International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 05 | May - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Scalable Streaming Data at the Egde Using Deep Learning

P. Kamakshi Thai[1], B. Aishwarya[2], Thakur Anirudh[3], Mohammad Saniya[4]
[1]Assistant Professor of Department of CSE (AI & ML) of ACE Engineering College, India.
[2,3,4] Students of Department CSE (AI & ML) Of ACE Engineering College, India.

ABSTRACT

In recent years, the proliferation of IoT devices and edge sensors has led to an exponential increase in real-time streaming data, challenging conventional deep learning methods designed for static, centralized datasets. ScaDLES, addresses these challenges by introducing a framework for training deep learning models directly on edge devices. The primary objective is to overcome issues such as heterogeneous streaming rates, limited memory and storage, and high communication overhead typical in distributed environments. The system is implemented using Python with PyTorch for model training, Apache Kafka for simulating streaming data, and Docker containers to emulate a distributed edge environment. it indicates that ScaDLES significantly enhances training speed and reduces resource consumption compared to traditional distributed deep learning frameworks. It's not only improves the feasibility of real-time model updates on the edge but also paves the way for applications in surveillance, autonomous vehicles, and smart cities.

Keywords:

Edge Computing, Streaming Data, Deep Learning, Distributed Training Gradient- Compression.

INTRODUCTION

Edge computing has become increasingly vital in the era of IoT and real-time data processing. Traditional deep learning training methods rely on cloud infrastructure with homogeneous compute resources and high-speed internet, making them unsuitable for edge environments. Edge devices typically suffer from limited computing power, inconsistent bandwidth, and receive data in a non-identical and non-independent (non-IID) fashion. The ScaDLES project addresses these challenges by proposing an adaptive, resource-efficient distributed deep learning approach for edge streaming data.

BACKGROUND OF THE PROJECT

Conventional distributed deep learning frameworks like those used in cloud and data center environments assume powerful and uniform hardware, stable network conditions, and balanced data distribution across devices. These assumptions often fail in edge computing environments where devices are resource-constrained, connected via unstable networks, and collect data that varies in type, volume, and distribution. Additionally, many edge devices receive data as continuous streams, introducing variability in data arrival rates. In such cases, traditional training approaches suffer from inefficiencies like communication bottlenecks, model staleness, and poor accuracy due to skewed data. ScaDLES (Scalable Distributed Learning for Edge Streams) is designed to overcome these limitations by incorporating asynchronous training, adaptive aggregation, and communication-efficient learning techniques. It dynamically adjusts to the constraints and heterogeneity of edge systems to enable effective and scalable training directly on the edge.

LITERATURE REVIEW

1. **Kairouz et al. (2019) :**

   This paper provides a comprehensive overview of federated learning, discussing its core principles, challenges, and applications. It highlights key research problems such as data heterogeneity, system scalability, and privacy preservation. While highly informative, the paper primarily focuses on theoretical and conceptual analysis without presenting new empirical methods, limiting its practical contribution for edge-specific implementations.

2. Li et al. (2020) :

This paper proposes FedProx, an extension of federated averaging that introduces a proximal term to mitigate the effects of system heterogeneity. The algorithm stabilizes training across devices with differing capabilities and data distributions. However, the study does not consider asynchronous training or streaming data scenarios, which are critical for edge environments addressed by ScaDLES.

3. Zhao et al. (2018) :

This paper investigates the performance of federated learning under non-IID data distributions. It demonstrates that non-IID data significantly degrades model accuracy, and suggests strategies such as shared data initialization. However, it does not offer adaptive aggregation mechanisms like those used in ScaDLES to deal with continuous non-IID streaming data.

4. Lin et al. (2020):

This study presents Deep Gradient Compression to reduce the communication burden in federated learning through gradient sparsification and quantization. The proposed method shows substantial bandwidth savings with minimal accuracy loss. Nevertheless, it focuses solely on synchronous updates and lacks consideration of adaptive techniques for fluctuating edge bandwidth conditions.

5. Xie et al. (2019) :

This paper introduces asynchronous federated optimization, which allows devices to update models independently, reducing synchronization overhead. It confirms that asynchrony improves training efficiency without major accuracy loss. However, the model aggregation method assumes stable network conditions and does not integrate streaming data variation.

6. Bonawitz et al. (2019) :

This paper describes a real-world implementation of federated learning at Google scale, tackling issues like device availability, privacy, and system robustness. While practical and impactful, it does not explore advanced aggregation techniques or adaptation to real-time streaming data, which ScaDLES focuses on.

7. Wang et al. (2020) :

The authors propose adaptive federated optimization strategies that adjust learning parameters based on local conditions to enhance accuracy and reduce communication. Although this work introduces adaptivity, it is primarily aimed at static datasets and does not extend to dynamic data rates or asynchronous updates.

8. Rieke et al. (2020) :

This paper explores federated learning in healthcare applications, emphasizing data privacy and communication efficiency. It validates the applicability of FL in sensitive domains. However, it lacks algorithmic innovations and does not address heterogeneity or streaming data, both critical to edge environments.

9. Kang et al. (2019) :

This study analyzes incentive mechanisms to encourage device participation in federated learning. It provides models for fair and rational contribution assessment. While valuable for managing participation, it does not address learning performance, communication efficiency, or real-time training adaptability.

10. Haddadpour et al. (2021) :

This paper proposes local SGD with periodic averaging to reduce synchronization needs and communication cost. The method achieves near-optimal convergence with fewer updates. Still, it assumes batch training from static datasets and lacks integration with real-time streaming.

11. Yurochkin et al. (2019) :

The authors introduce a Bayesian nonparametric method for federated learning, enabling client-specific model personalization. This supports better accuracy on heterogeneous data. However, the method is computationally expensive and not optimized for resource-limited edge scenarios.

12. Jeong et al. (2018) :

This study presents federated distillation, where devices share model outputs instead of gradients or weights. This reduces bandwidth usage and enhances privacy. However, it is less effective when data distributions are highly skewed, and it lacks adaptability to streaming rates.

13. Wang et al. (2021) :

The authors propose partial model training, where only a subset of the model is trained at a time. This reduces resource consumption, making it suitable for edge devices. Yet, it does not deal with asynchronous updates or adaptivity to real-time data streams.

14. Sattler et al. (2019) :

This paper introduces sparse binary communication techniques for federated learning, reducing update size significantly. It demonstrates high communication efficiency, although its applicability to streaming, asynchronous setups like ScaDLES remains unexplored**.**

COMPARISION TABLE

S. No	Author(s)	Title	Methodology	Key Contribution
1	Kairouz et al. (2019)	Advances in Federated Learning	Overview of FL principles and challenges	Establishes FL foundation but assumes homogeneous conditions
2	Li et al. (2020)	FedProx	Adds proximal term to address heterogeneity	Improves training under resource variability
3	Zhao et al. (2018)	Non-IID Data in FL	Studies effect of data skew	Highlights non-IID impact on accuracy
4	Lin et al. (2020)	Deep Gradient Compression	Uses gradient sparsification	Reduces communication overhead
5	Xie et al. (2019)	Asynchronous FedOpt	Asynchronous optimization in FL	Reduces sync delays

6	Bonawitz et al. (2019)	Google FL System	Production-level FL implementation	Addresses practical deployment issues
7	Wang et al. (2020)	Adaptive FL	Dynamically adjusts learning rate	Enhances accuracy and stability
8	Rieke et al. (2020)	FL in Healthcare	FL applications for sensitive data	Focuses on privacy and efficiency
9	Kang et al. (2019)	Incentive Mechanisms in FL	Incentive-aware FL participation	Promotes fairness in updates
10	Haddadpour et al. (2021)	Local SGD with Averaging	Periodic model averaging in FL	Balances performance and cost
11	Yurochkin et al. (2019)	Bayesian FL	Model personalization	Enables heterogeneous model adaptation
12	McMahan et al. (2017)	FedAvg	Federated averaging algorithm	Fundamental to FL, baseline for ScaDLES
13	Jeong et al. (2018)	Federated Distillation	Sends model outputs instead of parameters	Reduces communication load
14	Wang et al. (2021)	Partial Model Training	Trains part of the model	Efficient use of limited device resources
15	Sattler et al. (2019)	Sparse Binary Communication	Uses sparse and binary updates	Maximizes bandwidth efficiency

Research Gaps Addressed by ScaDLES:

Handling Heterogeneity in Streaming Rates Across Edge Devices

- Existing federated learning approaches often assume periodic and batch-style data availability, which doesn't hold true in edge environments where data may arrive at irregular and unpredictable rates. ScaDLES introduces dynamic batching and adaptive scheduling mechanisms that allow devices to process and transmit model updates based on their individual data arrival patterns, minimizing idle times and maximizing utilization of edge resources.

Improving Model Accuracy Under Non-IID Data Conditions

- Traditional FL models trained on non-IID data tend to converge slowly or to suboptimal solutions due to statistical divergence between local and global objectives. ScaDLES addresses this with weighted and adaptive aggregation strategies that account for the quality and quantity of each device's contributions, reducing bias and improving convergence stability.

Reducing Training Delays Without Relying on Global Synchronization

- Many distributed learning techniques suffer from the "straggler problem," where slower devices delay global updates. ScaDLES leverages asynchronous training and communication protocols that allow devices to train and share updates independently, preventing bottlenecks and significantly accelerating training in highly heterogeneous systems.

Optimizing Communication for Bandwidth-Constrained Environments

- Communication is a major bottleneck in edge learning due to limited and unreliable network links. ScaDLES incorporates gradient compression, sparsification, and model distillation techniques to reduce update sizes without sacrificing model performance. It also selectively prioritizes significant updates.

PROPOSED SYSTEM

To support real-time data processing, we utilize Apache Kafka to simulate continuous streaming input. This allows the model to learn and adapt on the fly, rather than relying on large, pre-collected static datasets. Model training is performed using PyTorch, chosen for its flexibility and support for dynamic computation graphs, which are ideal for streaming and online learning scenarios. The training process is conducted in an online manner, meaning the model continuously updates itself as new data arrives, improving its performance over time.To reduce communication overhead between distributed edge nodes and the central coordinator, we implement gradient compression techniques. These ensure that only essential information is transmitted, thereby saving bandwidth and reducing latency.We simulate a realistic distributed edge environment using Docker containers, each representing a separate edge device. This setup allows us to effectively test scalability, system heterogeneity, and synchronization behavior in a controlled setting.

CONCLUSION AND FUTURE SCOPE

The reviewed literature provides a solid foundation for decentralized deep learning; however, many existing solutions rely on idealized conditions such as synchronous training, uniform data distribution, and stable resource availability. These assumptions are rarely valid in real-world edge environments. In contrast, ScaDLES introduces a practical and adaptive framework specifically designed to address the unique challenges of edge computing—such as real-time data streaming, resource heterogeneity, and non-IID data distribution—making it a significant step forward in scalable deep learning at the edge.

REFERENCES

1. Kairouz, P. et al. (2019). Advances and Open Problems in Federated Learning. arXiv:1912.04977.
2. Li, T. et al. (2020). Federated Optimization in Heterogeneous Networks. MLSys.
3. Zhao, Y. et al. (2018). Federated Learning with Non-IID Data. arXiv:1806.00582.
4. Lin, Y. et al. (2020). Deep Gradient Compression. arXiv:1712.01887.
5. Xie, C. et al. (2019). Asynchronous Federated Optimization. arXiv:1903.03934.
6. Bonawitz, K. et al. (2019). Towards Federated Learning at Scale: System Design. SysML.
7. Wang, S. et al. (2020). Tackling the Objective Inconsistency Problem in Heterogeneous FL. NeurIPS.
8. Rieke, N. et al. (2020). The Future of Digital Health with Federated Learning. npj Digital Medicine.
9. Kang, J. et al. (2019). Incentive Mechanism for Reliable Federated Learning. IEEE Communications Magazine.
10. Haddadpour, F. et al. (2021). Local SGD with Periodic Averaging. AISTATS.
11. Yurochkin, M. et al. (2019). Bayesian Nonparametric Federated Learning of Neural Networks. ICML.

Volume: 09 Issue: 05 | May - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

12. McMahan, H. B. et al. (2017). Communication-Efficient Learning of Deep Networks from Decentralized Data. AISTATS.
13. Jeong, E. et al. (2018). Communication-Efficient On-Device Machine Learning. NIPS Workshop.
14. Wang, H. et al. (2021). Federated Learning with Partial Model Training. arXiv:2102.07078.
15. Sattler, F. et al. (2019). Robust and Communication-Efficient Federated Learning from Non-IID Data. IEEE Trans. on Neural Networks and Learning Systems.