021.106775.

[18] L. Li, Y. Fan, M. Tse, and K.-Y. Lin, ‘A review of applications in federated learning’, *Comput. Ind. Eng.*, vol. 149, p. 106854, Aug. 2020, doi: 10.1016/j.cie.2020.106854.

[19] D. Kang and C. W. Ahn, ‘Communication Cost Reduction with Partial Structure in Federated Learning’, *Electronics*, vol. 10, no. 17, Art. no. 17, Jan. 2021, doi: 10.3390/electronics10172081.

[20] X. Yao, C. Huang, and L. Sun, ‘Two-Stream Federated Learning: Reduce the Communication Costs’, in *2018 IEEE Visual Communications and Image Processing (VCIP)*, Dec. 2018, pp. 1–4. doi: 10.1109/VCIP.2018.8698609.

[21] Y. Jiang, J. Konečný, K. Rush, and S. Kannan, ‘Improving Federated Learning Personalization via Model Agnostic Meta Learning’, Sep. 2019, Accessed: Aug. 30, 2023. [Online]. Available: https://openreview.net/forum?id=BkeaEyBYDB

[22] A. Fallah, A. Mokhtari, and A. Ozdaglar, ‘Personalized Federated Learning with Theoretical Guarantees: A Model-Agnostic Meta-Learning Approach’, in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2020, pp. 3557–3568. Accessed: Aug. 30, 2023. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/24389bfe4fe2eba8bf9aa9203a44cdad-Abstract.html

[23] K. Bonawitz *et al.*, ‘Towards Federated Learning at Scale: System Design’, *Proc. Mach. Learn. Syst.*, 2019, doi: 10.48550/ARXIV.1902.01046.

[24] H. G. Abreha, M. Hayajneh, and M. A. Serhani, ‘Federated Learning in Edge Computing: A Systematic Survey’, *Sensors*, vol. 22, no. 2, Art. no. 2, Jan. 2022, doi: 10.3390/s22020450.

[25] ‘Real-time End-to-End Federated Learning: An Automotive Case Study’. Accessed: Aug. 30, 2023. [Online]. Available: https://www.computer.org/csdl/proceedings-article/compsac/2021/246300a459/1wLcxK9WHW8

[26] M. Grama, M. Musat, L. Muñoz-González, J. Passerat-Palmbach, D. Rueckert, and A. Alansary, ‘Robust Aggregation for Adaptive Privacy Preserving Federated Learning in Healthcare’. arXiv, Sep. 17, 2020. Accessed: Aug. 30, 2023. [Online]. Available: http://arxiv.org/abs/2009.08294

[27] H. Lycklama, L. Burkhalter, A. Viand, N. Küchler, and A. Hithnawi, ‘RoFL: Robustness of Secure Federated Learning’, *arXiv e-prints*. Jul. 01, 2021. doi: 10.48550/arXiv.2107.03311.

[28] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, ‘Communication-Efficient Learning of Deep Networks from Decentralized Data’, in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, PMLR, Apr. 2017, pp. 1273–1282. Accessed: Aug. 28, 2023. [Online]. Available: https://proceedings.mlr.press/v54/mcmahan17a.html

[29] C. Yang *et al.*, ‘Characterizing Impacts of Heterogeneity in Federated Learning upon Large-Scale Smartphone Data’, in *Proceedings of the Web Conference 2021*, Ljubljana Slovenia: ACM, Apr. 2021, pp. 935–946. doi: 10.1145/3442381.3449851.

[30] A. Rehman, I. Razzak, and G. Xu, ‘Federated Learning for Privacy Preservation of Healthcare Data From Smartphone-Based Side-Channel Attacks’, *IEEE J. Biomed. Health Inform.*, vol. 27, no. 2, pp. 684–690, Feb. 2023, doi: 10.1109/JBHI.2022.3171852.

[31] S. Ek, F. Portet, P. Lalanda, and G. Vega, ‘Evaluation and comparison of federated learning algorithms for Human Activity Recognition on smartphones’, *Pervasive Mob. Comput.*, vol. 87, p. 101714, Dec. 2022, doi: 10.1016/j.pmcj.2022.101714.

[32] L. Li, Y. Fan, and K.-Y. Lin, 'A Survey on federated learning', in *2020 IEEE 16th International Conference on Control & Automation (ICCA)*, Oct. 2020, pp. 791–796. doi: 10.1109/ICCA51439.2020.9264412.

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

[34] O. for C. Rights (OCR), 'Summary of the HIPAA Privacy Rule', HHS.gov. Accessed: Aug. 28, 2023. [Online]. Available: https://www.hhs.gov/hipaa/for-professionals/privacy/laws-regulations/index.html

[35] H. B. McMahan, E. Moore, D. Ramage, and B. A. Y. Arcas, 'Federated Learning of Deep Networks using Model Averaging', *ArXiv*, Feb. 2016, Accessed: Aug. 28, 2023. [Online]. Available: https://www.semanticscholar.org/paper/Federated-Learning-of-Deep-Networks-using-Model-McMahan-Moore/8b419080cd37bdc30872b76f405ef6a93eae3304

[36] N. Cavdar Aksoy, E. Tumer Kabadayi, C. Yilmaz, and A. Kocak Alan, 'A typology of personalisation practices in marketing in the digital age', *J. Mark. Manag.*, vol. 37, no. 11–12, pp. 1091–1122, Jul. 2021, doi: 10.1080/0267257X.2020.1866647.

[37] Y. Liu *et al.*, 'Vertical Federated Learning', 2022, doi: 10.48550/ARXIV.2211.12814.

[38] E. Bakopoulou, B. Tillman, and A. Markopoulou, 'FedPacket: A Federated Learning Approach to Mobile Packet Classification', *IEEE Trans. Mob. Comput.*, vol. 21, no. 10, pp. 3609–3628, Oct. 2022, doi: 10.1109/TMC.2021.3058627.

[39] H. Jiang, T. Cui, and K. Yang, 'Design of Sponsored Search Auction Mechanism for Federated Learning Advertising Platform', *Comput. Intell. Neurosci.*, vol. 2022, pp. 1–15, Apr. 2022, doi: 10.1155/2022/5787491.

[40] Y. Li, X. Tao, X. Zhang, J. Liu, and J. Xu, 'Privacy-Preserved Federated Learning for Autonomous Driving', *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 8423–8434, Jul. 2022, doi: 10.1109/TITS.2021.3081560.

[41] Z. Du, C. Wu, T. Yoshinaga, K.-L. A. Yau, Y. Ji, and J. Li, 'Federated Learning for Vehicular Internet of Things: Recent Advances and Open Issues', *IEEE Open J. Comput. Soc.*, vol. 1, pp. 45–61, 2020, doi: 10.1109/OJCS.2020.2992630.

[42] 'Federated Learning for Internet of Things: Recent Advances, Taxonomy, and Open Challenges | IEEE Journals & Magazine | IEEE Xplore'. Accessed: Aug. 30, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/9460016

[43] 'Anti-money laundering and countering the financing of terrorism legislative package'. Accessed: Aug. 30, 2023. [Online]. Available: https://finance.ec.europa.eu/publications/anti-money-laundering-and-countering-financing-terrorism-legislative-package_en

[44] Q. Yang, Y. Liu, T. Chen, and Y. Tong, 'Federated Machine Learning: Concept and Applications', *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 2, pp. 1–19, Nov. 2019, doi: 10.1145/3298981.

[45] O. Toujani, A. Benhamza, and Y. Hachaïchi, *The Impact of Big Data and Artificial Intelligence in the Insurance Sector*. 2023. doi: 10.13140/RG.2.2.17883.85286.

[46] M. Eling, D. Nuessle, and J. Staubli, 'The impact of artificial intelligence along the insurance value chain and on the insurability of risks', *Geneva Pap. Risk Insur. - Issues Pract.*, vol. 47, no. 2, pp. 205–241, Apr. 2022, doi: 10.1057/s41288-020-00201-7.

[47] N. Kumar, J. Srivastava, and H. Bisht, *Artificial Intelligence in Insurance Sector*. 2019.

[48] Y.-A. Mamiko, 'The Impact of Big Data and Artificial Intelligence (AI) in the Insurance Sector', 2020.

[49] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, 'A survey on federated learning', *Knowl.-Based Syst.*, vol. 216, p. 106775, Mar. 2021, doi: 10.1016/j.knosys.2021.106775.

[50] K. Wei *et al.*, 'Federated Learning With Differential Privacy: Algorithms and Performance Analysis', *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 3454–3469, 2020, doi: 10.1109/TIFS.2020.2988575.

[51] O. R. A. Almanifi, C.-O. Chow, M.-L. Tham, J. H. Chuah, and J. Kanesan, 'Communication and computation efficiency in Federated Learning: A survey', *Internet Things*, vol. 22, p. 100742, Jul. 2023, doi: 10.1016/j.iot.2023.100742.

[52] A. Singh, P. Vepakomma, O. Gupta, and R. Raskar, 'Detailed comparison of communication efficiency of split learning and federated learning'. arXiv, Sep. 18, 2019. Accessed: Aug. 30, 2023. [Online]. Available: http://arxiv.org/abs/1909.09145

[53] X. Ma, J. Zhu, Z. Lin, S. Chen, and Y. Qin, 'A state-of-the-art survey on solving non-IID data in Federated Learning', *Future Gener. Comput. Syst.*, vol. 135, pp. 244–258, Oct. 2022, doi: 10.1016/j.future.2022.05.003.

[54] H. Wang, Z. Kaplan, D. Niu, and B. Li, 'Optimizing Federated Learning on Non-IID Data with Reinforcement Learning', in *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*, Toronto, ON, Canada: IEEE, Jul. 2020, pp. 1698–1707. doi: 10.1109/INFOCOM41043.2020.9155494.

[55] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, 'Federated Learning with Non-IID Data', 2018, doi: 10.48550/arXiv.1806.00582.

[56] K. Wei *et al.*, 'Federated Learning With Differential Privacy: Algorithms and Performance Analysis', *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 3454–3469, 2020, doi: 10.1109/TIFS.2020.2988575.

[57] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, 'Federated Learning: Challenges, Methods, and Future Directions', *IEEE Signal Process. Mag.*, vol. 37, no. 3, Art. no. 3, May 2020, doi: 10.1109/MSP.2020.2975749.

[58] J. Zhang, J. Chen, D. Wu, B. Chen, and S. Yu, *Poisoning Attack in Federated Learning using Generative Adversarial Nets*. 2019, p. 380. doi: 10.1109/TrustCom/BigDataSE.2019.00057.

[59] A. N. Bhagoji, S. Chakraborty, P. Mittal, and S. Calo, 'Analyzing Federated Learning through an Adversarial Lens', in *Proceedings of the 36th International Conference on Machine Learning*, PMLR, May 2019, pp. 634–643. Accessed: Aug. 30, 2023. [Online]. Available: https://proceedings.mlr.press/v97/bhagoji19a.html

[60] H. S. Sikandar, H. Waheed, S. Tahir, S. U. R. Malik, and W. Rafique, 'A Detailed Survey on Federated Learning Attacks and Defenses', *Electronics*, vol. 12, no. 2, Art. no. 2, Jan. 2023, doi: 10.3390/electronics12020260.

[61] K. Bonawitz, P. Kairouz, B. Mcmahan, and D. Ramage, 'Federated learning and privacy', *Commun. ACM*, vol. 65, no. 4, pp. 90–97, Apr. 2022, doi: 10.1145/3500240.

[62] J. M. Stanton, 'Galton, Pearson, and the Peas: A Brief History of Linear Regression for Statistics Instructors', *J. Stat. Educ.*, vol. 9, no. 3, p. 3, Jan. 2001, doi: 10.1080/10691898.2001.11910537.

[63] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, 'Support vector machines', *IEEE Intell. Syst. Their Appl.*, vol. 13, no. 4, pp. 18–28, Jul. 1998, doi: 10.1109/5254.708428.

[64] C. Cortes and V. Vapnik, 'Support-vector networks', *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995, doi: 10.1007/BF00994018.

[65] 'Linear discriminant analysis: A detailed tutorial - IOS Press'. Accessed: Aug. 30, 2023. [Online]. Available: https://content.iospress.com/articles/ai-communications/aic729

[66] O. A. Montesinos López, A. Montesinos López, and J. Crossa, ‘Overfitting, Model Tuning, and Evaluation of Prediction Performance’, in *Multivariate Statistical Machine Learning Methods for Genomic Prediction*, O. A. Montesinos López, A. Montesinos López, and J. Crossa, Eds., Cham: Springer International Publishing, 2022, pp. 109–139. doi: 10.1007/978-3-030-89010-0_4.

[67] R. Froud, S. H. Hansen, H. K. Ruud, J. Foss, L. Ferguson, and P. M. Fredriksen, ‘Relative Performance of Machine Learning and Linear Regression in Predicting Quality of Life and Academic Performance of School Children in Norway: Data Analysis of a Quasi-Experimental Study’, *J. Med. Internet Res.*, vol. 23, no. 7, p. e22021, Jul. 2021, doi: 10.2196/22021.

[68] J. R. Quinlan, ‘Induction of decision trees’, *Mach. Learn.*, vol. 1, no. 1, pp. 81–106, Mar. 1986, doi: 10.1007/BF00116251.

[69] J. Ye, J.-H. Chow, J. Chen, and Z. Zheng, ‘Stochastic gradient boosted distributed decision trees’, in *Proceedings of the 18th ACM conference on Information and knowledge management*, Hong Kong China: ACM, Nov. 2009, pp. 2061–2064. doi: 10.1145/1645953.1646301.

[70] ‘Random Forests | SpringerLink’. Accessed: Aug. 30, 2023. [Online]. Available: https://link.springer.com/article/10.1023/A:1010933404324

[71] R. Haffar, D. Sánchez, and J. Domingo-Ferrer, ‘Explaining predictions and attacks in federated learning via random forests’, *Appl. Intell.*, vol. 53, no. 1, pp. 169–185, Jan. 2023, doi: 10.1007/s10489-022-03435-1.

[72] L. Alzubaidi *et al.*, ‘Review of deep learning: concepts, CNN architectures, challenges, applications, future directions’, *J. Big Data*, vol. 8, no. 1, p. 53, Mar. 2021, doi: 10.1186/s40537-021-00444-8.

[73] S. Albawi, O. Bayat, S. Al-Azawi, and O. N. Ucan, ‘Social Touch Gesture Recognition Using Convolutional Neural Network’, *Comput. Intell. Neurosci.*, vol. 2018, pp. 1–10, Oct. 2018, doi: 10.1155/2018/6973103.

[74] ‘Max-pooling convolutional neural networks for vision-based hand gesture recognition | IEEE Conference Publication | IEEE Xplore’. Accessed: Aug. 30, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/6144164

[75] C. He, M. Annavaram, and S. Avestimehr, ‘Group Knowledge Transfer: Federated Learning of Large CNNs at the Edge’, in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2020, pp. 14068–14080. Accessed: Aug. 30, 2023. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/a1d4c20b182ad7137ab3606f0e3fc8a4-Abstract.html

[76] ‘A survey on the application of recurrent neural networks to statistical language modeling - ScienceDirect’. Accessed: Aug. 30, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S088523081400093X

[77] A. Sherstinsky, ‘Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network’, *Phys. Nonlinear Phenom.*, vol. 404, p. 132306, Mar. 2020, doi: 10.1016/j.physd.2019.132306.

[78] M. N. Fekri, K. Grolinger, and S. Mir, ‘Distributed load forecasting using smart meter data: Federated learning with Recurrent Neural Networks’, *Int. J. Electr. Power Energy Syst.*, vol. 137, p. 107669, May 2022, doi: 10.1016/j.ijepes.2021.107669.

[79] ‘Water | Free Full-Text | Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting’. Accessed: Aug. 30, 2023. [Online]. Available: https://www.mdpi.com/2073-4441/11/7/1387

[80] L. Gonog and Y. Zhou, ‘A Review: Generative Adversarial Networks’, in *2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, Jun. 2019, pp. 505–510. doi: 10.1109/ICIEA.2019.8833686.

[81] I. Goodfellow *et al.*, ‘Generative Adversarial Nets’, in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2014. Accessed: Aug. 30, 2023. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2014/hash/5ca3e9b122f61f8f06494c97b1afccf3-Abstract.html

[82] A. Tabassum, A. Erbad, W. Lebda, A. Mohamed, and M. Guizani, ‘FEDGAN-IDS: Privacy-preserving IDS using GAN and Federated Learning’, *Comput. Commun.*, vol. 192, pp. 299–310, Aug. 2022, doi: 10.1016/j.comcom.2022.06.015.

[83] ‘Inference attacks based on GAN in federated learning | Emerald Insight’. Accessed: Aug. 30, 2023. [Online]. Available: https://www.emerald.com/insight/content/doi/10.1108/IJWIS-04-2022-0078/full/html

[84] ‘Information | Free Full-Text | A Systematic Review of Transformer-Based Pre-Trained Language Models through Self-Supervised Learning’. Accessed: Aug. 30, 2023. [Online]. Available: https://www.mdpi.com/2078-2489/14/3/187

[85] ‘Electronics | Free Full-Text | Training Vision Transformers in Federated Learning with Limited Edge-Device Resources’. Accessed: Aug. 30, 2023. [Online]. Available: https://www.mdpi.com/2079-9292/11/17/2638

[86] S. Kamei and S. Taghipour, ‘A comparison study of centralized and decentralized federated learning approaches utilizing the transformer architecture for estimating remaining useful life’, *Reliab. Eng. Syst. Saf.*, vol. 233, p. 109130, May 2023, doi: 10.1016/j.ress.2023.109130.

[87] R. Nian, J. Liu, and B. Huang, ‘A review On reinforcement learning: Introduction and applications in industrial process control’, *Comput. Chem. Eng.*, vol. 139, p. 106886, Aug. 2020, doi: 10.1016/j.compchemeng.2020.106886.

[88] R. S. Sutton and A. G. Barto, *Reinforcement learning: an introduction*. in Adaptive computation and machine learning. Cambridge, Mass: MIT Press, 1998.

[89] S. Dridi, ‘Reinforcement Learning - A Systematic Literature Review’, Open Science Framework, preprint, Apr. 2022. doi: 10.31219/osf.io/qxng6.

[90] S. Yan *et al.*, ‘Node Selection Algorithm for Federated Learning Based on Deep Reinforcement Learning for Edge Computing in IoT’, *Electronics*, vol. 12, no. 11, p. 2478, May 2023, doi: 10.3390/electronics12112478.

[91] E. C. Pinto Neto, S. Sadeghi, X. Zhang, and S. Dadkhah, ‘Federated Reinforcement Learning in IoT: Applications, Opportunities and Open Challenges’, *Appl. Sci.*, vol. 13, no. 11, p. 6497, May 2023, doi: 10.3390/app13116497.

[92] D. Zou *et al.*, ‘FedMC: Federated Reinforcement Learning on the Edge with Meta-Critic Networks’, in *2022 IEEE International Performance, Computing, and Communications Conference (IPCCC)*, Austin, TX, USA: IEEE, Nov. 2022, pp. 344–351. doi: 10.1109/IPCCC55026.2022.9894336.

[93] R. O. Ogundokun, S. Misra, R. Maskeliunas, and R. Damasevicius, ‘A Review on Federated Learning and Machine Learning Approaches: Categorization, Application Areas, and Blockchain Technology’, *Information*, vol. 13, no. 5, p. 263, May 2022, doi: 10.3390/info13050263.

[94] N. Letizia, ‘Parallelism and Horizontal Federated Learning: A Review’, Apr. 2023.

[95] W. Huang, T. Li, D. Wang, S. Du, J. Zhang, and T. Huang, ‘Fairness and accuracy in horizontal federated learning’, *Inf. Sci.*, vol. 589, pp. 170–185, Apr. 2022, doi: 10.1016/j.ins.2021.12.102.

[96] Q. Li *et al.*, 'Vertical Federated Learning: Taxonomies, Threats, and Prospects', 2023, doi: 10.48550/ARXIV.2302.01550.

[97] Y. Liu *et al.*, 'Vertical Federated Learning'. arXiv, Nov. 24, 2022. Accessed: Sep. 08, 2023. [Online]. Available: http://arxiv.org/abs/2211.12814

[98] S. Feng and H. Yu, 'Multi-Participant Multi-Class Vertical Federated Learning'. arXiv, Jan. 29, 2020. Accessed: Sep. 08, 2023. [Online]. Available: http://arxiv.org/abs/2001.11154

[99] X. Zhang, W. Yin, M. Hong, and T. Chen, 'Hybrid Federated Learning: Algorithms and Implementation'. arXiv, Feb. 17, 2021. Accessed: Sep. 08, 2023. [Online]. Available: http://arxiv.org/abs/2012.12420

[100] Z. Zhou *et al.*, 'A Novel Optimized Asynchronous Federated Learning Framework', in *2021 IEEE 23rd Int Conf on High Performance Computing & Communications; 7th Int Conf on Data Science & Systems; 19th Int Conf on Smart City; 7th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys)*, Haikou, Hainan, China: IEEE, Dec. 2021, pp. 2363–2370. doi: 10.1109/HPCC-DSS-SmartCity-DependSys53884.2021.00357.

[101] Q. Jing, W. Wang, J. Zhang, H. Tian, and K. Chen, 'Quantifying the Performance of Federated Transfer Learning'. arXiv, Dec. 29, 2019. Accessed: Sep. 08, 2023. [Online]. Available: http://arxiv.org/abs/1912.12795

[102] Y. Liu, Y. Kang, C. Xing, T. Chen, and Q. Yang, 'Secure Federated Transfer Learning', *IEEE Intell. Syst.*, vol. 35, no. 4, pp. 70–82, Jul. 2020, doi: 10.1109/MIS.2020.2988525.

[103] Y. Huang *et al.*, 'A High-Precision Method for 100-Day-Old Classification of Chickens in Edge Computing Scenarios Based on Federated Computing', *Animals*, vol. 12, no. 24, p. 3450, Dec. 2022, doi: 10.3390/ani12243450.

[104] D. Gao, Y. Liu, A. Huang, C. Ju, H. Yu, and Q. Yang, 'Privacy-preserving Heterogeneous Federated Transfer Learning', in *2019 IEEE International Conference on Big Data (Big Data)*, Los Angeles, CA, USA: IEEE, Dec. 2019, pp. 2552–2559. doi: 10.1109/BigData47090.2019.9005992.

[105] K. Bonawitz *et al.*, 'Towards Federated Learning at Scale: System Design', *Proc. Mach. Learn. Syst.*, 2019, doi: 10.48550/ARXIV.1902.01046.

[106] K. Bonawitz *et al.*, 'Practical Secure Aggregation for Privacy Preserving Machine Learning'. 2017. Accessed: Aug. 30, 2023. [Online]. Available: https://eprint.iacr.org/2017/281

[107] X. Liu, H. Li, G. Xu, R. Lu, and M. He, 'Adaptive privacy-preserving federated learning', *Peer--Peer Netw. Appl.*, vol. 13, no. 6, pp. 2356–2366, Nov. 2020, doi: 10.1007/s12083-019-00869-2.

[108] A. Triastcyn and B. Faltings, 'Federated Learning with Bayesian Differential Privacy', in *2019 IEEE International Conference on Big Data (Big Data)*, Dec. 2019, pp. 2587–2596. doi: 10.1109/BigData47090.2019.9005465.

[109] J. Xu, B. S. Glicksberg, C. Su, P. Walker, J. Bian, and F. Wang, 'Federated Learning for Healthcare Informatics', *J. Healthc. Inform. Res.*, vol. 5, no. 1, pp. 1–19, Mar. 2021, doi: 10.1007/s41666-020-00082-4.

[110] Y. Li, Y. Zhou, A. Jolfaei, D. Yu, G. Xu, and X. Zheng, 'Privacy-Preserving Federated Learning Framework Based on Chained Secure Multiparty Computing', *IEEE Internet Things J.*, vol. 8, no. 8, pp. 6178–6186, Apr. 2021, doi: 10.1109/JIOT.2020.3022911.

[111] W. Du and M. J. Atallah, 'Secure Multi-Party Computation Problems and Their Applications: A Review And Open Problems', *Electr. Eng. Comput. Sci. 11 Httpssurfacesyredueecs11*.

[112] J. Ma, S.-A. Naas, S. Sigg, and X. Lyu, ‘Privacy-preserving federated learning based on multi-key homomorphic encryption’, *Int. J. Intell. Syst.*, vol. 37, no. 9, pp. 5880–5901, 2022, doi: 10.1002/int.22818.

[113] C. Zhang, S. Li, J. Xia, W. Wang, F. Yan, and Y. Liu, ‘BatchCrypt: Efficient Homomorphic Encryption for Cross-Silo Federated Learning’.

[114] P. Fremantle, B. Aziz, J. Kopecký, and P. Scott, ‘Federated Identity and Access Management for the Internet of Things’, in *2014 International Workshop on Secure Internet of Things*, Sep. 2014, pp. 10–17. doi: 10.1109/SIoT.2014.8.

[115] J. Alsulami, ‘Towards a Federated Identity and Access Management Across Universities’, 2021.

[116] K. Taylor and J. Murty, ‘Implementing Role Based Access Control for Federated Information Systems on the Web’.

[117] B. Anggorojati, P. N. Mahalle, N. R. Prasad, and R. Prasad, ‘Capability-based Access Control Delegation Model on the Federated IoT Network: The 15th International Symposium on Wireless Personal Multimedia Communications’, *2012 15th Int. Symp. Wirel. Pers. Multimed. Commun. WPMC*, pp. 604–608, 2012.

[118] S. Vellanki and M. Sobolewski, ‘Federated Role-based Access Control in Exertion-oriented Programming’, *Inf. Assur.*, 2009.

[119] M. Fisichella, G. Lax, and A. Russo, ‘Partially-federated learning: A new approach to achieving privacy and effectiveness’, *Inf. Sci.*, vol. 614, pp. 534–547, Oct. 2022, doi: 10.1016/j.ins.2022.10.082.

[120] M. Asad, M. Aslam, S. F. Jilani, S. Shaukat, and M. Tsukada, ‘SHFL: K-Anonymity-Based Secure Hierarchical Federated Learning Framework for Smart Healthcare Systems’, *Future Internet*, vol. 14, no. 11, Art. no. 11, Nov. 2022, doi: 10.3390/fi14110338.

[121] I. Tenison, S. A. Sreeramadas, V. Mugunthan, E. Oyallon, E. Belilovsky, and I. Rish, ‘Gradient Masked Averaging for Federated Learning’. arXiv, Jan. 28, 2022. doi: 10.48550/arXiv.2201.11986.

[122] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, ‘Federated Learning: Strategies for Improving Communication Efficiency’. arXiv, Oct. 30, 2017. Accessed: Aug. 30, 2023. [Online]. Available: http://arxiv.org/abs/1610.05492

[123] ‘EIFFeL | Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security’. Accessed: Aug. 30, 2023. [Online]. Available: https://dl.acm.org/doi/abs/10.1145/3548606.3560611

[124] V. S. Mai, R. La, T. Zhang, Y. X. Huang, and A. Battou, ‘Federated Learning with Server Learning for Non-IID Data’, *NIST*, Mar. 2023, Accessed: Sep. 08, 2023. [Online]. Available: https://www.nist.gov/publications/federated-learning-server-learning-non-iid-data

[125] ‘ISO/IEC 2382-31:1997(en), Information technology — Vocabulary — Part 31: Artificial intelligence — Machine learning’. Accessed: Sep. 08, 2023. [Online]. Available: https://www.iso.org/obp/ui/#iso:std:iso-iec:2382:-31:ed-1:v1:en

[126] ‘IEEE Standards Association’, IEEE Standards Association. Accessed: Sep. 08, 2023. [Online]. Available: https://standards.ieee.org

[127] R. Gosselin, L. Vieu, F. Loukil, and A. Benoit, ‘Privacy and Security in Federated Learning: A Survey’, *Appl. Sci.*, vol. 12, no. 19, p. 9901, Oct. 2022, doi: 10.3390/app12199901.

[128] T. Nguyen and M. T. Thai, ‘Preserving Privacy and Security in Federated Learning’, *IEEEACM Trans. Netw.*, pp. 1–11, 2023, doi: 10.1109/TNET.2023.3302016.

[129] Z. Xing *et al.*, ‘Zero-Knowledge Proof-based Practical Federated Learning on Blockchain’. arXiv, Apr. 24, 2023. Accessed: Sep. 08, 2023. [Online]. Available: http://arxiv.org/abs/2304.05590

[130] M. Xu *et al.*, ‘Stochastic Resource Allocation in Quantum Key Distribution for Secure Federated Learning’, in *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*,

Rio de Janeiro, Brazil: IEEE, Dec. 2022, pp. 4377–4382. doi: 10.1109/GLOBECOM48099.2022.10001071.

[131] R. Kaewpuang, M. Xu, D. Niyato, H. Yu, Z. Xiong, and X. S. Shen, ‘Adaptive Resource Allocation in Quantum Key Distribution (QKD) for Federated Learning’. arXiv, Aug. 29, 2022. Accessed: Sep. 08, 2023. [Online]. Available: http://arxiv.org/abs/2208.11270

[132] ‘Personalized federated learning for a better customer experience’, Amazon Science. Accessed: Sep. 08, 2023. [Online]. Available: https://www.amazon.science/blog/personalized-federated-learning-for-a-better-customer-experience

[133] T. Yang *et al.*, ‘Applied Federated Learning: Improving Google Keyboard Query Suggestions’. arXiv, Dec. 06, 2018. Accessed: Sep. 08, 2023. [Online]. Available: http://arxiv.org/abs/1812.02903

[134] ‘Federated Learning: Collaborative Machine Learning without Centralized Training Data – Google Research Blog’. Accessed: Sep. 08, 2023. [Online]. Available: https://blog.research.google/2017/04/federated-learning-collaborative.html

[135] M. G. Arivazhagan, V. Aggarwal, A. K. Singh, and S. Choudhary, ‘Federated Learning with Personalization Layers’. arXiv, Dec. 02, 2019. Accessed: Sep. 08, 2023. [Online]. Available: http://arxiv.org/abs/1912.00818

[136] C. Briggs, Z. Fan, and P. Andras, ‘Federated learning with hierarchical clustering of local updates to improve training on non-IID data’, in *2020 International Joint Conference on Neural Networks (IJCNN)*, Glasgow, United Kingdom: IEEE, Jul. 2020, pp. 1–9. doi: 10.1109/IJCNN48605.2020.9207469.

[137] M. R. Sprague *et al.*, ‘Asynchronous Federated Learning for Geospatial Applications’, in *ECML PKDD 2018 Workshops*, vol. 967, A. Monreale, C. Alzate, M. Kamp, Y. Krishnamurthy, D. Paurat, M. Sayed-Mouchaweh, A. Bifet, J. Gama, and R. P. Ribeiro, Eds., in Communications in Computer and Information Science, vol. 967. , Cham: Springer International Publishing, 2019, pp. 21–28. doi: 10.1007/978-3-030-14880-5_2.

[138] Q. Li, Y. Diao, Q. Chen, and B. He, ‘Federated Learning on Non-IID Data Silos: An Experimental Study’. arXiv, Oct. 28, 2021. Accessed: Sep. 08, 2023. [Online]. Available: http://arxiv.org/abs/2102.02079

[139] C. Gong, Z. Zheng, Y. Shao, B. Li, F. Wu, and G. Chen, ‘To Store or Not? Online Data Selection for Federated Learning with Limited Storage’. arXiv, Feb. 26, 2023. Accessed: Sep. 08, 2023. [Online]. Available: http://arxiv.org/abs/2209.00195

[140] Z. Zhu, J. Hong, and J. Zhou, ‘Data-Free Knowledge Distillation for Heterogeneous Federated Learning’, in *Proceedings of the 38th International Conference on Machine Learning*, PMLR, Jul. 2021, pp. 12878–12889. Accessed: Sep. 08, 2023. [Online]. Available: https://proceedings.mlr.press/v139/zhu21b.html

[141] S. Alam, L. Liu, M. Yan, and M. Zhang, ‘FedRolex: Model-Heterogeneous Federated Learning with Rolling Sub-Model Extraction’, 2022, doi: 10.48550/ARXIV.2212.01548.

[142] Z. Zhao, J. Xia, L. Fan, X. Lei, G. K. Karagiannidis, and A. Nallanathan, ‘System Optimization of Federated Learning Networks With a Constrained Latency’, *IEEE Trans. Veh. Technol.*, vol. 71, no. 1, pp. 1095–1100, Jan. 2022, doi: 10.1109/TVT.2021.3128559.

[143] F. Haddadpour, M. M. Kamani, A. Mokhtari, and M. Mahdavi, ‘Federated Learning with Compression: Unified Analysis and Sharp Guarantees’, in *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, PMLR, Mar. 2021, pp. 2350–2358. Accessed: Sep. 08, 2023. [Online]. Available: https://proceedings.mlr.press/v130/haddadpour21a.html

[144] K. Bonawitz *et al.*, 'Practical Secure Aggregation for Federated Learning on User-Held Data'. arXiv, Nov. 14, 2016. Accessed: Sep. 08, 2023. [Online]. Available: http://arxiv.org/abs/1611.04482

[145] P. Zhao, Y. Huang, J. Gao, L. Xing, H. Wu, and H. Ma, 'Federated Learning-Based Collaborative Authentication Protocol for Shared Data in Social IoV', *IEEE Sens. J.*, vol. 22, no. 7, pp. 7385–7398, Apr. 2022, doi: 10.1109/JSEN.2022.3153338.

[146] H. Wang and J. Xu, 'Combating Client Dropout in Federated Learning via Friend Model Substitution'. arXiv, May 08, 2023. Accessed: Sep. 08, 2023. [Online]. Available: http://arxiv.org/abs/2205.13222

[147] X. Wu, F. Huang, Z. Hu, and H. Huang, 'Faster Adaptive Federated Learning', *Proc. AAAI Conf. Artif. Intell.*, vol. 37, no. 9, Art. no. 9, Jun. 2023, doi: 10.1609/aaai.v37i9.26235.

[148] 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*, Rennes France: ACM, Dec. 2018, pp. 1–8. doi: 10.1145/3286490.3286559.

[149] Y. LeCun and C. Cortes, 'The mnist database of handwritten digits', 2005. Accessed: Sep. 08, 2023. [Online]. Available: https://www.semanticscholar.org/paper/The-mnist-database-of-handwritten-digits-LeCun-Cortes/dc52d1ede1b90bf9d296bc5b34c9310b7eaa99a2

[150] T. Sun, D. Li, and B. Wang, 'Decentralized Federated Averaging', *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–12, 2022, doi: 10.1109/TPAMI.2022.3196503.

[151] Q. Li, C. Tai, and W. E, 'Stochastic Modified Equations and Dynamics of Stochastic Gradient Algorithms I: Mathematical Foundations', *J. Mach. Learn. Res.*, vol. 20, no. 40, pp. 1–47, 2019.

[152] H. Yuan and T. Ma, 'Federated Accelerated Stochastic Gradient Descent', in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2020, pp. 5332–5344. Accessed: Sep. 08, 2023. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/39d0a8908fbe6c18039ea8227f827023-Abstract.html

[153] K. el Mekkaoui, D. Mesquita, P. Blomstedt, and S. Kaski, 'Federated stochastic gradient Langevin dynamics', in *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, PMLR, Dec. 2021, pp. 1703–1712. Accessed: Sep. 08, 2023. [Online]. Available: https://proceedings.mlr.press/v161/mekkaoui21a.html

[154] S. Nikoloutsopoulos, I. Koutsopoulos, and M. K. Titsias, 'Personalized Federated Learning with Exact Stochastic Gradient Descent'. arXiv, Feb. 20, 2022. Accessed: Sep. 08, 2023. [Online]. Available: http://arxiv.org/abs/2202.09848

[155] 'PPML Series #2 - Federated Optimization Algorithms - FedSGD and FedAvg | Shreyansh Singh'. Accessed: Sep. 08, 2023. [Online]. Available: https://shreyansh26.github.io/post/2021-12-18_federated_optimization_fedavg/

[156] M. Moshawrab, M. Adda, A. Bouzouane, H. Ibrahim, and A. Raad, 'Reviewing Federated Learning Aggregation Algorithms; Strategies, Contributions, Limitations and Future Perspectives', *Electronics*, vol. 12, no. 10, p. 2287, May 2023, doi: 10.3390/electronics12102287.

[157] R. Hu, Y. Gong, and Y. Guo, 'Federated Learning with Sparsification-Amplified Privacy and Adaptive Optimization', in *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, Montreal, Canada: International Joint Conferences on Artificial Intelligence Organization, Aug. 2021, pp. 1463–1469. doi: 10.24963/ijcai.2021/202.

[158] N. Tonellotto, A. Gotta, F. M. Nardini, D. Gadler, and F. Silvestri, 'Neural network quantization in federated learning at the edge', *Inf. Sci.*, vol. 575, pp. 417–436, Oct. 2021, doi: 10.1016/j.ins.2021.06.039.

[159] X. Wu, Y. Zhang, M. Shi, P. Li, R. Li, and N. N. Xiong, 'An adaptive federated learning scheme with differential privacy preserving', *Future Gener. Comput. Syst.*, vol. 127, pp. 362–372, Feb. 2022, doi: 10.1016/j.future.2021.09.015.

[160] Y. Deng, M. M. Kamani, and M. Mahdavi, 'Adaptive Personalized Federated Learning', Oct. 2020, Accessed: Sep. 09, 2023. [Online]. Available: https://openreview.net/forum?id=g0a-XYjpQ7r

[161] A. A. Lydia and F. S. Francis, 'Adagrad - An Optimizer for Stochastic Gradient Descent', vol. 6, no. 0972, 2019.

[162] J. Jin, J. Ren, Y. Zhou, L. Lyu, J. Liu, and D. Dou, 'Accelerated Federated Learning with Decoupled Adaptive Optimization', 2022, doi: 10.48550/ARXIV.2207.07223.

[163] 'Accelerating Fair Federated Learning: Adaptive Federated Adam', *Trans. Mach. Learn. Res.*, May 2023, Accessed: Sep. 09, 2023. [Online]. Available: https://openreview.net/forum?id=xSPrjsdhvF

[164] S. Caldas *et al.*, 'LEAF: A Benchmark for Federated Settings'. arXiv, Dec. 09, 2019. Accessed: Sep. 09, 2023. [Online]. Available: http://arxiv.org/abs/1812.01097

[165] R. Johnson and T. Zhang, 'Accelerating Stochastic Gradient Descent using Predictive Variance Reduction', in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2013. Accessed: Sep. 09, 2023. [Online]. Available: https://papers.nips.cc/paper_files/paper/2013/hash/ac1dd209cbcc5e5d1c6e28598e8cbbe8-Abstract.html

[166] T. T. F. Authors, 'TensorFlow Federated'. Dec. 2018. Accessed: Sep. 09, 2023. [Online]. Available: https://github.com/tensorflow/federated

[167] 'TensorFlow Federated'. Accessed: Sep. 09, 2023. [Online]. Available: https://www.tensorflow.org/federated

[168] A. Ziller *et al.*, 'PySyft: A Library for Easy Federated Learning', in *Federated Learning Systems: Towards Next-Generation AI*, M. H. ur Rehman and M. M. Gaber, Eds., in Studies in Computational Intelligence. , Cham: Springer International Publishing, 2021, pp. 111–139. doi: 10.1007/978-3-030-70604-3_5.

[169] A. Budrionis, M. Miara, P. Miara, S. Wilk, and J. G. Bellika, 'Benchmarking PySyft Federated Learning Framework on MIMIC-III Dataset', *IEEE Access*, vol. 9, pp. 116869–116878, 2021, doi: 10.1109/ACCESS.2021.3105929.

[170] FedAI, 'An Industrial Grade Federated Learning Framework', Fate. Accessed: Sep. 09, 2023. [Online]. Available: https://fate.fedai.org/

[171] 'FederatedAI/FATE'. Federated AI Ecosystem, Sep. 08, 2023. Accessed: Sep. 09, 2023. [Online]. Available: https://github.com/FederatedAI/FATE

[172] M. Nasr, R. Shokri, and A. Houmansadr, 'Comprehensive Privacy Analysis of Deep Learning: Passive and Active White-box Inference Attacks against Centralized and Federated Learning', in *2019 IEEE Symposium on Security and Privacy (SP)*, San Francisco, CA, USA: IEEE, May 2019, pp. 739–753. doi: 10.1109/SP.2019.00065.

[173] M. Alazab, S. P. Rm, P. M, P. K. R. Maddikunta, T. R. Gadekallu, and Q.-V. Pham, 'Federated Learning for Cybersecurity: Concepts, Challenges, and Future Directions', *IEEE Trans. Ind. Inform.*, vol. 18, no. 5, pp. 3501–3509, May 2022, doi: 10.1109/TII.2021.3119038.

[174] K. Varma, Y. Zhou, N. Baracaldo, and A. Anwar, 'LEGATO: A LayerwisE Gradient AggregaTiOn Algorithm for Mitigating Byzantine Attacks in Federated Learning'. arXiv, Jul. 26, 2021. Accessed: Sep. 09, 2023. [Online]. Available: http://arxiv.org/abs/2107.12490

[175] X. Yuan, P. He, Q. Zhu, and X. Li, 'Adversarial Examples: Attacks and Defenses for Deep Learning'. arXiv, Jul. 06, 2018. Accessed: Sep. 09, 2023. [Online]. Available: http://arxiv.org/abs/1712.07107

[176] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, 'You Only Look Once: Unified, Real-Time Object Detection', in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.

[177] S. Shen, S. Tople, and P. Saxena, 'Auror: defending against poisoning attacks in collaborative deep learning systems', in *Proceedings of the 32nd Annual Conference on Computer Security Applications*, Los Angeles California USA: ACM, Dec. 2016, pp. 508–519. doi: 10.1145/2991079.2991125.

[178] Y. Huang, S. Gupta, Z. Song, K. Li, and S. Arora, 'Evaluating Gradient Inversion Attacks and Defenses in Federated Learning', presented at the Advances in Neural Information Processing Systems, Nov. 2021. Accessed: Sep. 09, 2023. [Online]. Available: https://openreview.net/forum?id=0CDKgyYaxC8

[179] J. Li, A. S. Rakin, X. Chen, Z. He, D. Fan, and C. Chakrabarti, 'ResSFL: A Resistance Transfer Framework for Defending Model Inversion Attack in Split Federated Learning', in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans, LA, USA: IEEE, Jun. 2022, pp. 10184–10192. doi: 10.1109/CVPR52688.2022.00995.

[180] J. Li, N. Li, and B. Ribeiro, 'Effective passive membership inference attacks in federated learning against overparameterized models', presented at the The Eleventh International Conference on Learning Representations, Sep. 2022. Accessed: Sep. 09, 2023. [Online]. Available: https://openreview.net/forum?id=QsCSLPP55Ku

[181] S. Dayal, D. Alhadidi, A. Abbasi Tadi, and N. Mohammed, 'Comparative Analysis of Membership Inference Attacks in Federated Learning', in *International Database Engineered Applications Symposium Conference*, Heraklion, Crete Greece: ACM, May 2023, pp. 185–192. doi: 10.1145/3589462.3589502.

[182] A. Suri, P. Kanani, V. J. Marathe, and D. W. Peterson, 'Subject Membership Inference Attacks in Federated Learning', 2022, doi: 10.48550/ARXIV.2206.03317.

[183] F. Elhattab and S. Bouchenak, 'Mitigating Membership Inference Attacks in Federated Learning', in *COMPAS'23 : Conférence francophone en informatique*, Annecy, France, Jul. 2023. Accessed: Sep. 09, 2023. [Online]. Available: https://hal.science/hal-04124475

[184] L. Lyu *et al.*, 'Privacy and Robustness in Federated Learning: Attacks and Defenses', *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–21, 2022, doi: 10.1109/TNNLS.2022.3216981.

[185] N. Bouacida and P. Mohapatra, 'Vulnerabilities in Federated Learning', *IEEE Access*, vol. 9, pp. 63229–63249, 2021, doi: 10.1109/ACCESS.2021.3075203.

[186] P. Danquah and H. Kwabena-Adade, 'Public Key Infrastructure: An Enhanced Validation Framework', *J. Inf. Secur.*, vol. 11, no. 04, pp. 241–260, 2020, doi: 10.4236/jis.2020.114016.

[187] P. Gupta, K. Yadav, B. B. Gupta, M. Alazab, and T. R. Gadekallu, 'A Novel Data Poisoning Attack in Federated Learning based on Inverted Loss Function', *Comput. Secur.*, vol. 130, p. 103270, Jul. 2023, doi: 10.1016/j.cose.2023.103270.

[188] C. Yin and Q. Zeng, 'Defending Against Data Poisoning Attack in Federated Learning With Non-IID Data', *IEEE Trans. Comput. Soc. Syst.*, pp. 1–13, 2023, doi: 10.1109/TCSS.2023.3296885.

[189] V. Shejwalkar and A. Houmansadr, 'Manipulating the Byzantine: Optimizing Model Poisoning Attacks and Defenses for Federated Learning', in *Proceedings 2021 Network and Distributed System Security Symposium*, Virtual: Internet Society, 2021. doi: 10.14722/ndss.2021.24498.

[190] V. Tolpegin, S. Truex, M. E. Gursoy, and L. Liu, 'Data Poisoning Attacks Against Federated Learning Systems', in *Computer Security – ESORICS 2020*, vol. 12308, L. Chen, N. Li, K. Liang, and S. Schneider, Eds., in Lecture Notes in Computer Science, vol. 12308. , Cham:

Springer International Publishing, 2020, pp. 480–501. doi: 10.1007/978-3-030-58951-6_24.

[191] A. K. Nair, E. D. Raj, and J. Sahoo, ‘A robust analysis of adversarial attacks on federated learning environments’, *Comput. Stand. Interfaces*, vol. 86, p. 103723, Aug. 2023, doi: 10.1016/j.csi.2023.103723.

[192] P. Liu, X. Xu, and W. Wang, ‘Threats, attacks and defenses to federated learning: issues, taxonomy and perspectives’, *Cybersecurity*, vol. 5, no. 1, p. 4, Dec. 2022, doi: 10.1186/s42400-021-00105-6.

[193] Z. Li, J. Zhang, L. Liu, and J. Liu, ‘Auditing Privacy Defenses in Federated Learning via Generative Gradient Leakage’, in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans, LA, USA: IEEE, Jun. 2022, pp. 10122–10132. doi: 10.1109/CVPR52688.2022.00989.

[194] ‘Understanding Clipping for Federated Learning: Convergence and Client-Level Differential Privacy | OpenReview’. Accessed: Sep. 09, 2023. [Online]. Available: https://openreview.net/forum?id=zBVjxKB6g84

[195] A. Wainakh *et al.*, ‘Federated Learning Attacks Revisited: A Critical Discussion of Gaps, Assumptions, and Evaluation Setups’, *Sensors*, vol. 23, no. 1, p. 31, Dec. 2022, doi: 10.3390/s23010031.

[196] M. S. Ozdayi, M. Kantarcioglu, and Y. R. Gel, ‘Defending against Backdoors in Federated Learning with Robust Learning Rate’, 2020, doi: 10.48550/ARXIV.2007.03767.

[197] ‘DBA: Distributed Backdoor Attacks against Federated Learning | OpenReview’. Accessed: Sep. 09, 2023. [Online]. Available: https://openreview.net/forum?id=rkgyS0VFvr

[198] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, ‘How To Backdoor Federated Learning’, in *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, PMLR, Jun. 2020, pp. 2938–2948. Accessed: Sep. 09, 2023. [Online]. Available: https://proceedings.mlr.press/v108/bagdasaryan20a.html

[199] M. Du, R. Jia, and D. Song, ‘ROBUST ANOMALY DETECTION AND BACKDOOR AT- TACK DETECTION VIA DIFFERENTIAL PRIVACY’, 2020.

[200] P. Rieger, T. D. Nguyen, M. Miettinen, and A.-R. Sadeghi, ‘DeepSight: Mitigating Backdoor Attacks in Federated Learning Through Deep Model Inspection’, in *Proceedings 2022 Network and Distributed System Security Symposium*, San Diego, CA, USA: Internet Society, 2022. doi: 10.14722/ndss.2022.23156.

[201] C. Wu, X. Yang, S. Zhu, and P. Mitra, ‘Mitigating Backdoor Attacks in Federated Learning’. arXiv, Jan. 14, 2021. doi: 10.48550/arXiv.2011.01767.

[202] Y. Wang, D.-H. Zhai, and Y. Xia, ‘SCFL: Mitigating backdoor attacks in federated learning based on SVD and clustering’, *Comput. Secur.*, vol. 133, p. 103414, Oct. 2023, doi: 10.1016/j.cose.2023.103414.

[203] J. R. Douceur, ‘The Sybil Attack’, in *Peer-to-Peer Systems*, vol. 2429, P. Druschel, F. Kaashoek, and A. Rowstron, Eds., in Lecture Notes in Computer Science, vol. 2429. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 251–260. doi: 10.1007/3-540-45748-8_24.

[204] Y. Jiang, Y. Li, Y. Zhou, and X. Zheng, ‘Sybil Attacks and Defense on Differential Privacy based Federated Learning’, in *2021 IEEE 20th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, Shenyang, China: IEEE, Oct. 2021, pp. 355–362. doi: 10.1109/TrustCom53373.2021.00062.

[205] C. Fung, C. J. M. Yoon, and I. Beschastnikh, 'The Limitations of Federated Learning in Sybil Settings', presented at the 23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2020), 2020, pp. 301–316. Accessed: Sep. 09, 2023. [Online]. Available: https://www.usenix.org/conference/raid2020/presentation/fung

[206] C. Fung, C. J. M. Yoon, and I. Beschastnikh, 'Mitigating Sybils in Federated Learning Poisoning'. arXiv, Jul. 15, 2020. doi: 10.48550/arXiv.1808.04866.