


**Please cite the Published Version**

Manzoor, Habib Ullah, Shabiar, Attia, Nguyen, Dinh C, Mohjazi, Lina, Kaushik, Aryan  and Zoha, Ahmed (2024) Rethinking Federated Learning: An Adversarial Perspective on Global vs. Local Learning for Load Forecasting. In: 2024 IEEE Conference on Standards for Communications and Networking (CSCN), 25 November 2024 - 27 November 2024, Belgrade, Serbia.

**DOI:** <https://doi.org/10.1109/cscn63874.2024.10849714>

**Publisher:** IEEE

**Version:** Accepted Version

**Downloaded from:** <https://e-space.mmu.ac.uk/638327/>

**Usage rights:**  In Copyright

**Additional Information:** © 2024 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

**Enquiries:**

If you have questions about this document, contact [openresearch@mmu.ac.uk](mailto:openresearch@mmu.ac.uk). Please include the URL of the record in e-space. If you believe that your, or a third party's rights have been compromised through this document please see our Take Down policy (available from <https://www.mmu.ac.uk/library/using-the-library/policies-and-guidelines>)

# Rethinking Federated Learning: An Adversarial Perspective on Global vs. Local Learning for Load Forecasting

Habib Ullah Manzoor<sup>1</sup>, Attia Shabiar<sup>2</sup>, Dinh C. Nguyen<sup>3</sup>, Lina Mohjazi<sup>1</sup>, Aryan Kaushik<sup>4</sup>, and Ahmed Zoha<sup>1\*</sup>

<sup>1</sup> James Watt School of Engineering, University of Glasgow, United Kingdom

<sup>2</sup> Ghulam Ishaq Khan Institute, Topi, Pakistan.

<sup>3</sup> Department of Electrical and Computer Engineering, University of Alabama in Huntsville, USA

<sup>4</sup> Department of Computing & Mathematics, Manchester Metropolitan University, UK

Email: h.manzoor.1@research.gla.ac.uk, attiashabbir371@gmail.com, dinh.nguyen@uah.edu,

lina.mohjazi@glasgow.ac.uk, a.kaushik@ieee.org, Ahmed.Zoha@glasgow.ac.uk

\*Corresponding Author: Ahmed.Zoha@glasgow.ac.uk

**Abstract**—Resilient federated learning (FL) systems are essential for accurate load forecasting, especially when under adversarial attacks. Since these systems aggregate decentralized data from various sources, they are particularly vulnerable to attacks that can undermine forecast accuracy and reliability. To enhance robustness in load forecasting, our study investigates methods for strengthening FL systems by optimizing the balance between global and local learning processes. This paper explores the trade-offs between global and local learning in federated load forecasting under adversarial conditions. We develop a neural network framework tailored for federated short-term load forecasting and assess its performance against model poisoning attacks. Our experiments demonstrate that increasing the number of local training epochs while reducing global communication rounds can significantly enhance model robustness. Specifically, when local epochs are increased from 1 to 10 and global epochs are decreased from 1000 to 100, the average client Mean Absolute Percentage Error (MAPE) decreases from 92.3% to 4.3% under attack conditions. This improvement stems from a reduced attack surface and the concept of catastrophic forgetting, where local models gradually mitigate adversarial effects through extended training on authentic data, providing valuable insights for the design of secure and efficient distributed energy forecasting systems.

**Index Terms**—Distributed learning, Cyber security, Load forecasting, Cyber attack

## I. INTRODUCTION

Energy is a fundamental driver of modern economies, yet it is also a significant contributor to global warming, responsible for approximately two-thirds of greenhouse gas emissions [1]. Recognizing the urgent need for action, countries and organizations have established ambitious targets. For instance, the European Union aims to reduce emissions by 40% and increase energy efficiency by 27% by 2030 [2]. As energy demand continues to grow, the importance of effective energy management cannot be overstated. Short-term load forecasting (STLF) has emerged as an essential tool, allowing utility companies to predict demand fluctuations, optimize resource

allocation, and seamlessly integrate renewable energy sources into the grid [3]. STLF further supports the optimization of electric vehicle (EV) charging infrastructure by anticipating household energy demand patterns, thus helping utilities manage peak demand and enhance grid reliability. Financially, STLF proves invaluable for Xcel Energy, for example, saved 2.5 million USD by reducing its forecasting errors from 15.7% to 12.2%, and G.E. Energy has documented yearly savings of 5 billion USD through enhanced forecasting accuracy, leading to an operating cost reduction of 12–20 per MWh [4], [5].

Historically, centralized approaches to STLF have been widely used, requiring that data be transmitted from individual households or buildings to a central server for analysis. While effective, these approaches result in significant network traffic and associated costs [6]. Additionally, centralized machine learning strategies raise concerns about data security and privacy. For instance, sharing sensitive energy consumption data with central servers heightens the risk of exposure to unauthorized access and breaches, which is increasingly worrisome under stringent data protection regulations, such as the GDPR [7]. Beyond regulatory concerns, centralized systems are also vulnerable to cyber-attacks, which can have severe consequences for energy infrastructure. In 2022, the energy sector accounted for 10.7% of all cyber incidents, placing it among the most targeted industries [8]. High-profile attacks on centralized systems, such as the 2021 Colonial Pipeline ransomware attack and cyber intrusions on State Load Dispatch Centres (SLDCs) in India, have underscored these vulnerabilities [9], [10]. A decentralized approach could mitigate some of these risks, dispersing data and reducing the potential impact of attacks.

Advanced metering infrastructure and the widespread adoption of smart meters have revolutionized data collection capabilities, enabling utilities to gather energy consumption data at frequent intervals. In the UK, for instance, over 15 million

smart meters are operational across residential and commercial properties, facilitating granular data collection and allowing for more accurate predictions [11]. This decentralized data collection is crucial for monitoring and controlling power systems, particularly as intermittent renewable energy sources become more prevalent [12]. Forecasting at the household level, however, presents challenges due to the stochastic nature of residential energy use, which can vary widely based on daily behaviors, habits, and external factors [13]. Moreover, the need for customized machine learning models for individual meters has led to a significant increase in computational requirements. Developing these models in a centralized manner can quickly become cost-prohibitive and burdensome, highlighting the need for more scalable, distributed forecasting solutions that can address these complexities while supporting real-time energy management.

Federated learning (FL) has emerged as a promising approach for short-term energy forecasting, addressing the privacy issues associated with distributed energy sources [14], [15]. FL, as defined by IEEE Std 3652.1-2020, is a machine learning framework designed to train models collaboratively on decentralized data sources, emphasizing privacy and compliance without requiring data centralization [16]. It is categorized into horizontal and vertical FL, based on data feature distribution [16]. It enables collaborative model development by leveraging the collective capabilities of edge devices without the exchange of raw training data. In this setup, each device trains independently, sending updated model weights to a central server for aggregation. The server redistributes these weights for further training iterations, continuing until the desired model accuracy is achieved [17], [18].

Despite its benefits in aggregating insights from multiple devices to improve model quality, FL also creates unintended vulnerabilities to adversarial attacks [19]. Following the foundational work of [20], recent studies have identified various strategies for conducting adversarial attacks within FL systems [21]–[24]. FL systems face various attack methods that threaten their integrity and security. Data poisoning attacks occur when attackers inject harmful samples into local training data, causing the global model to adopt undesirable behaviors and potentially fail [25], [26]. In model poisoning attacks [27]–[30], the focus shifts to manipulating the updates sent by clients, skewing the global model’s behavior. Furthermore, Inference attacks enable adversaries to deduce sensitive information from clients’ local data, while targeted model inversion attacks allow attackers to reconstruct specific clients’ training data by exploiting model updates [31]. Backdoor attacks involve compromised clients injecting malicious behaviors into the global model through manipulated updates [32]. Finally, Sybil attacks entail creating multiple fake client identities to flood the system with harmful updates, undermining the model’s integrity [33].

According to IEEE Std 3652.1-2020, in B2C IoT applications, designed FL framework should be able to defend against data recovery from read-write attacks and channel monitoring [16]. In a read-write attack, an adversary gains

the ability to both read and modify the model updates being sent between the participating nodes and the central server. In light of these threats, various defensive measures have been proposed to counter attacks [34]–[38]. However, all of these defense frameworks add complexity to the system. This highlights the need for a resilient yet simplified FL system. One way to enhance robustness in an FL system is to reduce the attack surface. Specifically, in the case of model poisoning attacks, the attacker typically targets each communication round. By increasing the number of local epochs, the FL model requires fewer global epochs (or communication rounds) to converge, reducing the attacker’s opportunities to interfere.

Additionally, emphasizing local learning enables the local models to train on their respective data across multiple epochs, helping mitigate the effects of attacks due to a phenomenon known as catastrophic forgetting [39]. In neural networks, catastrophic forgetting refers to the model’s tendency to overwrite previous knowledge when exposed to new data. However, this phenomenon can be leveraged as a defensive mechanism in FL. As local models continue to train on fresh, authentic data, they gradually “forget” the harmful patterns introduced by poisoned updates. By focusing more on local learning, the model can increasingly ignore or “forget” adversarial influences from previous rounds, essentially erasing the impact of attacks over time.

This process can be especially useful when combined with fewer communication rounds. With less frequent synchronization, the local models develop independently on non-poisoned data for extended periods, which helps them reinforce benign patterns and diminish malicious alterations. Thus, catastrophic forgetting, often seen as a limitation in machine learning, can be repurposed to serve as an implicit filtering mechanism, reducing the lingering impact of adversarial updates and creating a more robust federated learning framework. In this paper we examine the effect of local and global epochs for federated load forecasting in detail under adversarial attack.

Here are the main contributions of this paper:

- 1) Developed a neural network framework for federated load forecasting.
- 2) Investigated the impact of adversarial attacks on federated load forecasting performance.
- 3) Examined the role of local and global epochs in mitigating adversarial effects.

The remainder of this paper is organized as follows: Section II outlines the framework for the federated learning training process. In Section III, we discuss threat modeling. Section IV presents our experiments and results, and finally, Section V concludes our work.

## II. FEDERATED LOAD FORECASTING

In load forecasting, FL enables multiple clients to collaboratively train a global neural network model while preserving the privacy of their local datasets. According to IEEE Std 3652.1-2020, this approach aligns with horizontal FL, where clients with similar feature spaces cooperate by iteratively training local models and aggregating their learned parameters

to improve the global model's accuracy and generalizability [40]. In this setup, the data owners, such as energy providers or individual devices retain control over their local datasets and perform training on their data, sending only model updates to a coordinator (central server). Server, often managed by the model owner, aggregates the updates from multiple data owners to refine the global model and subsequently distributes it back for further rounds of local training. This setup, as shown in Figure 1, outlines the systematic process of FL for load forecasting, ensuring both privacy and collaborative learning by leveraging neural network architectures.

- 1) Initialization of Local Models In FL, each client  $k$  initializes its local neural network model  $M_k$  with random weights  $w_k^{(0)}$ . The initial model can be based on a predefined architecture suitable for load forecasting. The goal is to train the local models on the clients' private datasets while maintaining data privacy.

$$M_k = \text{initialize}(w_k^{(0)}) \quad (1)$$

- 2) Local Training Each client  $k$  trains its local neural network model  $M_k$  on its own dataset  $D_k$  for a defined number of local epochs  $E_k$ . The local training process aims to minimize the loss function  $L$ , which measures the difference between the predicted load and the actual load. The optimization is performed using techniques such as stochastic gradient descent (SGD). The updated weights after local training are denoted as  $w_k^{(e)}$  for epoch  $e$ .

$$w_k^{(e)} = w_k^{(e-1)} - \eta \nabla L(w_k^{(e-1)}, D_k) \quad (2)$$

where  $\eta$  is the learning rate.

- 3) Weight Aggregation After completing local training, the clients send their updated weights  $w_k^{(E_k)}$  to the server. The server aggregates the received weights to form the new global model weights  $w^{(t)}$  using a weighted average based on the number of samples in each client's dataset  $n_k$ :

$$w^{(t)} = \frac{\sum_{k=1}^K n_k w_k^{(E_k)}}{\sum_{k=1}^K n_k} \quad (3)$$

where  $K$  is the total number of clients. This step ensures that the global model reflects the contributions of all clients while maintaining data privacy.

- 4) Global Model Update The aggregated global neural network model  $M$  is updated with the new weights  $w^{(t)}$ . This global model serves as the updated representation of the collective knowledge from all participating clients. The global model can then be evaluated on a validation set to assess its performance and convergence.

$$M = \text{update}(w^{(t)}) \quad (4)$$

- 5) Iterative Process and Convergence The FL process is iterative; steps 2 to 4 are known as communication round

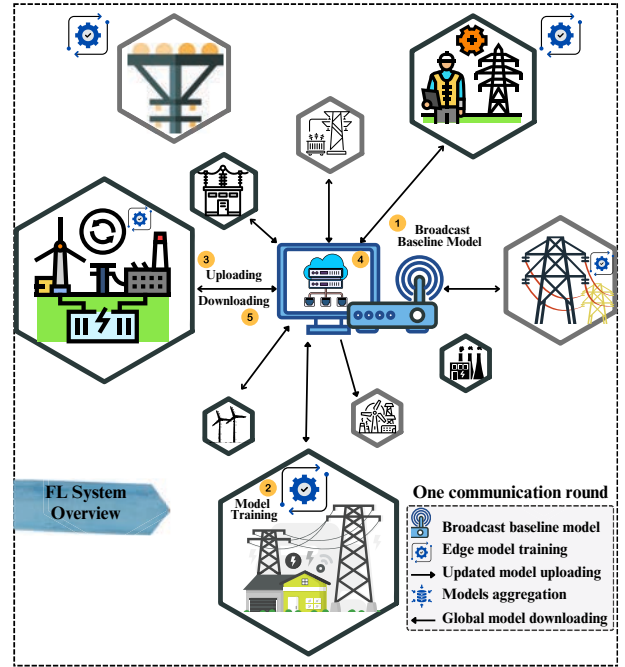


Fig. 1. Overview of a FL system.

or global epoch, repeated for several times. The convergence criterion can be based on the improvement of the average loss across global evaluations or a specified number of rounds.

Furthermore, IEEE Std 3652.1-202 state that an economic incentive mechanism is vital in FL applications. The server typically manages this, calculating and distributing payments to data owners and collecting fees from model users. This structure fosters participation by rewarding data owners, ensuring system sustainability, and supporting a model where both data and model usage remain privacy-preserving and regulated under the coordinator's oversight [16].

### III. THREAT MODELING

Threat modeling is a systematic method for identifying risks and vulnerabilities within a system by examining its structure and elements [41]. This process entails recognizing potential adversaries, understanding their goals, investigating the techniques they might use, and evaluating the potential consequences of a successful attack on the system's confidentiality, integrity, or availability [42]. In Federated Learning (FL), attackers possess varying degrees of system knowledge. In a white-box attack, the attacker has full access to the training and test sets, model structures, and updates [43]. A grey-box attacker has partial knowledge, meaning they are aware of some aspects of the model or data, but lack complete information [44]. Lastly, in a black-box attack, the attacker has no access to model structures or parameters but can still exploit the training set to craft an attack [45]. This research assumes a white-box attacker manipulating model parameters for poisoning. The primary goal of the attacker is to degrade the performance of both local and global models. By altering

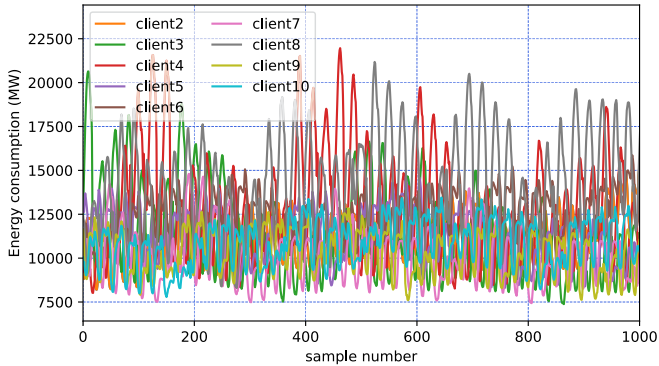


Fig. 2. A sample of data set.

the layers of a local model, the attacker skews predictions, disrupting decision-making processes. This manipulation can corrupt the global model, negatively impacting the performance of other clients in the FL system.

In this study, we implement a Model Flipping Attack [46]. This attack involves inverting the actual updates, allowing the composition of updates that closely resemble the originals but can undermine the model’s integrity, as the attacker modifies the weights of specific model layers. The attack is only created on the first layer of local model of first client [29]. Flipping the weights of a single layer may have minimal impact, whereas flipping all layers can lead to significant disruptions. To reduce the chances of detection and enhance the attack’s longevity, we adopt a partial poisoning strategy, targeting two layers at a time. This method allows the attacker to impair system performance without completely compromising the entire FL system [26].

#### IV. EXPERIMENTS AND RESULTS

In this paper all the experiments were conducted on a desktop with an Intel Core i5-6200U CPU (2.30 GHz), 12 GB RAM, running Windows 10 Pro (64-bit), using Visual Studio Code and Python 3.12.2.

##### A. Dataset for Analysis

In our research, we focus on a substation dataset from PJM Interconnection LLC, a regional transmission organization (RTO) in the United States, which is publicly available on Kaggle [47] and offers critical insights into energy consumption across various substations. Specifically, we utilize the COMED\_hourly Dataset, comprising 66,500 samples with values ranging from a minimum of 7,263 (MW) to a maximum of 21,349 (MW). the dataset is graphically represented in Figure 2. Designed for load forecasting, this dataset includes five key attributes: readings from the last hour, the last day, the last week, the 24-hour average, and the weekly average. For our experiments, we establish ten independent clients, each contributing distinct data from this dataset, thereby enhancing the diversity and robustness of our analysis.

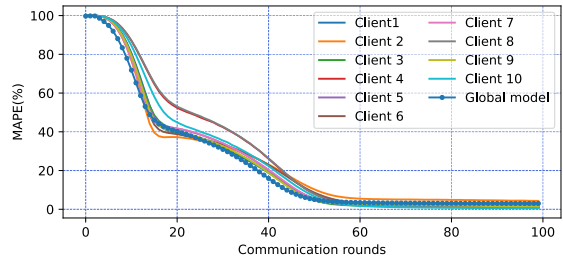


Fig. 3. MAPE of all clients during training process of global model.

##### B. Model Design

To conduct effective load forecasting, implement a three-layered Artificial Neural Network (ANN) for load forecasting, consisting of 100 neurons in the first layer, 50 in the second, and a single neuron in the output layer, all using Rectified Linear Unit (ReLU) activation functions. Train the network with the Adam optimizer and MAPE as the loss function. Split the dataset into training and testing sets at a 70/30 ratio, employing FL over 100 communication rounds, with each client having 2 local epochs and a batch size of 300. We assessed the global model’s performance by aggregating 10% of each client’s data to evaluate its accuracy and effectiveness.

##### C. Evaluation Metric

Evaluated the load forecasting models using the Mean Absolute Percentage Error (MAPE) metric, which provides a normalized measure of the average absolute percentage difference between actual and predicted values.

MAPE is calculated as:

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{A_i - P_i}{A_i} \right| \quad (5)$$

where  $A_i$  represents the actual value,  $P_i$  denotes the forecasted value, and  $n$  is the total number of data points.

##### D. Baseline results

After running federated learning (FL) for 100 communication rounds, the average client MAPE achieved was 1.4%. Baseline results are presented in Figure 3, showing that the model converged around 53 communication rounds. The FL system was then tested under a model flip attack, specifically targeting client one. This attack significantly impacted forecasting accuracy, increasing the average client MAPE from 1.4% to 92.2%. The training process of FL under this adversarial effect is depicted in Figure 4 [27], [28], [48], [49].

##### E. Effect local and global epochs

In this section, we analyze the relationship between local and global epochs (communication rounds) under adversarial attack through comprehensive experimentation. We varied the number of local and global epochs while keeping the cumulative epochs fixed at 1000. This means that each client runs 1000 epochs regardless of the changes in local and global epochs. An interesting pattern emerges as local epochs

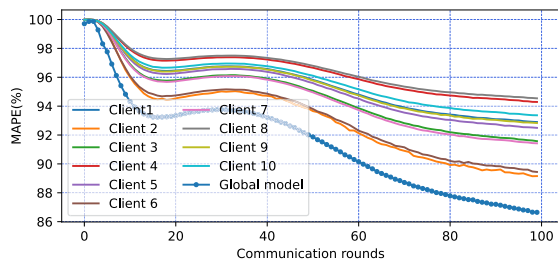


Fig. 4. MPAE of all clients during training process of global model under model flipping attack.

increase and global epochs decrease. The average client MAPE decreased from 92.3% to 4.3% when local epochs were increased from 1 to 10 and global epochs were decreased from 1000 to 100 in the presence of an adversarial attack. This effect is summarized in Table I and Figure 5. The relationship between local epochs and Avg MAPE appears to be non-linear, as the average MAPE decreases rapidly at first and then levels off. The initial significant drop (from 92.3% to 68.06% and further to 45.4%) shows that the model benefits most from the first few local epochs, suggesting diminishing returns as the number of local epochs increases beyond a certain point.

This reduction can be attributed to two primary reasons. First, when local epochs increase and global epochs decrease, the number of attack iterations is limited, as each attacker can only strike once in each round. This limitation effectively reduces the attack surface, thereby diminishing the adversarial influence on the model. Second, as the local epochs increase, the local models retain more information from the local data. This retention allows the models to adapt better to the data they are trained on while gradually forgetting the adversarial impact due to a phenomenon known as catastrophic forgetting. Catastrophic forgetting refers to the tendency of neural networks to forget previously learned information when trained on new data. By emphasizing local training over global updates, the model can prioritize learning from the local dataset, thereby mitigating the adverse effects of attacks and improving overall robustness.

#### F. Computational and communication cost

The communication cost is calculated based on the data transmitted between the server and devices [49]. In the experiments mentioned above, as the number of global epochs decreases, the communication cost also decreases. Our designed deep learning model is 38 KB. In each global epoch, each device transmits the model, and so does the server. When the communication rounds decreased from 1000 to 100, the communication cost reduced from 76,000 KB to 7,600 KB. However, the computational cost remains the same, as the cumulative epochs are fixed at 1000. Each device will perform the training 1000 times, regardless of the number of local and global epochs.

TABLE I  
EFFECT OF LOCAL AND GLOBAL EPOCHS ON AVG CLIENT MAPE UNDER MODEL FLIPPING ATTACK

Local Epoch	Global Epoch	Cumulative Epoch	Avg. MAPE (%)
1	1000	1000	92.3
2	500	1000	68.06
3	333	999	45.4
4	250	1000	33.7
5	200	1000	22.5
6	167	1002	13.2
7	143	1001	9.1
8	125	1000	7.3
9	111	999	5.3
10	100	1000	4.3

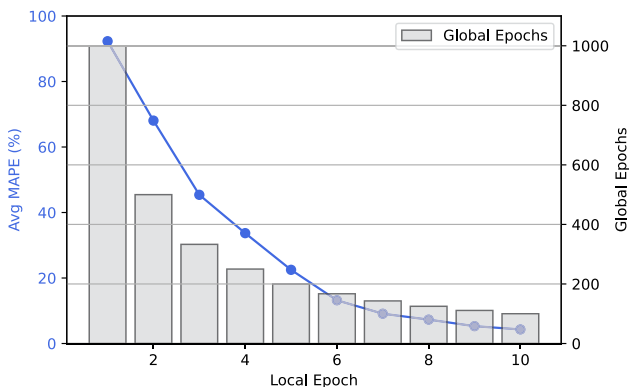


Fig. 5. Effect of local and global epochs on Avg client MAPE under model flipping attack.

## V. CONCLUSION

FL presents a promising approach for load forecasting in decentralized environments, offering improved privacy and scalability by aggregating data from diverse sources. This research highlights the critical role of optimizing the balance between global and local learning to enhance the robustness of FL systems under adversarial conditions. By increasing local training epochs and reducing global communication rounds, our experiments reveal a significant reduction in average client MAPE, dropping from 92.3% to 4.3% in the presence of model poisoning attacks. This improvement is attributed to a reduced attack surface and the effects of catastrophic forgetting, where extended local training enables models to progressively counteract adversarial influences.

## REFERENCES

- [1] (2020) Un environment programme. energy. [Online]. Available: <https://www.unep.org/explore-topics/energy>.
- [2] (2017) European environment agency. energy and climate change. [Online]. Available: <https://www.eea.europa.eu/signals/signals-2017/articles/energy-and-climate-change>
- [3] X. Luo, L. O. Oyedele, A. O. Ajayi, O. O. Akinade, J. M. D. Delgado, H. A. Owolabi, and A. Ahmed, "Genetic algorithm-determined deep feedforward neural network architecture for predicting electricity consumption in real buildings," *Energy and AI*, vol. 2, p. 100015, 2020.
- [4] G. Notton and C. Voyant, "Forecasting of intermittent solar energy resource," in *Advances in Renewable Energies and Power Technologies*. Elsevier, 2018, pp. 77–114.

- [5] G. Energy *et al.*, “Western wind and solar integration study,” Citeseer, Tech. Rep., 2010.
- [6] A. Taik and S. Cherkaoui, “Electrical load forecasting using edge computing and federated learning,” in *ICC 2020-2020 IEEE international conference on communications (ICC)*. IEEE, 2020, pp. 1–6.
- [7] N. Truong, K. Sun, S. Wang, F. Guitton, and Y. Guo, “Privacy preservation in federated learning: An insightful survey from the gdp perspective,” *Computers & Security*, vol. 110, p. 102402, 2021.
- [8] M. Worley. Ibm security x-force threat intelligence index 2023. [Online]. Available: <https://www.ibm.com/downloads/cas/DB4GL8YM>
- [9] S. M. Kerner. Colonial pipeline hack explained: Everything you need to know. [Online]. Available: <https://www.techtarget.com/whatis/feature/Colonial-Pipeline-hack-explained-Everything-you-need-to-know>
- [10] J. Greig. Suspected china-backed hackers target 7 indian electricity grid centers. [Online]. Available: <https://therecord.media/suspected-china-backed-hackers-target-7-indian-electricity-grid-centers>
- [11] W. Hurst, C. A. C. Montañez, and N. Shone, “Time-pattern profiling from smart meter data to detect outliers in energy consumption,” *IoT*, vol. 1, no. 1, p. 6, 2020.
- [12] E. Skomski, J.-Y. Lee, W. Kim, V. Chandan, S. Katipamula, and B. Hutchinson, “Sequence-to-sequence neural networks for short-term electrical load forecasting in commercial office buildings,” *Energy and Buildings*, vol. 226, p. 110350, 2020.
- [13] S. Chen, Y. Ren, D. Friedrich, Z. Yu, and J. Yu, “Prediction of office building electricity demand using artificial neural network by splitting the time horizon for different occupancy rates,” *Energy and AI*, vol. 5, p. 100093, 2021.
- [14] 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.
- [15] M. N. Fekri, K. Grolinger, and S. Mir, “Distributed load forecasting using smart meter data: Federated learning with recurrent neural networks,” *International Journal of Electrical Power & Energy Systems*, vol. 137, p. 107669, 2022.
- [16] “Ieee guide for architectural framework and application of federated machine learning,” *IEEE Std 3652.1-2020*, pp. 1–69, 2021.
- [17] N. Gholizadeh and P. Musilek, “Distributed learning applications in power systems: A review of methods, gaps, and challenges,” *Energies*, vol. 14, no. 12, p. 3654, 2021.
- [18] F. M. A. Khan, H. Abou-Zeid, A. Kaushik, and S. A. Hassan, “Advancing iiot with over-the-air federated learning: The role of iterative magnitude pruning,” *IEEE Internet of Things Magazine*, vol. 7, no. 5, pp. 46–52, 2024.
- [19] H. U. Manzoor, A. Shabbir, A. Chen, D. Flynn, and A. Zoha, “A survey of security strategies in federated learning: Defending models, data, and privacy,” *Future Internet*, vol. 16, no. 10, p. 374, 2024.
- [20] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, “How to backdoor federated learning,” in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020, pp. 2938–2948.
- [21] H. Wang, K. Sreenivasan, S. Rajput, H. Vishwakarma, S. Agarwal, J.-y. Sohn, K. Lee, and D. Papailiopoulos, “Attack of the tails: Yes, you really can backdoor federated learning,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 16070–16084, 2020.
- [22] A. N. Bhagoji, S. Chakraborty, P. Mittal, and S. Calo, “Analyzing federated learning through an adversarial lens,” in *International Conference on Machine Learning*. PMLR, 2019, pp. 634–643.
- [23] C. Xie, K. Huang, P.-Y. Chen, and B. Li, “Dba: Distributed backdoor attacks against federated learning,” in *International conference on learning representations*, 2019.
- [24] Z. Yin, Y. Yuan, P. Guo, and P. Zhou, “Backdoor attacks on federated learning with lottery ticket hypothesis,” *arXiv preprint arXiv:2109.10512*, 2021.
- [25] A. Shabbir, H. U. Manzoor, R. A. Ahmed, and Z. Halim, “Resilience of federated learning against false data injection attacks in energy forecasting,” in *2024 International Conference on Green Energy, Computing and Sustainable Technology (GECOST)*. IEEE, 2024, pp. 245–249.
- [26] A. Shabbir, H. U. Manzoor, K. Arshad, K. Assaleh, Z. Halim, and A. Zoha, “Sustainable and lightweight defense framework for resource constraint federated learning assisted smart grids against adversarial attacks,” *Authorea Preprints*, 2024.
- [27] H. U. Manzoor, A. R. Khan, M. Al-Quraan, L. Mohjazi, A. Taha, H. Abbas, S. Hussain, M. A. Imran, and A. Zoha, “Energy management in an agile workspace using ai-driven forecasting and anomaly detection,” in *2022 4th Global Power, Energy and Communication Conference (GPECOM)*. IEEE, 2022, pp. 644–649.
- [28] H. U. Manzoor, M. S. Khan, A. R. Khan, F. Ayaz, D. Flynn, M. A. Imran, and A. Zoha, “Fedclamp: An algorithm for identification of anomalous client in federated learning,” in *2022 29th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*. IEEE, 2022, pp. 1–4.
- [29] H. U. Manzoor, A. R. Khan, T. Sher, W. Ahmad, and A. Zoha, “Defending federated learning from backdoor attacks: Anomaly-aware fedavg with layer-based aggregation,” in *2023 IEEE 34th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. IEEE, 2023, pp. 1–6.
- [30] H. U. Manzoor, S. Hussain, D. Flynn, and A. Zoha, “Centralised vs. decentralised federated load forecasting in smart buildings: Who holds the key to adversarial attack robustness?” *Energy and Buildings*, vol. 324, p. 114871, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0378778824009873>
- [31] P. Liu, X. Xu, and W. Wang, “Threats, attacks and defenses to federated learning: issues, taxonomy and perspectives,” *Cybersecurity*, vol. 5, no. 1, pp. 1–19, 2022.
- [32] H. S. Sikandar, H. Waheed, S. Tahir, S. U. Malik, and W. Rafique, “A detailed survey on federated learning attacks and defenses,” *Electronics*, vol. 12, no. 2, p. 260, 2023.
- [33] Y. Chen, Y. Gui, H. Lin, W. Gan, and Y. Wu, “Federated learning attacks and defenses: A survey,” *arXiv preprint arXiv:2211.14952*, 2022.
- [34] C. Fung, C. J. Yoon, and I. Beschastnikh, “Mitigating sybils in federated learning poisoning,” *arXiv preprint arXiv:1808.04866*, 2018.
- [35] 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*, 2016, pp. 508–519.
- [36] Z. Sun, P. Kairouz, A. T. Suresh, and H. B. McMahan, “Can you really backdoor federated learning?” *arXiv preprint arXiv:1911.07963*, 2019.
- [37] C. Wu, S. Zhu, and P. Mitra, “Federated unlearning with knowledge distillation,” *arXiv preprint arXiv:2011.09441*, 2022.
- [38] C. Wu, X. Yang, S. Zhu, and P. Mitra, “Mitigating backdoor attacks in federated learning,” *arXiv preprint arXiv:2011.01767*, 2020.
- [39] T. L. Hayes, K. Kafle, R. Shrestha, M. Acharya, and C. Kanan, “Remind your neural network to prevent catastrophic forgetting,” in *European conference on computer vision*. Springer, 2020, pp. 466–483.
- [40] Q. Yang, L. Fan, R. Tong, and A. Lv, “White paper-ieee federated machine learning,” *IEEE Federated Machine Learning-White Paper*, pp. 1–18, 2021.
- [41] O. Zari, C. Xu, and G. Neglia, “Efficient passive membership inference attack in federated learning,” *arXiv preprint arXiv:2111.00430*, 2021.
- [42] 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)*. IEEE, 2019, pp. 739–753.
- [43] V. Shejwalkar and A. Houmansadr, “Manipulating the byzantine: Optimizing model poisoning attacks and defenses for federated learning,” in *NDSS*, 2021.
- [44] T. Kim, S. Singh, N. Madaan, and C. Joe-Wong, “Characterizing internal evasion attacks in federated learning,” in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2023, pp. 907–921.
- [45] K. N. Kumar, C. Vishnu, R. Mitra, and C. K. Mohan, “Black-box adversarial attacks in autonomous vehicle technology,” in *2020 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*. IEEE, 2020, pp. 1–7.
- [46] H. U. Manzoor, K. Arshad, K. Assaleh, and A. Zoha, “Enhanced adversarial attack resilience in energy networks through energy and privacy aware federated learning,” *Authorea Preprints*, 2024.
- [47] R. Mulla, “Hourly energy consumption.” [Online]. Available: <https://www.kaggle.com/datasets/robikscube/hourly-energy-consumption/data>.
- [48] H. U. Manzoor, A. R. Khan, D. Flynn, M. M. Alam, M. Akram, M. A. Imran, and A. Zoha, “Fedbranded: Leveraging federated learning for anomaly-aware load forecasting in energy networks,” *Sensors*, vol. 23, no. 7, p. 3570, 2023.
- [49] H. U. Manzoor, A. Jafri, and A. Zoha, “Adaptive single-layer aggregation framework for energy-efficient and privacy-preserving load forecasting in heterogeneous federated smart grids,” *Internet of Things*, p. 101376, 2024.