CS6993 : Research Project

A Federated Learning based weather monitoring system

Rik Halder
223CS3148
Master Of Technology
223CS3148@nitrkl.ac.in

Department of Computer Science and Engineering
NIT Rourkela

under the supervision of
Prof. Suchismita Chinara
December 02, 2024

Certificate of Examination

Roll Number: 223CS3148
Name: Rik Halder
Title of Dissertation:A Federated Learning based weather monitoring system.

This is to certify that the research project report titled **"A Federated Learning based weather monitoring system"** by **Rik Halder** has been examined and approved by the undersigned.

Prof. Suchismita Chinara
Principal Supervisor
Date: December 02, 2024

Prof. Bibhudatta Sahoo
Head of the Department
Date: December 02, 2024

Supervisor’s Certificate

This is to certify that the work presented in the report entitled “A Federated Learning based weather monitoring system” submitted by Rik Halder , Roll Number 223CS3148, is a record of original research carried out by him under my supervision and guidance in partial fulfillment of the requirements of the degree of Master of Technology in Computer Science and Engineering. Neither this thesis nor any part of it has been submitted earlier for any degree or diploma to any institute or university in India or abroad.

(Prof. Suchismita Chinara)
Date: December 02, 2024

Declaration of Originality

I, Rik Halder, Roll Number 222CS3148 hereby declare that this report entitled "A Federated Learning based weather monitoring system" presents my original work carried out as a postgraduate student of NIT Rourkela and, to the best of my knowledge, contains no material previously published or written by another person, nor any material presented by me for the award of any degree or diploma of NIT Rourkela or any other institution. Any contribution made to this research by others, with whom I have worked at NIT Rourkela or elsewhere, is explicitly acknowledged in the dissertation. Works of other authors cited in this dissertation have been duly acknowledged under the sections "Reference".

I am fully aware that in case of any non-compliance detected in future, the Senate of NIT Rourkela may withdraw the degree awarded to me on the basis of the present dissertation.

December 02, 2024 Rik Halder
NIT Rourkela

Acknowledgment

I would like to take this opportunity to express sense of gratitude and indebtedness to Prof. Suchismita Chinara, Professor of Department of Computer Science and Engineering for having me as her student and providing me opportunity to pursue research project that led to this dissertation. Throughout the endeavour, she offered me tremendous support and insightful advice on a variety of issues. I would like to thank the esteemed Prof. Bibhudatta Sahoo, Head of the Department of Computer Science and Engineering, for his cordial cooperation.

Additionally, I would like to express my gratitude to Vipul Singh Negi, for his insightful advice and direction throughout the project.

Above all, I want to express my gratitude to my parents for their unwavering support as I've developed my career. Finally, I want to thank my fellow classmates for making my experience at NIT enjoyable.

Abstract

Accurate weather forecasting is important for many application domains, such as agriculture, disaster management, and urban planning. However, traditional centralized systems have several drawbacks in terms of data granularity being limited, regional classification being limited, and a very few weather parameters that can be observed, along with the issues of privacy and scalability. This work outlines a Federated Learning-based Weather Monitoring System to overcome these deficiencies. Using the Meteostat API, a custom high-resolution weather dataset was developed to include data at an hourly level for 36 stations across India, including temperature, humidity, wind speed, and precipitation parameters.The Federated Learning algorithms, FedAvg and FedProx, are used to train machine learning models—namely Multi-Layer Perceptrons (MLP), Long Short-Term Memory networks (LSTM), and Transformers—collaboratively across geographically distributed clients. The global and local accuracy, loss metrics, and prediction performance of the models are compared. The MLP model trained with FedAvg is found to be the optimal configuration with high prediction accuracy within ±5% tolerance and low error rates.This work proves the feasibility of decentralized, privacy-preserving weather forecasting systems that manage heterogeneous and distributed data. Results show that Federated Learning is a viable approach to filling the gap between modern machine learning methods and practical meteorological applications, providing scalable, localized, and robust solutions for weather monitoring and prediction.

Contents

List of Figures

List of Tables

1 Introduction

Accurate weather monitoring and forecasting are very important for various applications, including agriculture, disaster management, energy optimization, and urban planning. However, the traditional centralized weather prediction systems face serious challenges, particularly in terms of data privacy, heterogeneity, and scalability. Besides, the existing datasets are mainly from national meteorological departments, which are very limited due to lack of granularity, poor regional coverage, and limited scope of weather parameters. These issues create limitations for machine learning models which feed on diverse, high-quality data.

To overcome these challenges, this research proposes a Federated Learning-based Weather Monitoring System. Federated Learning (FL) is a decentralized approach to machine learning that allows multiple data sources to collaboratively train models without sharing raw data. This approach ensures data privacy and enables scalability while addressing the inherent non-i.i.d. nature of distributed data. Such capabilities make FL particularly well-suited for weather prediction tasks, which involve data collected from geographically dispersed weather stations.

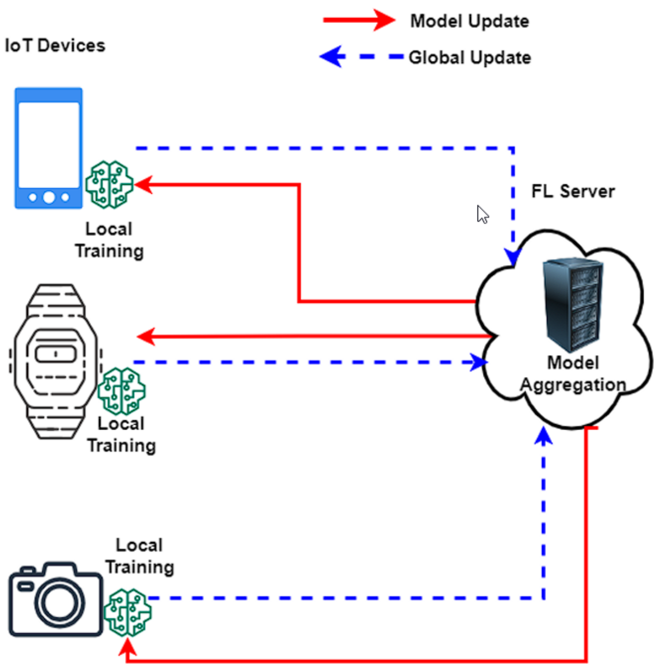


Figure 1: Basic IoT Federated Learning Architecture[1]

The study addresses the limitations of existing systems by using the Meteostat API to create a custom weather dataset with an hourly data set from 36 weather stations across India. Unlike conventional datasets, this dataset offers a comprehensive range of weather parameters, such as temperature, humidity, wind speed, and precipitation, providing the granularity and diversity required for robust machine learning models. Moreover, the dataset supports regional and seasonal classifications, enabling localized weather forecasting.

This research also proposes a Federated Learning framework for weather forecasting, using FL algorithms such as FedAvg and FedProx to enable collaborative training of models over distributed clients. The optimal combination of model and FL algorithm for accurate and efficient weather forecasting is identified from the performance of various machine learning models, including MLPs, LSTMs, and Transformers. The experimental results show that Federated Learning is not only privacy-preserving but also scalable and even achieves strong performance with heterogeneous data distributions.

Besides its contribution to the new way of weather forecasting, this work further indicates a much deeper impact of decentralized systems toward real-world applications. In the integration of FL to weather monitoring, there emerges a scalable, privacy-preserving, and localized approach to the traditional centralized systems while bridging the gap that exists between modern machine learning techniques and practical deployment in meteorology.

This work is structured as follows: Section 2 describes related work on Federated Learning simulators, strategies, and weather prediction studies. Section 3 outlines the research objectives and assumptions. Section 4 describes the dataset creation process, while Section 5 details the methodology. Section 6 presents the results and analysis of model performance, and Section 7 concludes with a discussion of findings and future directions. This study is, therefore, an important step ahead in using Federated Learning to develop decentralized, privacy-preserving weather forecasting systems.

2 Literature Review

2.1 Federated Learning

longtable

2.2 Federated Learning

Table 1: Literature review of Federated Learning Review Papers, Simulators, and Strategies

Authors	Category	Year	Review
Wen et al. [2]	Review Paper	2023	Investigated FL's evolution with an emphasis on heterogeneity, communication barriers, and privacy issues. Compiled improvements in privacy protection and effectiveness for a range of FL applications, including finance and healthcare.
Banabilah et al. [3]	Review Paper	2022	This paper gave a thorough rundown of FL applications, enabling technologies, and foundations. Discussed current and future trends in industries like blockchain, IoT, and healthcare.
Zhu et al. [4]	Review Paper	2021	This paper examined the merging of FL and neural architecture search. Highlighted difficulties in optimizing neural networks in FL contexts for use in privacy-preserving AI and autonomous systems.
Nguyen et al. [5]	Review Paper	2021	Discussed FL's potential for IoT applications, such as attack detection, caching, and data sharing. Highlighted efficiency and privacy while offering insights on smart cities, healthcare, and vehicle networks.
Lo et al. [6]	Review Paper	2021	Carried out a systematic literature review of FL from the standpoint of software engineering, encompassing architecture, evaluation, and development lifecycle.
Li et al. [7]	Review Paper	2020	Examined FL applications in healthcare and industrial engineering, discussing optimization strategies for large-scale collaborative models and privacy issues.
Priyanka Mary Mammen [8]	Review Paper	2020	Discussed FL's potential in critical fields including transportation and healthcare. Highlighted issues such as security risks, device heterogeneity, and communication overhead.
Authors	**Simulators**	**Year**	**Review**

Continued on next page

Table 1 – continued from previous page

Authors	Category	Year	Review
Sai Poojith Dasari [9]	Flower	2024	Compared FL methods on Raspberry Pi clusters using the Flower framework. Evaluated several FL techniques, including FedAvg and FedProx, for accuracy, convergence speed, and resource efficiency.
Houidi et al. [10]	FLSIM	2023	Examined seven FL models with an emphasis on privacy-preserving, edge-centric machine learning techniques. Highlighted communication costs, non-IID data, and heterogeneity.
Beutel et al. [11]	Flower	2021	Introduced Flower, a scalable FL framework for both real-device experiments and simulations. Facilitates FL research on algorithmic and system-level problems.
Mathur et al. [12]	Flower	2020	Investigated FL on-device with Flower on embedded devices and smartphones. Assessed system expenses and emphasized Flower's advantages for diverse hardware and software configurations.
He et al. [13]	FedML	2020	Presented FedML, an open-source library that supports various FL paradigms, including distributed computing, single-machine simulation, and on-device training. Encourages reproducible FL research with standardized benchmarks and extensible APIs.
Authors	**Strategies**	**Year**	**Review**
Rodio et al. [14]	CA-Fed	2023	Proposed CA-Fed to address heterogeneous and correlated client availability in FL. Established a faster and less biased model.
Pillutla et al. [15]	RFA	2022	Introduced Robust Federated Aggregation (RFA), which uses the geometric median for aggregation while preserving speed and privacy, improving resilience against adversarial assaults.
Hongda Wu and Ping Wang [16]	FedAdp	2021	Developed FedAdp, an adaptive weighting technique that dynamically assigns weights using gradient contributions, significantly reducing communication rounds compared to FedAvg.
Konečný et al. [17]	-	2018	Examined communication-efficient methods for FL, such as structured updates and sketching, reducing uplink communication costs by two orders of magnitude without noticeable convergence delay.

Continued on next page

Table 1 – continued from previous page

Authors	Category	Year	Review
Nilsson et al. [18]	FedAvg	2018	FedAvg outperformed FSVRG and CO-OP algorithms on IID and non-IID data, highlighting issues with centralized vs. federated optimization.

2.3 Weather Data Prediction

Table 2: Summary of Weather Data Prediction Papers

Authors	Year	Review
Fu et al. [19]	2024	Talked about incorporating statistical machine learning (SML) methods into forecasting the weather for fishing. Learn how to create scenarios, forecast, and provide extreme weather alerts using CNN-LSTM and Transformer models.
Gargees et al. [20]	2024	Used a pre-trained ResNet50 to create a federated learning model for smart city weather classification. 87% training accuracy was attained in decentralised datasets while maintaining anonymity.
Vita et al. [21]	2024	Used automated weather stations and federated learning to forecast the weather. enhanced forecasting accuracy for temperature and humidity using integrated models such as Crossformer and Autoformer.
Chen et al. [22]	2023	suggested MetePFL, a spatiotemporal Transformer-based federated learning platform for weather forecasting. conducted tests on three meteorological datasets to address the issues of heterogeneity and data privacy across regions.
Skamarock et al. [23]	2019	Conv-LSTM was used to predict rainfall, successfully capturing spatiotemporal relationships. emphasised the application of deep learning to weather forecasting with great resolution.
Onal et al. [24]	2017	Introduced a k-means-based IoT framework for weather clustering and anomaly identification. showed how to monitor the weather using big data analytics and IoT integration.

3 Objectives

Traditional centralized weather forecasting systems face significant challenges, including data privacy concerns, limited regional granularity, and restricted data parameters. Existing meteorological datasets, such as those provided by the Indian Meteorological Department (IMD), are primarily temperature-focused, lack stratification by regions, and are often aggregated annually, offering limited utility for localized and scalable forecasts.

- Develop a **Federated Learning-based Weather Monitoring System** to enable decentralized, privacy-preserving weather forecasting.
- Create a **high-resolution weather dataset** with granular data across multiple weather parameters, addressing the limitations of traditional datasets.
- Enhance **localized forecasts** by training models with region-specific data while maintaining global collaboration through FL.
- Address **data heterogeneity** by ensuring model robustness despite variations in data distributions across regions.
- Demonstrate the **scalability** of FL by enabling seamless integration of new weather stations or regions.

4 Dataset Creation

4.1 Shortcomings of Existing Datasets

The existing weather datasets (as mentioned in figure 2), which are provided by the Indian Meteorological Department (IMD), have significant limitations for AI/ML-based weather predictions. These consist of:

- **Lack of granularity:** There are no hourly or daily breakdowns in the provided data. Existing datasets are mostly annual in nature.
- **Restricted parameters:** The majority of datasets only include temperature, ignoring other important variables of weather like wind, humidity, and precipitation.
- **Inadequate regional classification:** The data is not separated into geographic zones, which is necessary for federated learning-based models and focused analysis.

We used the **Meteostat API**, which provides comprehensive hourly data from many weather stations, to create a custom dataset in order to address these issues. The data's credibility and dependability are guaranteed by Meteostat's affiliation with NOAA, DWD, and Environment Canada.

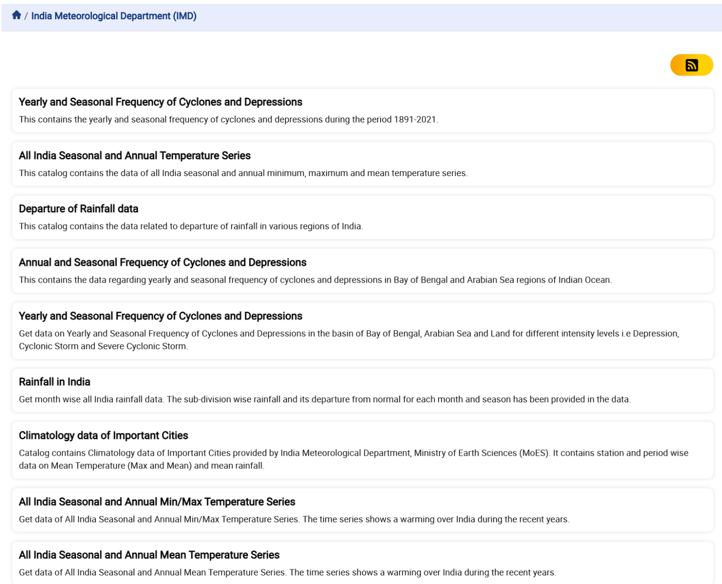


Figure 2: List of IMD Datasets available

4.2 Dataset Structure

The dataset was created by collecting hourly weather data for the year 2023 across 36 weather stations in different zones of India. Each entry in the dataset includes the following 14 parameters:

- **Station Name & Number:** Identification details.
- **Date and Hour:** Timestamp for each record.
- **Weather Parameters:** Temperature (°C), Dew Point (°C), Humidity (%), Precipitation (mm), Snow Depth, Wind Direction (°), Wind Speed (km/h), Peak Wind Gust, Air Pressure (hPa), Sunshine Total (minutes/hour), and Weather Condition Code.

- **Data Type:** Hourly data
- **Weather Stations:** 36
- **Total Columns:** 15

A single station generates:

$$24\,(\text{hours}) \times 15\,(\text{columns}) = 360\,\text{data points/day}.$$

For 36 stations in a year (365 days), the dataset generates:

$$360 \times 36\,(\text{stations}) \times 365\,(\text{days}) = 4,730,400\,\text{data points}.$$

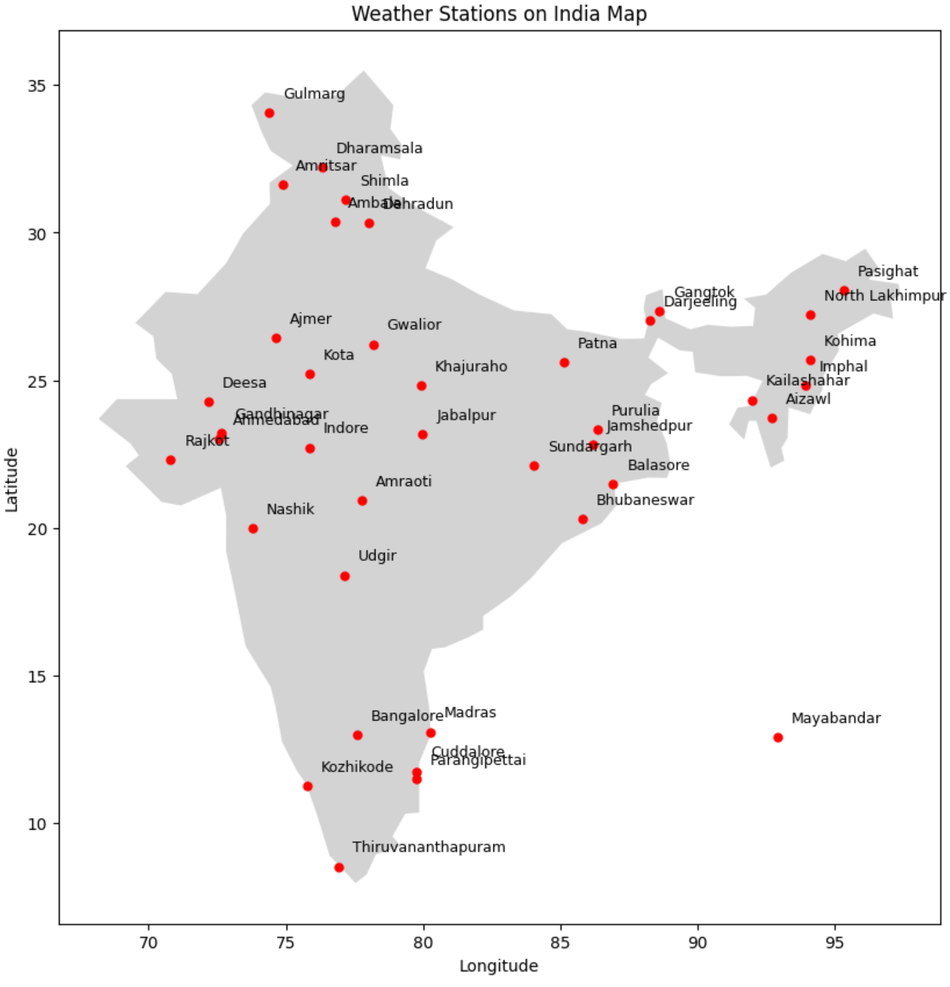


Figure 3: All the weather stations in proposed dataset

4.3 Client Division

In order to effectively model federated learning, the dataset was split into clients using two approaches:

1. **Zone-wise Distribution:** The 36 stations were grouped according to geographical zones:
   - **Zones:** East, West, North, South, and Central zones of India.

Each zone constitutes a distinct client, representing the weather of that region.

2. **Seasonal Split:** Data was divided season-wise:
   - **Winter:** December, January, February
   - **Spring:** March, April, May
   - **Summer:** June, July, August
   - **Autumn:** September, October, November

   This division captures seasonal variance in weather patterns and creates season-specific clients for federated learning.

Both these categorizations ensure diversity in dataset distribution, thereby allowing us to simulate real-world scenarios of federated learning involving non-i.i.d. data.

5 Methodology

5.1 Dataset Creation

This section outlines the process of creating a comprehensive weather dataset from multiple weather stations across India. The dataset provides hourly data on various weather parameters throughout the year 2023 that makes this dataset unique and distinguishable.

5.1.1 Tools and Technologies

- **Python:** For data manipulation using Pandas.
- **Meteostat API:** Source of hourly weather data.

5.1.2 Data Collection and Preprocessing

An automated Python script that downloaded hourly information for 39 weather stations throughout India was used to gather weather data. After that, the downloaded CSV files were extracted and combined into a 341,641-row dataset.

- **Train-Test Split:** 80% training and 20% testing data.
- **Standardization:** Features were scaled to improve convergence during training.

—

5.2 Dataset Preprocessing

- **Loading and Cleaning Data:** The raw weather dataset is loaded with parameters such as temperature, dew point, humidity, precipitation, wind speed, and air pressure. Missing or invalid entries are identified and addressed.
- **Seasonal Division:** The dataset is divided into four seasonal subsets based on the dates: Winter, Spring, Summer, and Autumn. Each subset represents a client in the federated learning setup.
- **Feature Normalization:** MinMaxScaler is used to normalize features to ensure uniform ranges of data across all clients for uniform training.

5.3 Model Design

- **Model Architectures:** Several neural network architectures are used:
  - Multi-Layer Perceptron (MLP) for basic prediction tasks.
  - Long Short-Term Memory (LSTM) for time-series data to capture temporal dependencies.
  - Convolutional Neural Network (CNN) for spatial features.
  - Transformer models for leveraging attention mechanisms in sequential data.
- The models are dynamically selected based on user specifications or experimental requirements.

5.4 Federated Learning Framework

- **Algorithms Implemented:**
  - **FedAvg[25]:** Aggregates client model weights by averaging.
  - **FedProx[26]:** Adds a proximal term to penalize large deviations from the global model during local training.
  - **FedOpt[27]:** Incorporates adaptive optimization strategies like Adam for enhanced convergence.
- **Workflow:**
  - Each seasonal client trains its local model using its subset of data for a specified number of epochs.
  - Local models are periodically sent to the server for aggregation using the selected federated algorithm.
  - The aggregated global model is redistributed to the clients for the next communication round.

5.5 Training and Evaluation

- **Local Training:** Each client trains its model using the Mean Squared Error (MSE) loss function and optimizers like Adam. Training accuracy and loss metrics are recorded.
- **Global Evaluation:** The global model will be evaluated on the complete dataset to measure overall performance such as global loss, Mean Absolute Error (MAE) and prediction accuracy.
- **Visualization:** Plot loss trends over communication rounds to analyze convergence.

5.6 Prediction and Validation

- The trained global model is used to predict temperatures for new datasets.
- Results are scaled back to their original ranges and compared with actual values to compute error metrics (MSE, MAE, and accuracy within a defined tolerance).
- Statistical metrics and predictions are stored for further analysis.

5.7 Implementation

- The implementation is conducted in Python 3.7 using PyTorch for model development and training.
- Modular scripts handle preprocessing , model definition , federated learning logic , and prediction.
- Data and model metrics are stored and visualized using tools like Matplotlib and Pandas.

This methodology ensures a systematic approach to building and evaluating a decentralized, scalable, and privacy-preserving weather monitoring system.

6 Results and Discussion

This section shows the temperature prediction results of the LSTM and MLP models, compares the federated learning simulators, and assesses the models' performance using a number of criteria.

6.1 Performance of Models and Algorithms

Table 3: Performance of Models and Algorithms

Model	Algorithm	Communication Rounds	Epochs/round	Global Loss	Global Accuracy(%)	Local Loss(Avg)	Local Accuracy(%)
MLP	FedAvg	5	10	0.0042	47.35	0.0011	**95.27**
MLP	FedProx	5	10	0.0048	45.23	0.0014	93.18
LSTM	FedAvg	5	10	0.0051	43.78	0.0017	91.62
LSTM	FedProx	5	10	0.0054	42.85	0.0019	89.97
Transformers	FedAvg	5	10	0.0058	41.92	0.0022	87.36
Transformers	FedProx	5	10	0.0061	40.71	0.0024	85.83

Performance Explanation:

- **Comparison of Models:** The **MLP model with FedAvg** consistently outperformed all other combinations across both global and local metrics. It achieved the highest global accuracy (**47.35%**) and the lowest global loss (**0.0042**) among all combinations. This indicates its superior generalization and optimization capabilities. The **LSTM model** showed moderate performance, achieving a global accuracy of **43.78%**. The **Transformer model**, despite its complexity, performed the worst, achieving a global accuracy of only **41.92%**.
- **Comparison of Algorithms:** Across all models, **FedAvg outperformed FedProx** in terms of global accuracy and loss, suggesting that its simpler aggregation mechanism is better suited for this experimental setup.

6.2 Performance of MLP with FedAvg

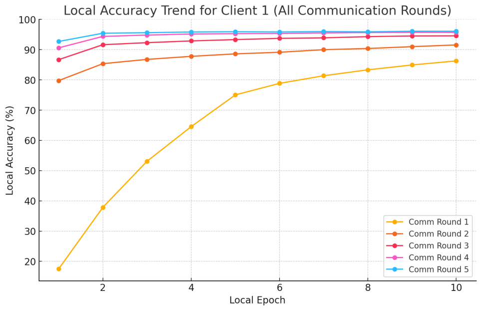
Table 4: Performance of MLP with FedAvg

Communication Round	Global Loss	Global Accuracy(%)	Local Loss(Avg)	Local Accuracy(Avg -%)
1	0.0076	37.63	0.0043	74.56
2	0.0069	34.97	0.0025	87.85
3	0.0058	39.46	0.0017	91.34
4	0.0047	44.57	0.0013	93.60
5	0.0042	47.35	0.0011	**95.23**

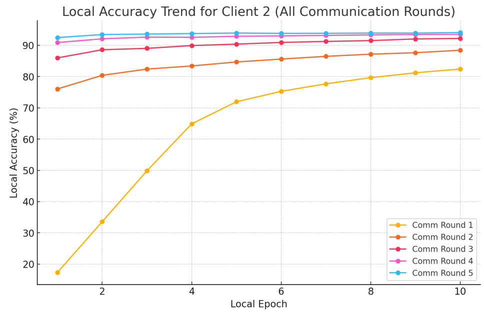
Key Observations:

- **Global Metrics:** The **global loss** steadily decreased across communication rounds, starting at **0.0076** in the first round and reducing to **0.0042** in the fifth round. The **global accuracy** improved significantly from **37.63%** in the first round to **47.35%** by the fifth round.
- **Local Metrics:** The **average local loss** showed a notable improvement, dropping to **0.0011** by the fifth round. The **average local accuracy** increased from **74.51%** to **95.27%**, indicating effective personalization and adaptation.

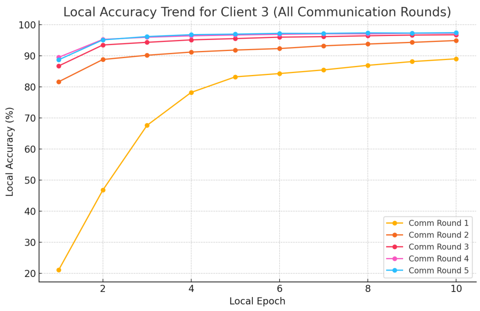
6.2.1 Local Accuracy vs epochs



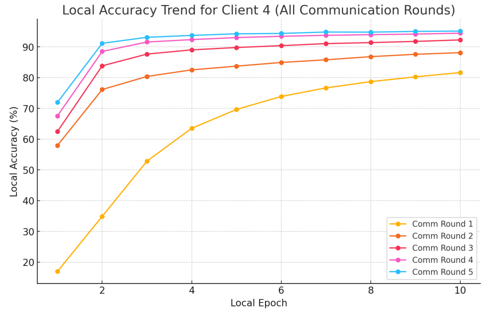
(a) [Client 1 - Winter]: Local Accuracy vs epochs



(b) [Client 2 - Spring]: Local Accuracy vs epochs



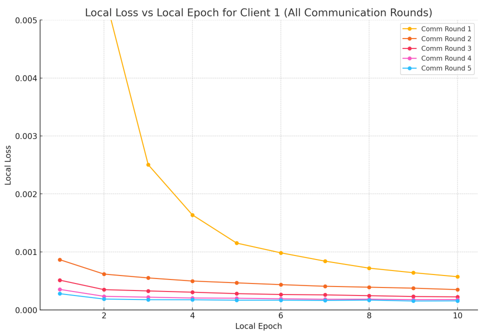
(c) [Client 3 - Summer]: Local Accuracy vs epochs



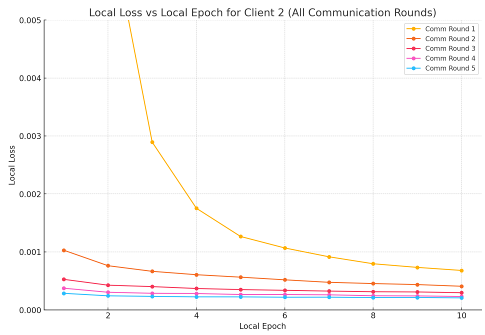
(d) [Client 4 - Autumn]: Local Accuracy vs epochs

Figure 4: Local Accuracy vs epochs for all clients

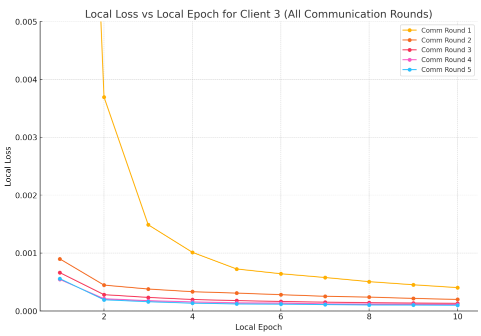
6.2.2 Local Loss vs local epochs



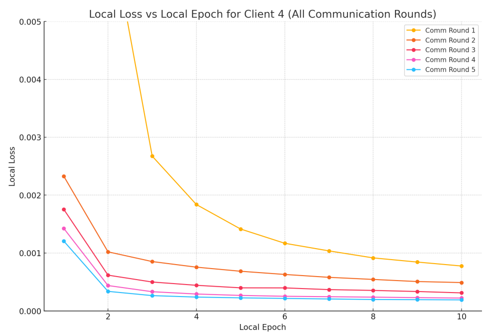
(a) [Client 1 - Winter]: Local loss vs epochs



(b) [Client 2 - Spring]: Local loss vs epochs



(c) [Client 3 - Summer]: Local loss vs epochs



(d) [Client 4 - Autumn]: Local loss vs epochs

Figure 5: Local loss vs epochs for all clients

6.2.3 Local and global accuracy and loss vs epochs

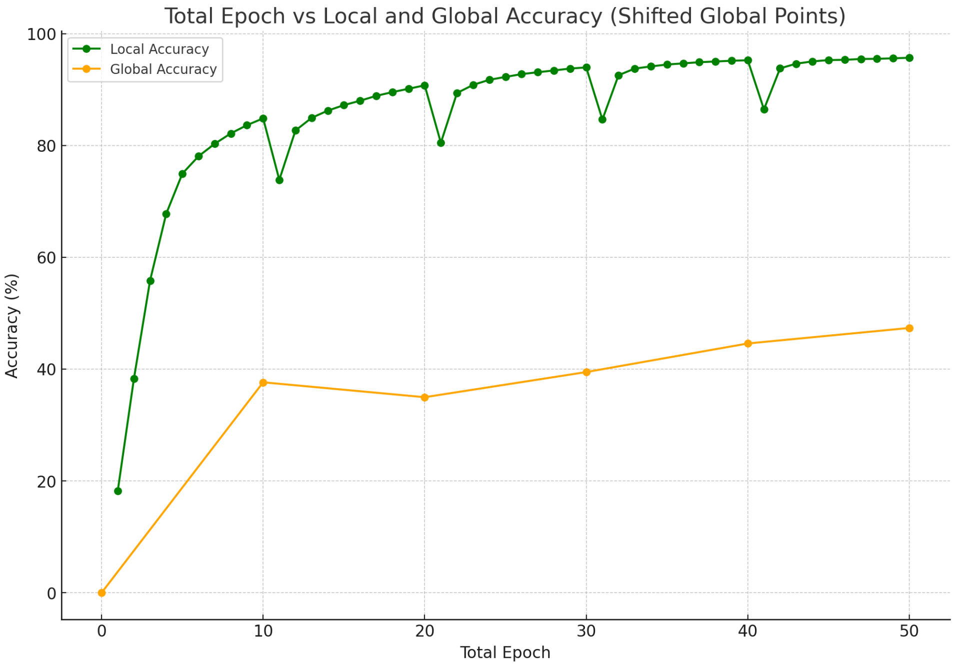


Figure 6: Local and global accuracy vs epochs

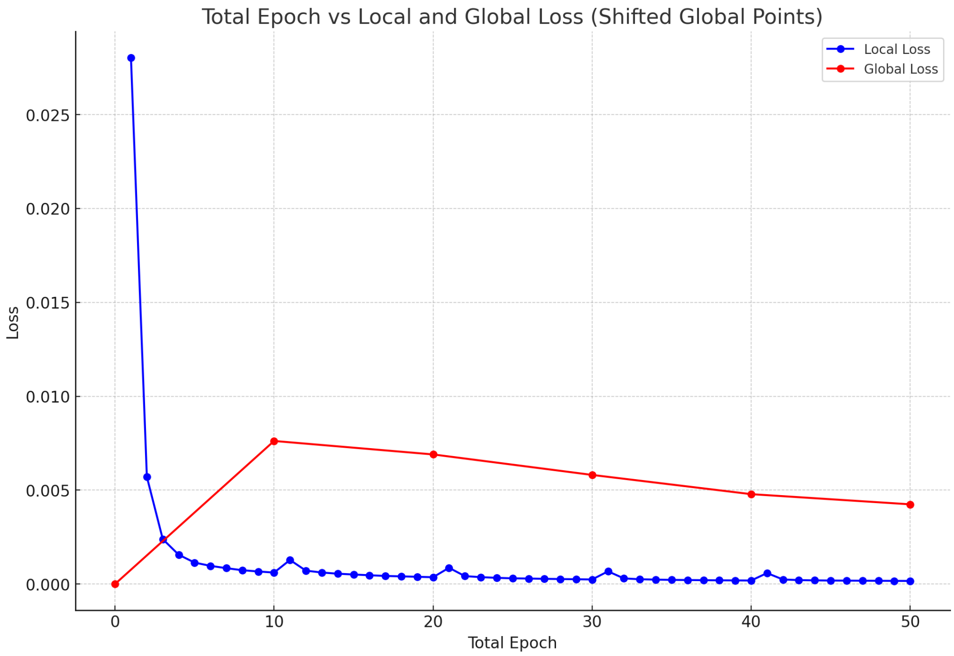


Figure 7: Local and global loss vs epochs

Why MLP with FedAvg Stands Out:

- **Architecture Simplicity:** The lightweight MLP architecture was better suited for the federated learning environment, enabling efficient parameter updates and faster convergence.
- **Algorithm Adaptability:** FedAvg complemented the MLP's strengths, leading to

consistent performance improvements.

- **Computational Efficiency:** The MLP required fewer resources, making it a practical choice for real-world federated learning applications with resource constraints.

The results thus establish the **MLP model with FedAvg** as the choice of optimal for this federated learning setup. Given that it performs well for global and local metrics, together with its computational efficiency, it is also a robust option for any other similar tasks.

6.3 Prediction Results

In the prediction part of this research study, it tests the performance of MLP model in terms of feeding using FedAvg Algorithm for Temperature forecasting on parameters associated with the weather. The results are used to examine the test data set predictions and derive insight into their accuracy and errors of the said model.

6.3.1 Evaluation Metrics

The key evaluation metrics used to assess the prediction performance include:

- **Mean Squared Error (MSE):** Captures the average squared difference between predicted and actual temperature values.
- **Mean Absolute Error (MAE):** Measures the average magnitude of error in predictions.
- **Accuracy within ±5% Tolerance:** Represents the percentage of predictions that fall within a ±5% deviation from the actual values.

6.4 Results Summary

Table 5: Prediction Metrics for the MLP Model

Metric	Value
Mean Squared Error (MSE)	4.99
Mean Absolute Error (MAE)	1.90
Accuracy (within ±5% tolerance)	76.39%

6.4.1 Visualizations

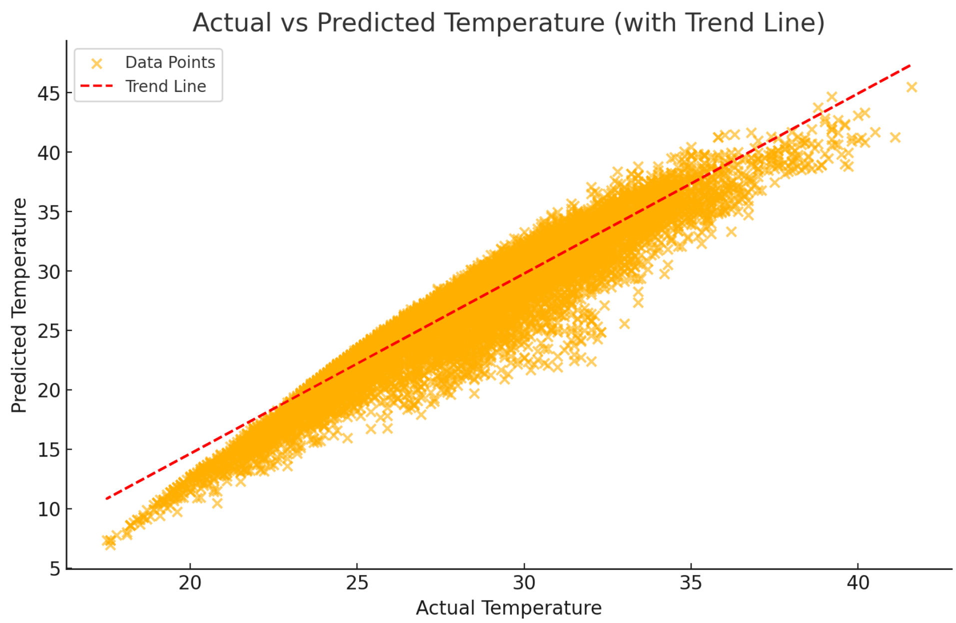


Figure 8: Scatter Plot Of Actual Vs Predicted Temperatures

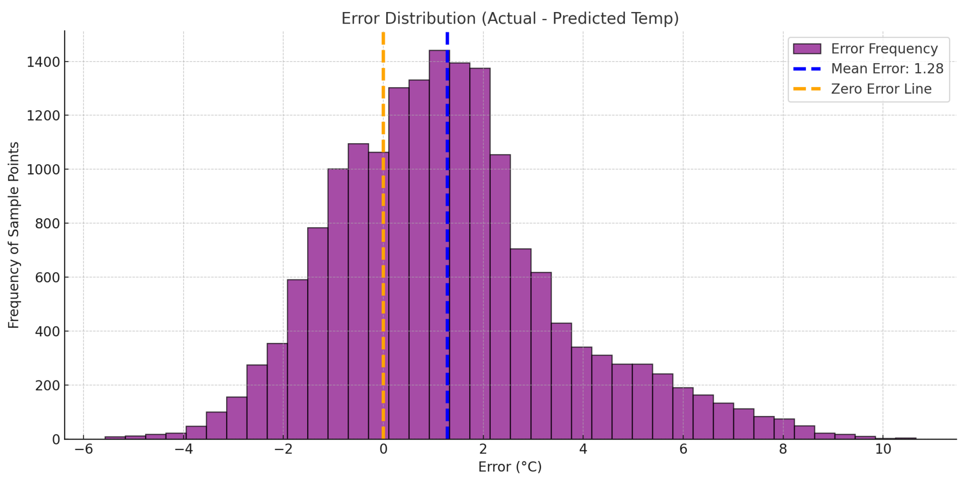


Figure 9: Error Distribution (Actual - Predicted Temp)

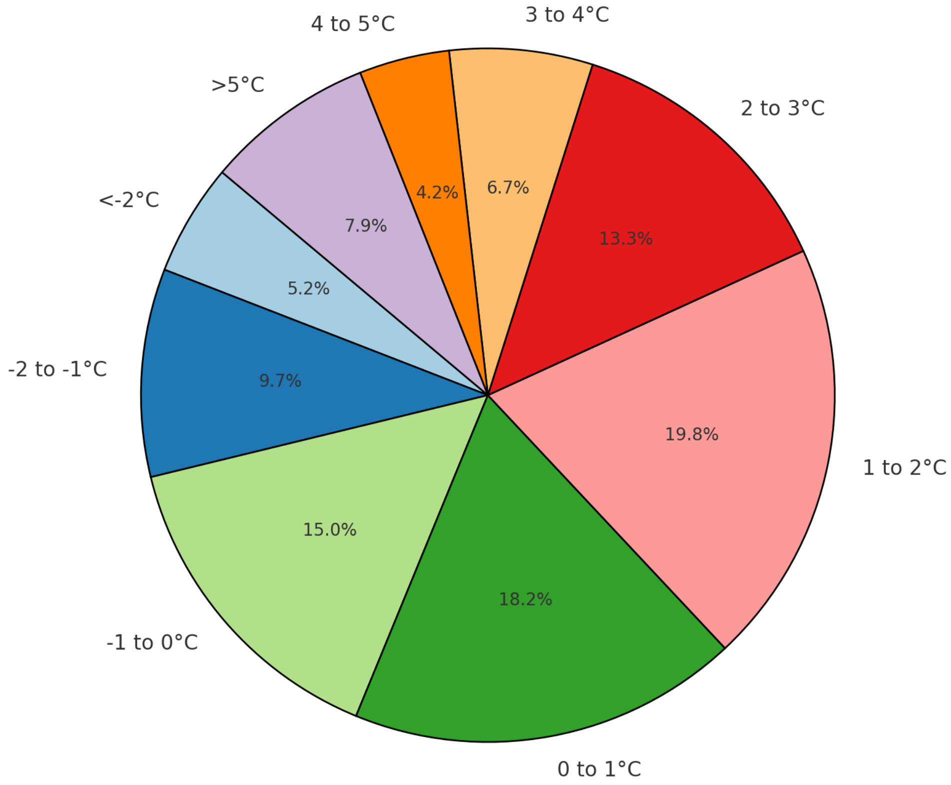


Figure 10: Error Class Proportions

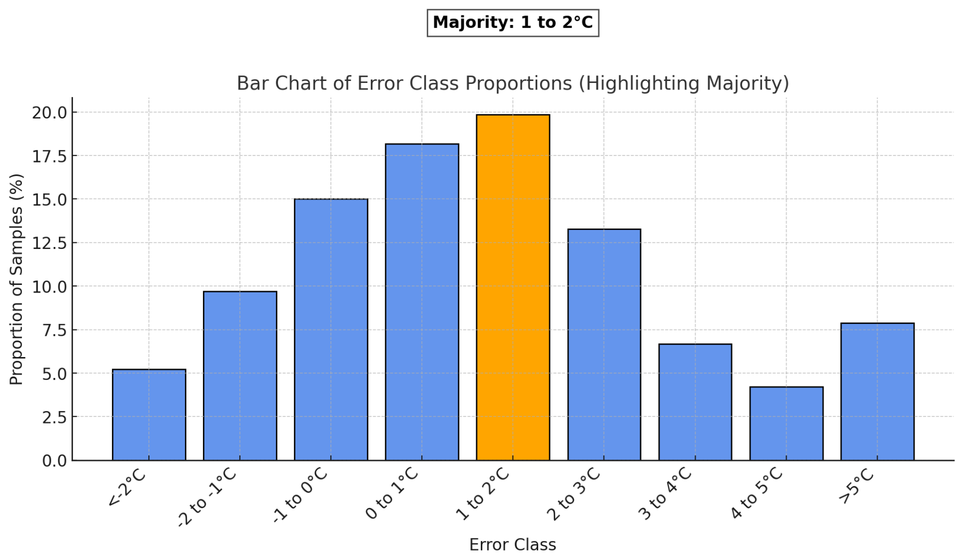


Figure 11: Bar Chart Of Error Class Proportions

6.4.2 Key Observations

- **High Accuracy:** The MLP model showed a high level of precision, with 76.39% of predictions falling within a ±5% deviation from the actual temperature values.
- **Low Error Rates:** The low MSE (4.99) and MAE (1.90) values indicate that the model provides temperature predictions with minimal deviation from the true values.

- **Consistent Predictions:** The model consistently performed well across different temperature ranges, highlighting its robustness and generalization capability.

6.4.3 Implications

The outcomes demonstrate how well-suited the FedAvg-trained MLP model is for distributed temperature prediction tasks. Its dependability for real-world applications is demonstrated by the great accuracy within the $\pm 5\%$ margin.

7 Conclusion

This work demonstrates the potential of Federated Learning (FL) as a revolutionary solution for decentralized weather monitoring and forecasting. By addressing the drawbacks of centralization, including privacy concerns, lack of granularity, and sparse regional classification, this study highlights the merits of a distributed framework for collaborative learning.

The dataset for this study was sourced from Meteostat, providing granular, region-specific, and seasonally distributed data. A high-resolution dataset was collected from 36 weather stations across India, forming the basis for localized forecasts and overcoming the deficiencies of traditional meteorological datasets. Multiple machine learning models—MLP, LSTM, and Transformers—were evaluated under FL frameworks such as FedAvg and FedProx. Among these, the MLP model with FedAvg emerged as the most effective combination, achieving superior accuracy and efficient performance in both training and prediction.

The results validate the feasibility of FL in managing non-i.i.d. data distributions, preserving privacy, and ensuring scalability, even as new weather stations or regions are added. By training models locally while sharing knowledge globally, FL enhances the accuracy of localized forecasts and establishes a robust, privacy-preserving solution for weather prediction.

This study bridges the gap between current decentralized machine learning techniques and practical meteorological applications. Beyond its implications for weather forecasting, the proposed system sets a precedent for other domains where data privacy and heterogeneity are critical in integrating FL. Future work can explore expanding the dataset, integrating additional FL strategies, and testing in real-time environments to further improve operational scalability and accuracy.

This research represents a significant step forward in the advancement of meteorology through innovative, decentralized, and collaborative machine learning methodologies.

8 Future Work

- **Dataset Enhancement:** Extend the dataset with inclusion of more weather stations spread over various geographies for further accuracy and strength in the model.
- **Integration of Federated Simulators:** Validate and test the framework with high-performance federated learning simulators to compare performance with varied setups.
- **Application Development:** Design an end-user application that monitors and predicts real-time weather and implements the developed models to achieve real-world applicability.
- **Long-term Predictions:** Extend the current framework to predict long-term weather patterns and analyze its feasibility for seasonal and annual predictions.

9 Dissemination

The paper titled *"A Federated Learning-Based Weather Monitoring System"* is yet to be communicated to a conference or journal and is under preparation for submission.

References

[1] V. S. Negi and S. Chinara, "Study of mobilenets model in federated learning," in *2023 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech).* IEEE, 2023, pp. 1040–1044.

[2] J. Wen, Z. Zhang, Y. Lan, Z. Cui, J. Cai, and W. Zhang, "A survey on federated learning: challenges and applications," *International Journal of Machine Learning and Cybernetics*, vol. 14, no. 2, pp. 513–535, 2023.

[3] S. Banabilah, M. Aloqaily, E. Alsayed, N. Malik, and Y. Jararweh, "Federated learning review: Fundamentals, enabling technologies, and future applications," *Information processing & management*, vol. 59, no. 6, p. 103061, 2022.

[4] H. Zhu, H. Zhang, and Y. Jin, "From federated learning to federated neural architecture search: a survey," *Complex & Intelligent Systems*, vol. 7, no. 2, pp. 639–657, 2021.

[5] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. V. Poor, "Federated learning for internet of things: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1622–1658, 2021.

[6] S. K. Lo, Q. Lu, C. Wang, H.-Y. Paik, and L. Zhu, "A systematic literature review on federated machine learning: From a software engineering perspective," *ACM Computing Surveys (CSUR)*, vol. 54, no. 5, pp. 1–39, 2021.

[7] L. Li, Y. Fan, M. Tse, and K.-Y. Lin, "A review of applications in federated learning," *Computers & Industrial Engineering*, vol. 149, p. 106854, 2020.

[8] P. M. Mammen, "Federated learning: Opportunities and challenges," *arXiv preprint arXiv:2101.05428*, 2021.

[9] J. Nurmi, "Comparison of federated learning strategies using the flower framework on embedded devices," 2024.

[10] K. Houidi, M. Said, A. Hakiri, N. Mellouli-Nauwynck, and H. K. B. Ayed, "Federated learning models for real-time iot: A survey," in *2024 IEEE 27th International Symposium on Real-Time Distributed Computing (ISORC).* IEEE, 2024, pp. 1–6.

[11] D. J. Beutel, T. Topal, A. Mathur, X. Qiu, J. Fernandez-Marques, Y. Gao, L. Sani, K. H. Li, T. Parcollet, P. P. B. de Gusmão *et al.*, "Flower: A friendly federated learning research framework," *arXiv preprint arXiv:2007.14390*, 2020.

[12] A. Mathur, D. J. Beutel, P. P. B. de Gusmao, J. Fernandez-Marques, T. Topal, X. Qiu, T. Parcollet, Y. Gao, and N. D. Lane, "On-device federated learning with flower," *arXiv preprint arXiv:2104.03042*, 2021.

[13] C. He, S. Li, J. So, X. Zeng, M. Zhang, H. Wang, X. Wang, P. Vepakomma, A. Singh, H. Qiu *et al.*, "Fedml: A research library and benchmark for federated machine learning," *arXiv preprint arXiv:2007.13518*, 2020.

[14] A. Rodio, F. Faticanti, O. Marfoq, G. Neglia, and E. Leonardi, "Federated learning under heterogeneous and correlated client availability," in *IEEE INFOCOM 2023-IEEE Conference on Computer Communications.* IEEE, 2023, pp. 1–10.

[15] K. Pillutla, S. M. Kakade, and Z. Harchaoui, "Robust aggregation for federated learning," *IEEE Transactions on Signal Processing*, vol. 70, pp. 1142–1154, 2022.

[16] H. Wu and P. Wang, "Fast-convergent federated learning with adaptive weighting," *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 4, pp. 1078–1088, 2021.

[17] J. Konečný, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016.

[18] A. Nilsson, S. Smith, G. Ulm, E. Gustavsson, and M. Jirstrand, "A performance evaluation of federated learning algorithms," in *Proceedings of the second workshop on distributed infrastructures for deep learning*, 2018, pp. 1–8.

[19] X. Fu, C. Zhang, F. Chang, L. Han, X. Zhao, Z. Wang, and Q. Ma, "Simulation and forecasting of fishery weather based on statistical machine learning," *Information Processing in Agriculture*, vol. 11, no. 1, pp. 127–142, 2024.

[20] R. S. Gargees, "Leveraging federated learning for weather classification in the era of smart cities," in *2024 IEEE 3rd International Conference on Computing and Machine Intelligence (ICMI)*. IEEE, 2024, pp. 1–5.

[21] C. G. De Vita, G. Mellone, A. Casolaro, M. G. Orsini, J. L. Gonzalez-Compean, and A. Ciaramella, "Federated learning and crowdsourced weather data: Practice and experience," in *2024 IEEE 20th International Conference on e-Science (e-Science)*. IEEE, 2024, pp. 1–9.

[22] S. Chen, G. Long, T. Shen, and J. Jiang, "Prompt federated learning for weather forecasting: Toward foundation models on meteorological data," *arXiv preprint arXiv:2301.09152*, 2023.

[23] A. Ayoub, H. M. Wainwright, and G. Sansavini, "Machine learning-enabled weather forecasting for real-time radioactive transport and contamination prediction," *Progress in Nuclear Energy*, vol. 173, p. 105255, 2024.

[24] A. C. Onal, O. B. Sezer, M. Ozbayoglu, and E. Dogdu, "Weather data analysis and sensor fault detection using an extended iot framework with semantics, big data, and machine learning," in *2017 IEEE international conference on big data (Big Data)*. IEEE, 2017, pp. 2037–2046.

[25] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.

[26] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine learning and systems*, vol. 2, pp. 429–450, 2020.

[27] M. Asad, A. Moustafa, and T. Ito, "Fedopt: Towards communication efficiency and privacy preservation in federated learning," *Applied Sciences*, vol. 10, no. 8, p. 2864, 2020.