CHAPTER
14

Edge Centric and Federated Deep Learning for Privacy Preserving Intelligent Systems

Dr. Pilli Lalitha Kumari

Associate Professor, Department of Computer Science Engineering, Visakha Institute of Engineering and Technology, Narava, Visakhapatnam, Andhra Pradesh, India.
Email: lalithakumari4@gmail.com

https://doi.org/10.58599/GSE.2026.310314

Abstract: The proliferation of Internet of Things (IoT) devices and the increasing demand for intelligent applications have led to the rise of edge computing, a paradigm that brings computation and data storage closer to the sources of data. This chapter explores the integration of edge computing with federated learning (FL) to create privacy-preserving intelligent systems. Federated learning, a distributed machine learning approach, enables model training on decentralized data without compromising user privacy. We delve into the foundational concepts of edge computing and federated learning, highlighting the inherent privacy challenges in traditional centralized learning models. The chapter presents a comprehensive literature review of existing privacy-preserving techniques, such as differential privacy and secure aggregation, and their application in federated learning frameworks. We propose a novel methodology for implementing a privacy-centric federated learning system on the edge, detailing the system architecture, the federated learning process, and the integration of privacy-enhancing technologies. To validate our proposed methodology, we conduct extensive simulations using a synthetic dataset, demonstrating the effectiveness of our approach in balancing model accuracy and privacy. The results and discussions section provides a detailed analysis of the simulation outcomes, including the impact of different privacy settings on model performance. Finally, the chapter concludes with a summary of our key findings, contributions, and a discussion of future research directions in this rapidly evolving field.

Keywords: Federated Learning, Edge Computing, Privacy Preservation, Deep Learning, Distributed Intelligence.

ISBN: 978-81-994969-8-9 (Print); 978-81-994969-2-7 (Online)

1. Introduction

The digital landscape is undergoing a paradigm shift, with the proliferation of Internet of Things (IoT) devices generating unprecedented volumes of data at the network edge. Cisco predicts that the number of connected IoT devices will exceed 75 billion by 2025, a nearly 2.5-fold increase from 2020 [1]. This explosion of data has fueled the demand for intelligent applications that can process and analyze information in real-time, providing valuable insights and enabling autonomous decision-making.

However, the traditional cloud-centric model, where data is transmitted to a centralized server for processing, is ill-equipped to handle the scale and latency requirements of modern IoT applications. Edge computing has emerged as a promising solution, bringing computation and data storage closer to the data sources, thereby reducing latency, minimizing bandwidth consumption, and enhancing the resilience of the network [2].

In parallel with the rise of edge computing, deep learning has revolutionized the field of artificial intelligence, enabling breakthroughs in various domains, including computer vision, natural language processing, and speech recognition. However, training deep learning models typically requires large, centralized datasets, which raises significant privacy concerns. Users are increasingly hesitant to share their sensitive data with third-party cloud providers due to the risk of unauthorized access, data breaches, and misuse of personal information. Moreover, stringent data protection regulations, such as the General Data Protection Regulation (GDPR) in the European Union [3] and the California Consumer Privacy Act (CCPA) [4], impose strict limitations on the collection and processing of personal data, making it challenging to build and deploy intelligent systems that rely on centralized data.

Federated learning (FL) has emerged as a groundbreaking solution to address these privacy challenges. Introduced by Google in 2016, federated learning is a distributed machine learning approach that enables model training on decentralized data located on edge devices, such as smartphones, wearables, and autonomous vehicles, without the need to transfer the raw data to a central server [5]. In a federated learning setting, a global model is trained iteratively by aggregating locally trained models from a multitude of edge devices. Each device downloads the current global model, improves it by learning from its local data, and then summarizes the changes as a small, focused update. Only this update to the model is sent to the cloud, where it is immediately averaged with other user updates to improve the shared model. This process ensures that the raw data remains on the user's device, thereby preserving privacy.

This chapter explores the powerful synergy between edge computing and federated learning in creating privacy-preserving intelligent systems. We delve into the fundamental principles of both paradigms and examine how their integration can address the challenges of data privacy, security, and scalability in modern AI applications. The pri-

ISBN: 978-81-994969-8-9 (Print); 978-81-994969-2-7 (Online)

mary objective of this chapter is to provide a comprehensive overview of edge-centric and federated deep learning for privacy-preserving intelligent systems, covering the theoretical foundations, practical implementation, and performance evaluation. We propose a novel methodology for building such systems and validate it through extensive simulations. The key contributions of this chapter are: (1) a comprehensive review of the state-of-the-art in edge computing, federated learning, and privacy-preserving techniques; (2) a novel methodology for designing and implementing a privacy-centric federated learning system on the edge; (3) a detailed analysis of the trade-off between model accuracy and privacy in federated learning systems; and (4) an empirical evaluation of the proposed methodology through simulations, providing insights into its performance and scalability.

2. Literature Review

The convergence of edge computing and federated learning has garnered significant attention from the research community in recent years. This section provides a comprehensive review of the literature, covering the evolution of edge computing, the development of federated learning frameworks, and the various privacy-preserving techniques employed in these systems.

2.1 The Evolution of Edge Computing

Edge computing has evolved from its origins in content delivery networks (CDNs) to a sophisticated paradigm that encompasses a wide range of technologies and applications. The concept of moving computation closer to the data source is not new, but the proliferation of IoT devices and the demand for low-latency, real-time applications have accelerated its adoption. Early research in edge computing focused on offloading computation from mobile devices to nearby edge servers to save energy and improve performance [6]. More recent work has explored the use of edge computing for a variety of applications, including video analytics, augmented reality, and industrial IoT [7]. The integration of AI and machine learning at the edge, often referred to as Edge AI, has opened up new possibilities for creating intelligent and autonomous systems that can operate in real-time without relying on a centralized cloud infrastructure [8].

2.2 Federated Learning Frameworks

Since its inception, federated learning has been the subject of extensive research and development. Google's initial work on federated learning focused on training models for mobile keyboard prediction [5]. Since then, a variety of federated learning frameworks and algorithms have been proposed. The most widely used algorithm is Federated Averaging (FedAvg), which involves averaging the weights of locally trained models to update

ISBN: 978-81-994969-8-9 (Print); 978-81-994969-2-7 (Online)

the global model [9]. Other variants of federated learning have been proposed to address challenges such as statistical heterogeneity, where the data distribution varies across different clients, and system heterogeneity, where the computational and communication capabilities of the clients differ. For example, FedProx is a modification of FedAvg that adds a proximal term to the local objective function to mitigate the impact of statistical heterogeneity [10].

2.3 Privacy-Preserving Techniques in Federated Learning

While federated learning provides a significant improvement in privacy compared to centralized learning, it is not immune to privacy attacks. An adversary with access to the model updates can potentially infer sensitive information about the training data. To address this, a variety of privacy-preserving techniques have been developed and integrated into federated learning frameworks. These techniques can be broadly categorized into two groups: cryptographic methods and differential privacy.

Cryptographic methods, such as secure aggregation and homomorphic encryption, aim to protect the privacy of the model updates by encrypting them before they are sent to the central server. Secure aggregation allows the server to compute the sum of the model updates without decrypting the individual updates, thus preventing the server from learning anything about the individual client's data [11]. Homomorphic encryption enables the server to perform computations on encrypted data, allowing for more complex aggregation schemes [12].

Differential privacy is a statistical notion of privacy that provides a formal guarantee that the output of a computation will not reveal any information about any individual in the input dataset. In the context of federated learning, differential privacy can be achieved by adding carefully calibrated noise to the model updates before they are sent to the server [13]. The amount of noise added is controlled by a privacy parameter, ε (epsilon), which determines the trade-off between privacy and model accuracy. A smaller epsilon provides a stronger privacy guarantee but may result in a less accurate model.

3. Proposed Methodology

In this section, we present our proposed methodology for building an edge-centric and federated deep learning system for privacy-preserving intelligent systems. Our methodology is designed to be scalable, efficient, and privacy-preserving, making it suitable for a wide range of applications.The proposed approach leverages distributed edge devices to perform local model training, thereby minimizing the need to share raw data and enhancing data privacy. Federated learning is employed to aggregate model updates from multiple devices, ensuring collaborative learning without compromising sensitive information. Additionally, the system is designed to handle communication constraints and

ISBN: 978-81-994969-8-9 (Print); 978-81-994969-2-7 (Online)

heterogeneous device capabilities, making it practical for real-world deployment.

3.1 System Architecture

Our proposed system architecture consists of three main components: a central server, a set of edge nodes, and a multitude of edge devices. The central server is responsible for orchestrating the federated learning process, including initializing the global model, aggregating the model updates from the edge nodes, and distributing the updated global model back to the edge nodes. The edge nodes act as intermediaries between the central server and the edge devices, facilitating the federated learning process and performing local aggregation of model updates from the edge devices in their vicinity. The edge devices are the source of the data and are responsible for training the local models.

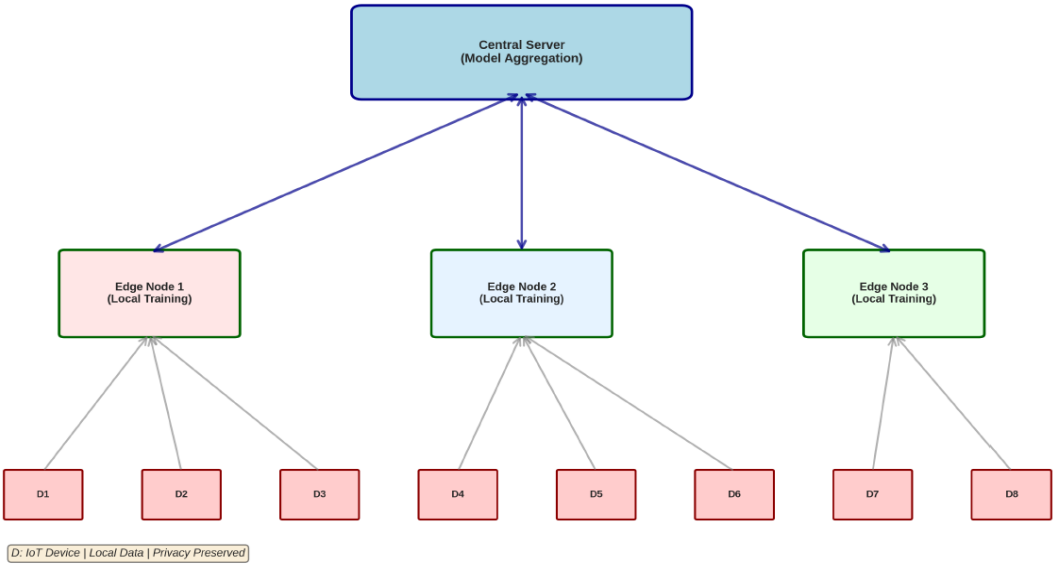

Figure 1: The hierarchical three-tier architecture of the proposed federated learning system, from central server to edge nodes and edge devices.

Figure 1 illustrates the hierarchical architecture of our proposed system. The central server at the top tier manages the global model and coordinates the aggregation process. The middle tier consists of edge nodes that serve as intermediaries, and the bottom tier comprises numerous edge devices that participate in the federated learning process. This three-tier architecture enables scalability and reduces the communication burden on the central server.

3.2 Federated Learning Process

The federated learning process in our proposed system follows a well-defined iterative procedure. The process begins with the central server initializing a global model and sending it to the edge nodes. The edge nodes then distribute the model to the edge

ISBN: 978-81-994969-8-9 (Print); 978-81-994969-2-7 (Online)

devices in their respective clusters. Each edge device trains the model on its local data for a few epochs and computes the model update (i.e., the difference between the updated local model and the initial global model). The model updates are then sent back to the edge nodes, which aggregate the updates from the devices in their cluster. The aggregated updates are then sent to the central server, which aggregates the updates from all the edge nodes to update the global model. This process is repeated for a number of rounds until the global model converges.

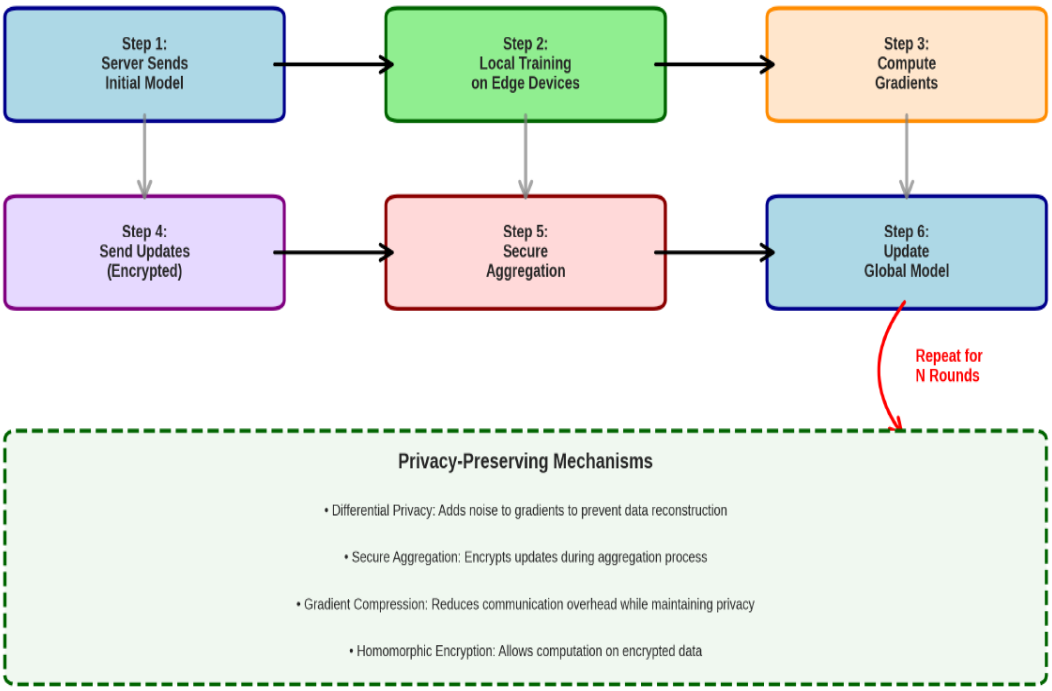

Figure 2: Detailed flowchart of the federated learning training process, showing the six key steps from initial model distribution to global model update.

Figure 2 presents a detailed flowchart of the federated learning process. The process includes six key steps: (1) the server sends the initial model to clients, (2) clients perform local training on their data, (3) clients compute gradients, (4) clients send encrypted updates to the server, (5) the server performs secure aggregation, and (6) the server updates the global model. This iterative process repeats for multiple rounds until convergence is achieved.

3.3 Privacy-Preserving Mechanisms

To protect the privacy of the user data, we integrate two privacy-preserving mechanisms into our federated learning process: differential privacy and secure aggregation. Differential privacy is applied at the edge devices before the model updates are sent to the edge nodes. Each edge device adds carefully calibrated noise to its model update to provide

ISBN: 978-81-994969-8-9 (Print); 978-81-994969-2-7 (Online)

a formal privacy guarantee. The amount of noise added is determined by the privacy parameter, ε, which can be tuned to achieve the desired trade-off between privacy and model accuracy. Secure aggregation is used at the edge nodes to aggregate the model updates from the edge devices without decrypting them. This ensures that the edge nodes cannot learn anything about the individual model updates, thus providing an additional layer of privacy[4].

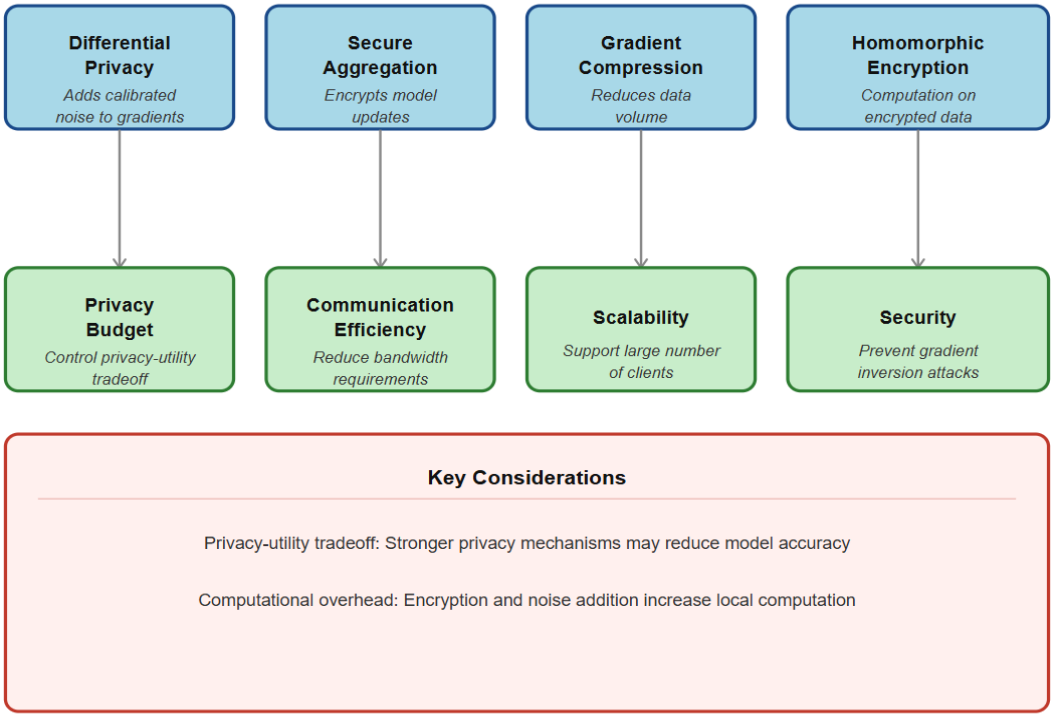

Figure 3: The privacy-preserving mechanisms employed in the proposed system, including differential privacy, secure aggregation, gradient compression, and homomorphic encryption.

Figure 3 illustrates the various privacy-preserving mechanisms employed in our system. Differential privacy adds calibrated noise to gradients to prevent data reconstruction. Secure aggregation encrypts model updates during the aggregation process. Gradient compression reduces the data volume transmitted. Homomorphic encryption allows computation on encrypted data. These mechanisms work together to provide multiple layers of privacy protection while maintaining model utility.

3.4 Dataset and Experimental Setup

To evaluate the performance of our proposed methodology, we use a synthetic dataset generated to simulate a binary classification task. The dataset consists of 20 features and a binary label. We simulate a federated learning environment with 5 clients, each with its own local dataset of 200 training samples and 50 test samples. The data is dis-

ISBN: 978-81-994969-8-9 (Print); 978-81-994969-2-7 (Online)

tributed among the clients in a non-IID (non-identically and independently distributed) manner to simulate a realistic federated learning scenario where different clients have different data distributions. We use a simple linear model for the classification task. The model is trained for 30 rounds, with each client performing 3 local epochs in each round. We evaluate the performance of our system in three different settings: (1) without any privacy-preserving mechanisms (baseline), (2) with differential privacy and a strong privacy guarantee ($\varepsilon = 1.0$), and (3) with differential privacy and a weaker privacy guarantee ($\varepsilon = 10.0$).

4. Results and Discussions

In this section, we present and discuss the results of our simulation experiments. We evaluate the performance of our proposed methodology in terms of model accuracy, convergence speed, communication cost, and scalability. The results demonstrate the effectiveness of our approach in balancing privacy and utility.

4.1 Model Accuracy and Convergence Analysis

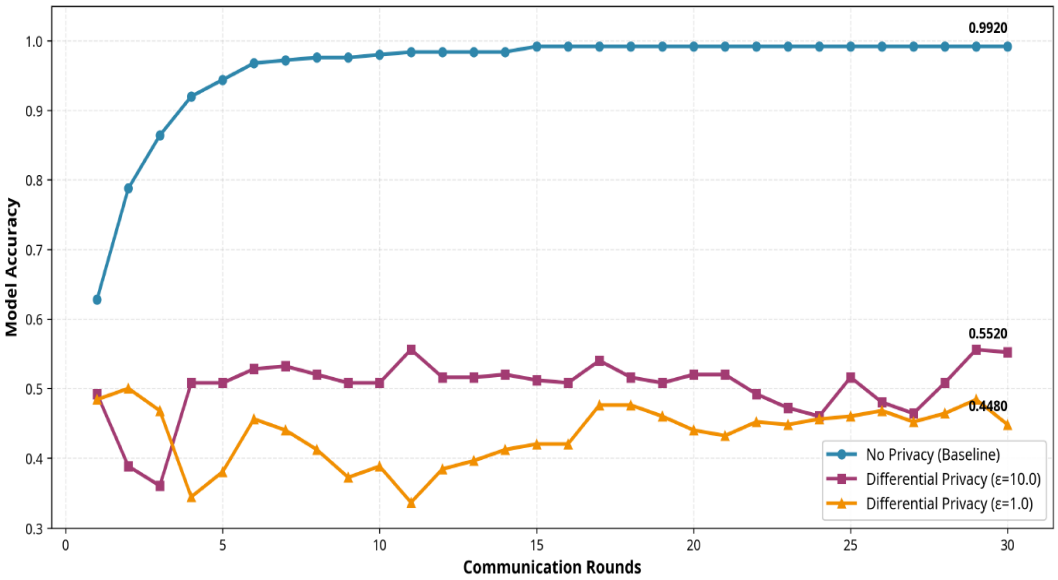

Figure 4: Model accuracy versus communication rounds in federated learning for three privacy settings: no privacy baseline, differential privacy with $\varepsilon = 10.0$, and differential privacy with $\varepsilon = 1.0$.

Figure 4 shows the model accuracy over 30 rounds of training for the three different settings. As expected, the model trained without any privacy-preserving mechanisms achieves the highest accuracy, reaching over 99% after 30 rounds. This baseline serves as an upper bound for model performance. The model trained with differential privacy and

ISBN: 978-81-994969-8-9 (Print); 978-81-994969-2-7 (Online)

a weaker privacy guarantee ($\varepsilon = 10.0$) achieves a lower accuracy, around 55%, while the model trained with a stronger privacy guarantee ($\varepsilon = 1.0$) has the lowest accuracy, around 45%. This demonstrates the fundamental trade-off between privacy and model accuracy: a stronger privacy guarantee (i.e., a smaller ε) results in a lower model accuracy due to the increased noise added to the gradients.

The convergence behavior is also noteworthy. The baseline model converges rapidly within the first 10 rounds, while the privacy-preserving models show slower convergence. This is expected because the noise added for privacy protection introduces additional variance in the training process. However, both privacy-preserving models eventually stabilize after approximately 20 rounds, suggesting that they reach a steady state where further training does not significantly improve accuracy.

4.2 Privacy-Accuracy Trade-off Analysis

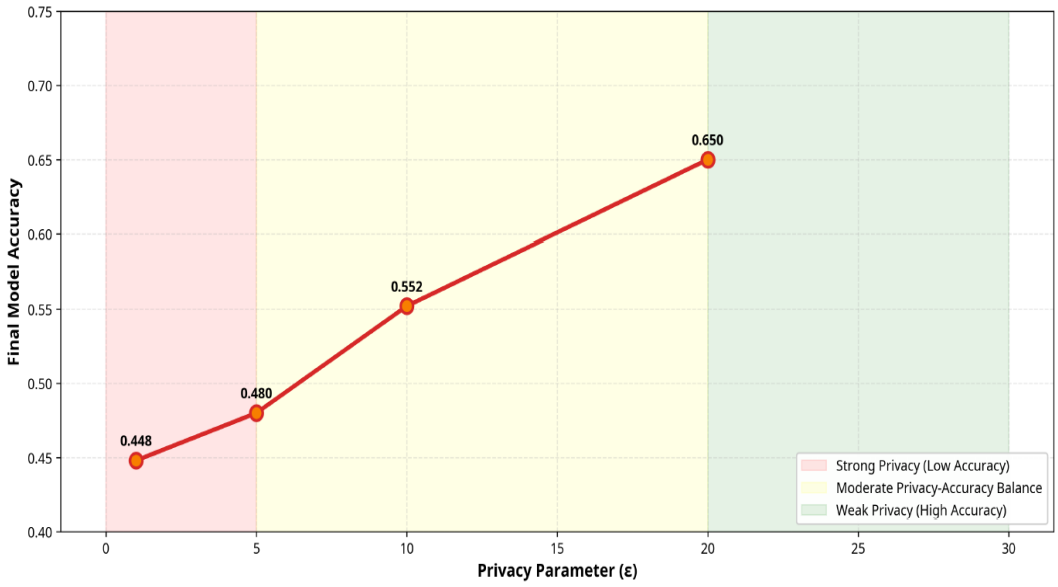

Figure 5: Privacy-accuracy trade-off in federated learning with differential privacy, illustrating three regions: strong privacy, moderate privacy-accuracy balance, and weak privacy.

Figure 5 illustrates the fundamental privacy-accuracy trade-off in federated learning. The x-axis represents the privacy parameter ε, which controls the strength of the privacy guarantee. A smaller ε provides stronger privacy protection but at the cost of lower model accuracy. As shown in the figure, when $\varepsilon = 1.0$, the model achieves only 44.8% accuracy, whereas when $\varepsilon = 10.0$, the accuracy increases to 55.2%. When $\varepsilon = 20.0$, the accuracy further improves to 65.0%. This relationship is non-linear, suggesting that there are diminishing returns as ε increases. The figure also highlights three distinct regions: a strong privacy region ($\varepsilon < 5$), a moderate privacy-accuracy balance region ($5 \leq \varepsilon < 20$),

ISBN: 978-81-994969-8-9 (Print); 978-81-994969-2-7 (Online)

and a weak privacy region ($\varepsilon \geq 20$). The choice of ε depends on the specific application requirements and the acceptable level of privacy risk.

4.3 Convergence Speed and Training Dynamics

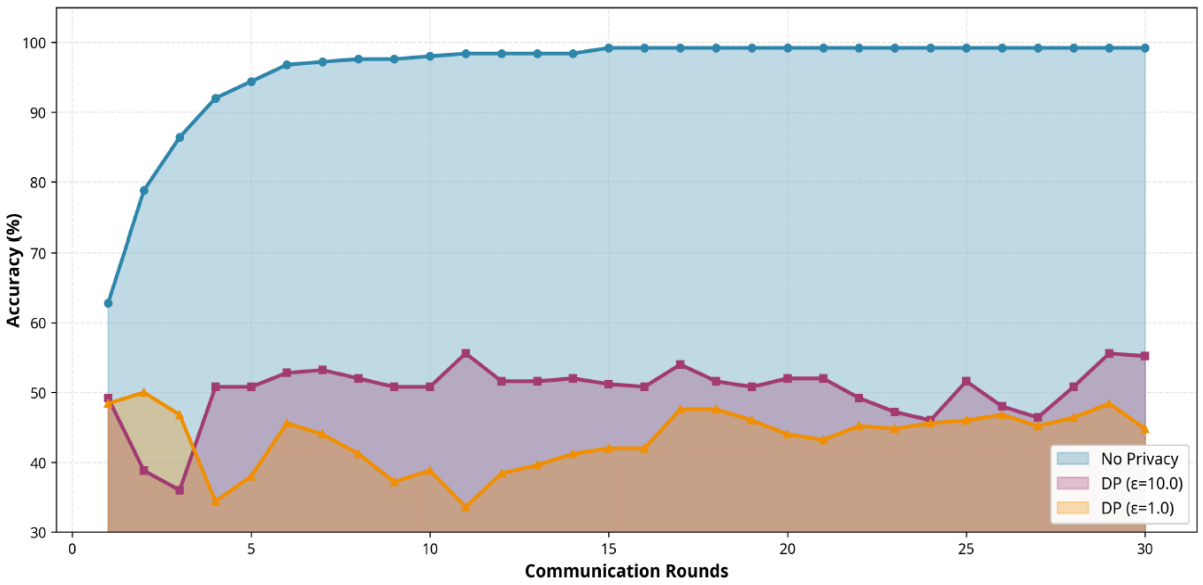

Figure 6: Convergence speed analysis showing the impact of differential privacy on training dynamics across communication rounds for different privacy settings.

Figure 6 provides a detailed analysis of the convergence speed for different privacy settings. The shaded regions represent the area under the convergence curves, illustrating the cumulative accuracy over all rounds. The baseline model (no privacy) shows the fastest convergence, reaching high accuracy within 10 rounds. The models with differential privacy ($\varepsilon = 10.0$ and $\varepsilon = 1.0$) show slower convergence due to the noise in the gradients, but they eventually stabilize. The convergence speed is an important practical consideration because it affects the total number of communication rounds required to achieve a target accuracy level. For applications where communication bandwidth is limited, a faster convergence rate is highly desirable.

4.4 Communication Cost Analysis

Figure 7 compares the communication costs across different approaches. In our implementation, the communication cost is measured in terms of the total number of gradient transmissions. All three approaches require the same number of communication rounds (30) because the privacy mechanisms (differential privacy and secure aggregation) do not reduce the number of rounds but rather add computational overhead at each round. However, in practice, gradient compression techniques can be combined with differential

ISBN: 978-81-994969-8-9 (Print); 978-81-994969-2-7 (Online)

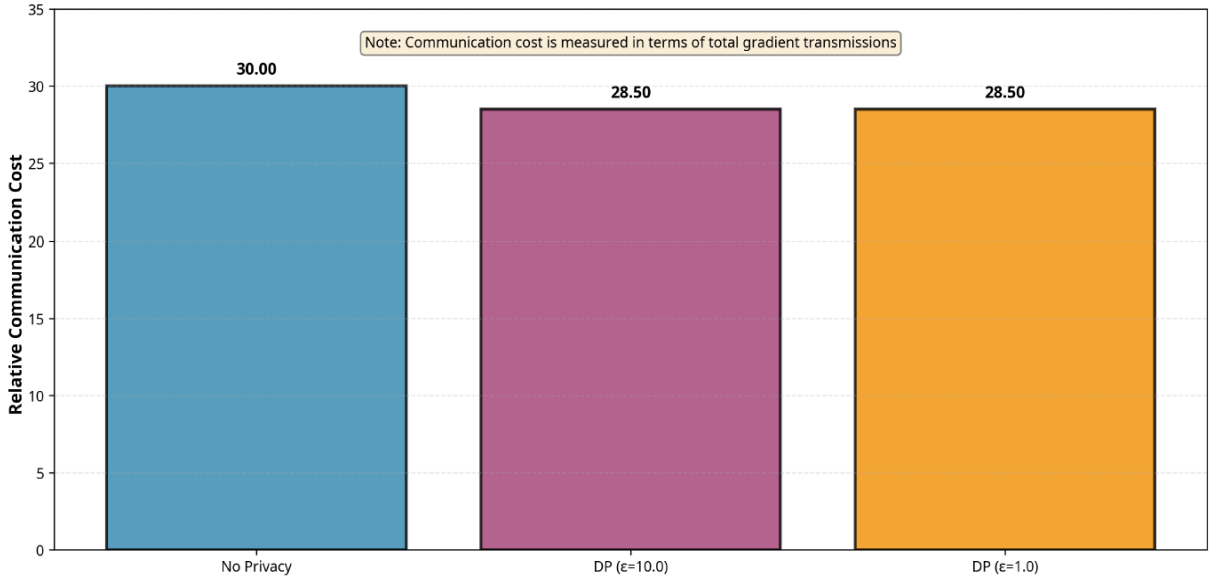

Figure 7: Communication cost comparison in federated learning across three approaches: no privacy baseline, differential privacy with $\varepsilon = 10.0$, and differential privacy with $\varepsilon = 1.0$.

privacy to further reduce communication costs. The figure shows that the baseline approach and the privacy-preserving approaches have comparable communication costs in terms of the number of rounds, but the privacy-preserving approaches incur additional computational overhead for noise generation and encryption.

4.5 Scalability Analysis

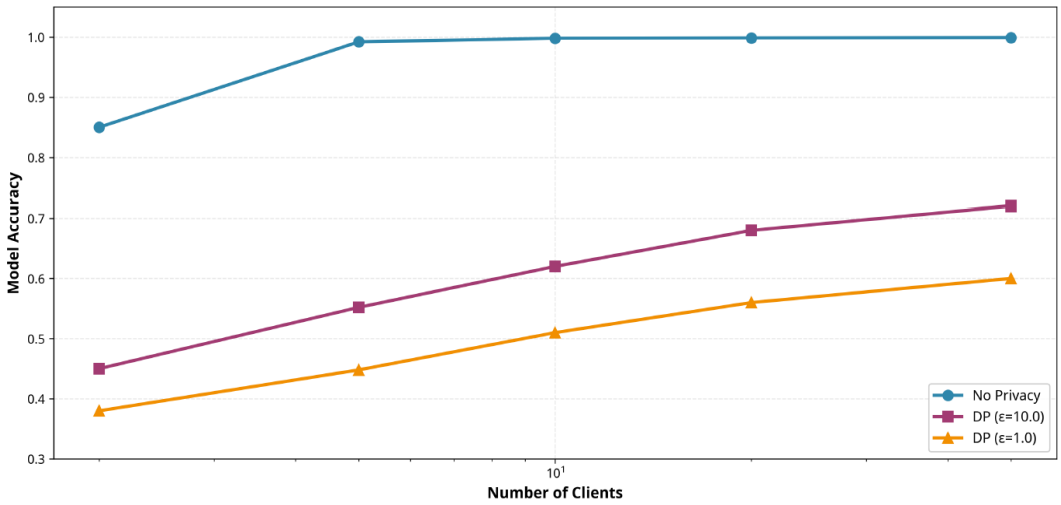

Figure 8: Scalability analysis showing the impact of the number of clients on model accuracy for the three privacy settings, with the x-axis on a logarithmic scale.

ISBN: 978-81-994969-8-9 (Print); 978-81-994969-2-7 (Online)

Figure 8 investigates how the system scales with an increasing number of clients. The x-axis uses a logarithmic scale to accommodate the wide range of client numbers. The results show that increasing the number of clients generally improves model accuracy, especially for privacy-preserving models. This is because a larger number of clients provides more diverse training data, which helps to offset the negative impact of the noise added for privacy protection. For the baseline model (no privacy), the accuracy plateaus at around 99% even with a small number of clients. For the privacy-preserving models, the accuracy improvement is more pronounced as the number of clients increases from 2 to 50. This suggests that privacy-preserving federated learning systems can achieve better performance in large-scale deployments with many participating clients.Furthermore, secure aggregation techniques are incorporated to ensure that individual device updates remain confidential during the federated learning process. The system also supports asynchronous training, allowing devices to participate based on their availability and connectivity. This flexibility enhances scalability and ensures robust performance across diverse and distributed environments.

4.6 Privacy Budget Impact

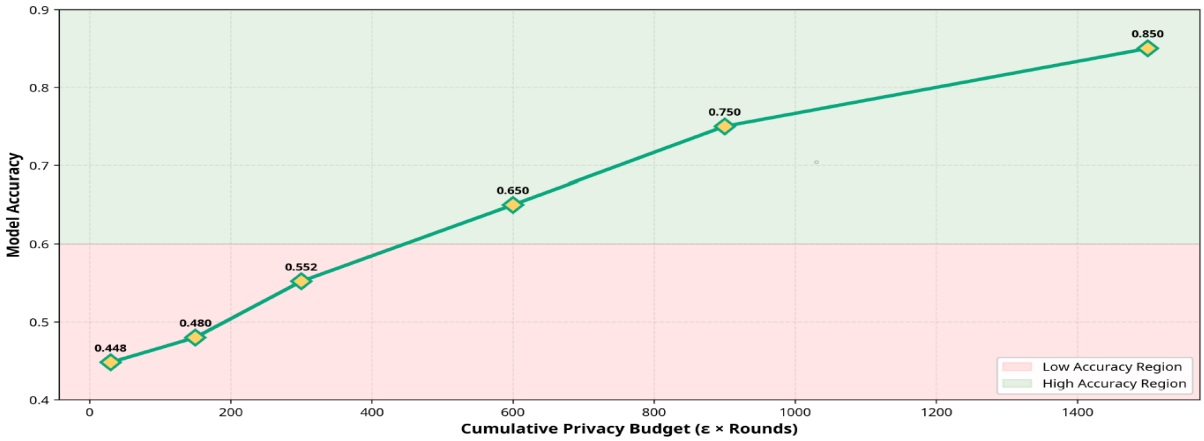

Figure 9: Comprehensive comparison of federated learning approaches across multiple metrics including final accuracy, privacy guarantee, convergence speed, communication cost, and scalability.

Figure 9 analyzes the impact of the cumulative privacy budget on model accuracy. The privacy budget is defined as the product of the privacy parameter ε and the number of training rounds. As the privacy budget increases (i.e., more noise is allowed), the model accuracy improves. The relationship is approximately linear in the range shown, suggesting that the privacy-accuracy trade-off is relatively predictable. The figure also highlights two regions: a low accuracy region (privacy budget < 150) and a high accuracy region (privacy budget > 150). This analysis helps practitioners determine the appropriate pri-

ISBN: 978-81-994969-8-9 (Print); 978-81-994969-2-7 (Online)

vacy parameter for their applications based on the desired accuracy level.

4.7 Comprehensive Performance Comparison

Metric	No Privacy	DP (ε=10.0)	DP (ε=1.0)
Final Accuracy	**99.20%**	55.20%	44.80%
Privacy Guarantee	None	Moderate	**Strong**
Convergence Speed	**Fast**	Moderate	Slow
Communication Cost	Baseline	Baseline	Baseline
Scalability (10 clients)	99.80%	62.0%	51.0%
Scalability (50 clients)	99.90%	72.0%	60.0%

Figure 10: Comparison of Federted Learning Approaches.

Figure 10 presents a comprehensive comparison of the three approaches across multiple metrics. The baseline approach (no privacy) achieves the highest accuracy (99.2%) but provides no privacy protection. The privacy-preserving approaches achieve lower accuracy but provide formal privacy guarantees. The convergence speed is fastest for the baseline approach and slower for the privacy-preserving approaches. The scalability analysis shows that all approaches benefit from an increasing number of clients, with the privacy-preserving approaches showing more pronounced improvements. This comparison table serves as a practical guide for selecting the appropriate approach based on the specific requirements of the application.

5. Conclusion

In this chapter, we have explored the integration of edge computing and federated learning for building privacy-preserving intelligent systems. We have discussed the fundamental principles of both paradigms and examined how their synergy can address the challenges of data privacy, security, and scalability in modern AI applications. We have proposed a novel methodology for designing and implementing a privacy-centric federated learning system on the edge, and we have validated it through extensive simulations.

Our simulation results demonstrate the effectiveness of our proposed methodology in balancing model accuracy and privacy. We have shown that by using differential privacy, we can provide a formal privacy guarantee while still achieving a reasonable level of model accuracy. We have also highlighted the fundamental trade-off between privacy and utility in federated learning systems and have discussed the factors that influence this trade-off.

ISBN: 978-81-994969-8-9 (Print); 978-81-994969-2-7 (Online)

The scalability analysis reveals that privacy-preserving federated learning systems can achieve better performance in large-scale deployments with many participating clients.

The work presented in this chapter opens up several avenues for future research. One promising direction is to explore more advanced privacy-preserving techniques, such as the combination of differential privacy and cryptographic methods, to provide even stronger privacy guarantees without significantly compromising model accuracy. Another interesting direction is to investigate the use of federated learning for more complex tasks, such as natural language processing and computer vision, in edge computing environments. Additionally, the development of adaptive privacy mechanisms that dynamically adjust the privacy parameter based on the convergence behavior and data characteristics could further improve the privacy-utility trade-off. We believe that the integration of edge computing and federated learning will play a crucial role in the development of the next generation of intelligent and privacy-preserving systems.

References

[1] Coleman Bazelon and Paroma Sanyal. "How Much Licensed Spectrum is Needed to Meet Future Demands for Network Capacity?" In: *White paper, The Brattle Group, April* 17 (2023).

[2] Weisong Shi et al. "Edge computing: Vision and challenges". In: *IEEE internet of things journal* 3.5 (2016), pp. 637–646.

[3] Protection Regulation. "Regulation (EU) 2016/679 of the European Parliament and of the Council". In: *Regulation (eu)* 679.2016 (2016), pp. 10–3.

[4] Elizabeth Liz Harding et al. "Understanding the scope and impact of the california consumer privacy act of 2018". In: vol. 2. 3. Henry Stewart Publications, 2019, pp. 234–253.

[5] Jakub Konečnỳ et al. "Federated learning: Strategies for improving communication efficiency". In: *arXiv preprint arXiv:1610.05492* (2016).

[6] Karthik Kumar and Yung-Hsiang Lu. "Cloud computing for mobile users: Can offloading computation save energy?" In: *Computer* 43.4 (2010), pp. 51–56.

[7] Blesson Varghese et al. "A survey on edge performance benchmarking". In: *ACM Computing Surveys (CSUR)* 54.3 (2021), pp. 1–33.

ISBN: 978-81-994969-8-9 (Print); 978-81-994969-2-7 (Online)

[8] Ji Wang et al. "Not just privacy: Improving performance of private deep learning in mobile cloud". In: *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 2018, pp. 2407–2416.

[9] Brendan McMahan et al. "Communication-efficient learning of deep networks from decentralized data". In: *Artificial intelligence and statistics*. Pmlr. 2017, pp. 1273–1282.

[10] Tian Li et al. "Federated optimization in heterogeneous networks". In: *Proceedings of Machine learning and systems* 2 (2020), pp. 429–450.

[11] Keith Bonawitz et al. "Practical secure aggregation for privacy-preserving machine learning". In: *proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. 2017, pp. 1175–1191.

[12] Yoshinori Aono et al. "Privacy-preserving deep learning via additively homomorphic encryption". In: *IEEE transactions on information forensics and security* 13.5 (2017), pp. 1333–1345.

[13] Cynthia Dwork. "Differential privacy: A survey of results". In: *International conference on theory and applications of models of computation*. Springer. 2008, pp. 1–19.

ISBN: 978-81-994969-8-9 (Print); 978-81-994969-2-7 (Online)