Advancing Network Anomaly Detection Using Deep Learning and Federated Learning in an Interconnected Environment

Hanen Dhrir[1], Maha Charfeddine[2] and Habib M. Kammoun[3]
[1]*Data Engineering and Semantics Research Unit, Faculty of Sciences of Sfax, Sfax, Tunisia*
[2]*REGIM-Lab: REsearch Groups in Intelligent Machines, National Engineering School of Sfax, Sfax, Tunisia*
[3]*REGIM-Lab: REsearch Groups in Intelligent Machines, Faculty of Sciences of Sfax, Sfax, Tunisia*

Keywords: Anomaly Detection, Federated Learning, Deep Learning, Network Security, Privacy.

Abstract: Network anomaly detection is a fundamental cybersecurity task that seeks to identify unusual patterns that could indicate security threats or system failures. Traditional centralized anomaly detection methods face issues such as data privacy. Federated Learning has emerged as a promising solution that distributes model training across multiple devices or nodes. Federated Learning improves anomaly detection by leveraging geographically distributed data sources while maintaining data privacy and security. This study presents a novel Federated Learning architecture designed specifically for network anomaly detection, addressing important information sensitivity issues in network environments. We compare some Deep Learning algorithms, such as Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN), and Multilayer Perceptron (MLP), using XGBoost for feature selection and Stochastic Gradient Descent (SGD) as an optimizer. To address the problem of imbalanced data, we use the Synthetic Minority Over-sampling Technique (SMOTE) with the UNSW-NB15 dataset. Our methodology is rigorously evaluated using standard evaluation metrics and compared to state-of-the-art approaches.

1 INTRODUCTION

Network anomaly detection is essential for securing digital infrastructure, preventing unauthorized access, and mitigating cyber threats. Traditional centralized methods face challenges in handling vast data, adapting to evolving threats, and operating in real-time, all while raising privacy concerns by aggregating sensitive data in a single location. This approach jeopardizes data confidentiality and demands extensive datasets to effectively capture network behaviors, often failing to scale and respond to advanced risks (Garg et al., 2020). Federated Learning offers a promising alternative by decentralizing model training across multiple devices or nodes. It keeps data local, sharing only model updates aggregated on a central server to create a global model. By leveraging geographically distributed data, Federated Learning enhances anomaly detection while maintaining data privacy and security. Each entity trains models locally and shares updates, reducing data breach risks. This approach addresses privacy issues and improves detection system robustness and accuracy through diverse data sources, enhancing model resilience (Aburomman and Reaz, 2016). The main contributions of this research work are as follows:

Introduction of a Novel Federated Learning Architecture: We propose a novel Federated Learning architecture specifically designed for network anomaly detection. This architecture ensures data privacy and security, addressing the important challenge of data sensitivity in network environments.

Evaluation of various Deep Learning Algorithms and Machine Learning Methods: Our research thoroughly evaluates several Deep Learning (DL) models, including Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN), and Multilayer Perceptron (MLP), employs XGBoost for feature selection, and applies Stochastic Gradient Descent (SGD) as an optimizer. These models are assessed for their effectiveness in accurately detecting network anomalies.

Application of SMOTE for Data Balancing: To address the challenge of imbalanced data, we apply the Synthetic Minority Over-sampling Technique (SMOTE) to the UNSW-NB15 dataset. This technique improves the performance of our models by ensuring a more balanced representation of the dataset.

Dhrir, H., Charfeddine, M. and Kammoun, H. M.
Advancing Network Anomaly Detection Using Deep Learning and Federated Learning in an Interconnected Environment.
DOI: 10.5220/0013134100003928

In *Proceedings of the 20th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE 2025)*, pages 343-350
ISBN: 978-989-758-742-9; ISSN: 2184-4895

Comprehensive Assessment Using Standard Evaluation Metrics: The proposed methodology is rigorously evaluated using the most common evaluation parameters. Our results are benchmarked against other state-of-the-art approaches, demonstrating the efficacy of our methods. The structure of this paper is as follows: Section 2, Theoretical Background, covers the foundations of Federated Learning and relevant Deep Learning methods for network anomaly detection. Section 3, Related Work, reviews relevant research and finding. Section 4, Methodology, describes the proposed Federated Learning approach for network anomaly detection, with an emphasis on data privacy. Section 5, Experiments and Results, presents the experimental setup and results, including a comparison with previous related works. Finally, Section 6, concludes the paper.

2 THEORETICAL BACKGROUND

This section covers the key concepts of Machine Learning, Deep Learning, and Federated Learning for network anomaly detection. It highlights how Federated Learning addresses the limitations of traditional methods by preserving privacy and decentralizing data, leading to more reliable and secure anomaly detection solutions, and includes a description of the Deep Learning algorithms used in our system.

2.1 Network Anomaly Detection

Network anomaly detection is a critical technique for identifying unusual network activity that could indicate security threats or breaches. It involves monitoring network traffic for deviations from normal patterns, signaling potential malicious actions or system failures. Historically, anomaly detection has relied on two approaches: signature-based and anomaly-based methods. Signature-based detection identifies known threats by comparing network activity to predefined patterns or signatures. While effective for known threats, it is limited in detecting new or unknown threats. On the other hand, anomaly-based detection monitors network behavior, identifying deviations from established norms. While capable of detecting novel threats, it often struggles with the need for comprehensive training data (Hdaib et al., 2024). To enhance these methods, Machine Learning (ML) and Deep Learning (DL) approaches are currently widely used. These techniques improve detection by Learning complex patterns and anomalies, overcoming the limitations of traditional methods. ML tools, such as classifiers, segment data into distinct categories like normal and abnormal, training on labeled datasets to detect patterns and classify new data points, recognizing outliers (Rafique et al., 2024). Autoencoders, a type of neural network, excel at this task by compressing input data and reconstructing it, Learning normal data patterns and detecting anomalies by comparing the reconstructed output with the original (Torabi et al., 2023). Additionally, Deep Learning models like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) automatically extract relevant features from large, unstructured datasets, improving anomaly detection by capturing intricate patterns and temporal dependencies (Lakey and Schlippe, 2024). While DL techniques offer significant advancements in network anomaly detection, they also raise privacy concerns. These methods often require centralizing large datasets for model training, which can lead to privacy breaches if sensitive data is not adequately protected.

2.2 Federated Learning for Network Anomaly Detection

Federated Learning is a revolutionary approach to machine Learning, providing a decentralized framework that enables multiple entities to collaboratively train a model while keeping their data local. Unlike traditional centralized methods, Federated Learning improves data privacy and security by keeping sensitive information in its original environment. Rather than aggregating raw data, this method gathers and combines model updates such as gradients or parameters—from each participant. This decentralized aggregation reduces the risk of data breaches and ensures compliance with stringent data protection regulations. Federated Learning's key principles include collaborative training across distributed nodes, secure aggregation of model updates, and robust communication protocols. Federated Learning improves anomaly detection across network environments by aggregating model updates from a variety of decentralized sources. This approach uses a broader set of data while maintaining individual privacy, resulting in more accurate and resilient detection systems that overcome the limitations of centralized methods(Bharati et al., 2022).

2.2.1 Architecture of Federated Learning for Network Anomaly Detection

Federated Learning is intended to handle decentralized data sources and strict privacy requirements. It enables multiple entities, such as devices or servers, to participate in model training without transmitting

raw data. The architecture depicted in figure 1 has three major components (Zhao et al., 2019):

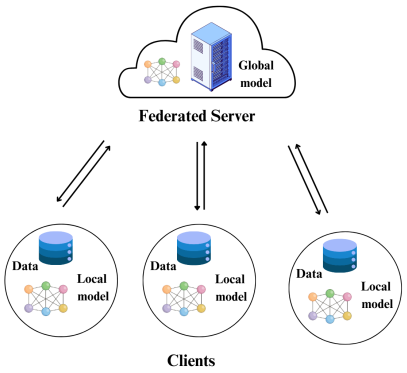

Figure 1: Architecture of Federated Learning.

- **Server as a Central Coordinator:** The server manages the global model, initialized with parameters from a pre-existing dataset. It selects a subset of client devices for each training iteration based on performance or availability, ensuring diverse data sources contribute efficiently to the global model.
- **Client Devices and Local Computation:** Each client device uses its private dataset to compute model updates based on the global model. These updates, encrypted or anonymized for privacy, are transmitted to the server without exposing raw data.
- **Secure Aggregation:** The server aggregates encrypted updates using methods like Federated averaging or secure multi-party computation (MPC). Weighted strategies may prioritize significant client contributions, improving model accuracy and robustness. The updated global model is then redistributed to clients, enabling iterative refinement for anomaly detection across diverse data.

2.2.2 Privacy Measures in Federated Learning

Federated Learning incorporates a number of privacy-preserving features throughout its decentralized framework. Data is distributed across client devices to avoid the centralization of sensitive information. Clients compute and send model updates in a secure, encrypted format, which the server aggregates using privacy-preserving methods. Furthermore, differential privacy techniques may be used to add noise to model updates, thereby protecting individual data points. The client selection procedure is randomized to guarantee diverse participation and avoid disproportionate disclosure of any single client's data (Lyu et al., 2022). The described Federated Learning architecture provides a strong foundation for collaborative, privacy-preserving anomaly detection in decentralized environments.

2.3 Deep Learning Algorithms

This section explores the three Deep Learning algorithms used in our system: Multilayer Perceptron (MLP), Long Short-Term Memory (LSTM) networks, and Convolutional Neural Networks (CNNs). Each algorithm is designed to handle different types of data, enhancing the detection and analysis of anomalies in network traffic, sequential data, and grid-like data structures. We detail how each is applied, focusing on their role in improving the accuracy and performance of our predictive models.

The Multilayer Perceptron (MLP): We use a multilayer perceptron (MLP), a type of artificial neural network (ANN) commonly applied to supervised Learning tasks like network anomaly detection. The MLP consists of perceptrons that compute the dot product of input features with a weight matrix, which contains trainable parameters. Our architecture includes three fully connected layers: two hidden layers with 256 units each, activated by ReLU, and an output layer that uses a sigmoid activation function for binary classification, predicting class probabilities.

Long Short-Term Memory (LSTM): Long Short-Term Memory (LSTM) networks handle sequential data by capturing long-term dependencies through specialized LSTM cells with input, forget, and output gates. Our setup includes one LSTM layer, two dense layers, and a final sigmoid layer, enabling efficient processing of time series data and pattern recognition.

Convolutional Neural Network (CNN): Our system also employs Convolutional Neural Networks, which excel at processing and analyzing grid-like data structures, such as images. CNNs are composed of several layers, including convolutional layers that apply filters to the input data to extract indispensable features like edges and textures in images. These convolutional layers are followed by pooling layers that reduce dimensionality and computational complexity by downsampling the feature maps. The architecture also integrates fully connected layers that use the flattened output from the convolutional and pooling layers to make final predictions. Through the use of convolutional and pooling operations, CNNs effectively capture spatial hierarchies and patterns in the data, making them ideal for tasks like image classification and object detection. The following section will go over the detailed methodology, which includes the al-

gorithms, datasets, and evaluation metrics used to implement and evaluate the effectiveness of this architecture in real-world network anomaly detection scenarios.

3 RELATED WORK

This section reviews significant contributions in the field of Federated Learning as applied to network anomaly detection, highlighting innovative approaches and methodologies that address the unique challenges of this domain. The emphasis is on adaptive Federated Learning frameworks and hierarchical architectures that enhance system robustness and data privacy. Authors in (Doriguzzi-Corin and Siracusa, 2024) introduced FLAD, an adaptive Federated Learning Approach specifically designed for DDoS attack detection. This approach aims to overcome the limitations of traditional Federated Learning algorithms by dynamically adjusting client selection and computational workloads during the training process, without necessitating the exchange of training or validation data between clients and the central server. FLAD is tailored for cybersecurity applications, with a particular focus on maintaining data privacy by ensuring that sensitive attack data is not shared between clients and the server. This method utilizes the CIC-DDoS2019 dataset (Saheb et al., 2021), which provides a robust framework for evaluating DDoS detection capabilities under a Federated Learning paradigm. The paper introduces FLAD, an adaptive Federated Learning mechanism that improves upon traditional methods like FEDAVG by dynamically tuning the Federated training process based on client performance. In (Marfo et al., 2022), the authors explore a Federated Learning architecture that incorporates a hierarchical setup involving clients, edge servers, and a global server. Clients retain their data locally and manage their models independently, thereby safeguarding data privacy by preventing direct data transfers to central servers. Edge servers play a critical role in this architecture by receiving model updates from clients, aggregating these updates, and forwarding the consolidated results to the global server. The global server then refines the global model using these aggregated updates and disseminates the improved model back to clients for local updates. This hierarchical structure, which introduces intermediate layers such as edge servers, distributes the computational load and enhances system reliability by mitigating the impact of potential failures. This tiered approach not only improves scalability but also increases the resilience of Federated Learning systems. The application of Multilayer Perceptron (MLP) within this framework is particularly effective for tasks like network anomaly detection, as demonstrated using the UNSW-NB15 dataset, which provides comprehensive coverage of network intrusion scenarios. These innovative approaches in Federated Learning for network anomaly detection underscore the importance of adaptive frameworks and hierarchical architectures in enhancing privacy, scalability, and system resilience. In (Priyadarshini, 2024), to improve anomaly detection in IoT environments, the authors use the UNSW-NB15 dataset and the Split Learning (SL) model. Split Learning maintains privacy by enabling the model to be trained on several devices without transferring raw data. This technique divides the model into multiple components, each of which is trained on a separate device. This lowers the computational load while enhancing security. The UNSW-NB15 dataset, which is renowned for its extensive records of network traffic, facilitates the efficient identification of cyberthreats within the infrastructure of smart cities. In the following section, we delve Deeper into the proposed methodology, further exploring their implication for robust and efficient anomaly detection.

4 METHODOLOGY

This section describes how we developed and evaluated our Federated Learning-based network anomaly detection system. Our methodology ensures a thorough and rigorous assessment of various Deep Learning models, incorporating feature selection and balancing techniques to enhance model performance and robustness, as illustrated in figure 2.

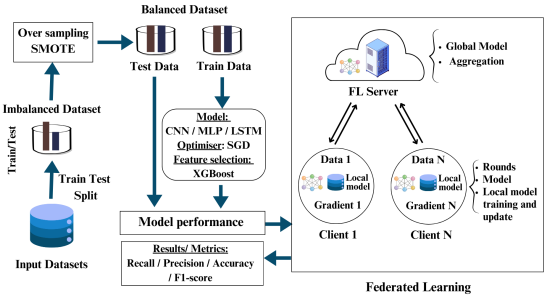

Figure 2: Overview of the proposed methodology.

4.1 Federated Network Anomaly Detection Framework

We employed Federated Learning to promote collaborative model training in a decentralized manner across ten clients. Each client keeps its local data, ensuring privacy, and sends model updates to the central server through fifty communication rounds, as illustrated in Figure 2. The experiments were conducted out on the UNSW-NB15 dataset, which was balanced using the SMOTE technique and then partitioned across ten clients. We explored various scenarios by implementing and training CNN, LSTM, and MLP models for each client. To achieve optimal performance in our Federated Learning-based system, we used Stochastic Gradient Descent (SGD) as an optimizer and XGBoost for feature selection.

4.2 Dataset Description

The UNSW-NB15 dataset (Moustafa and Slay, 2015), developed by the ACCS at UNSW, is used for training and evaluating network anomaly detection models. It covers nine attack types and includes 49 features from network traffic. With around 2.5 million records, it provides labeled data for both normal and malicious activity. The dataset is publicly available for research to enhance network security and intrusion detection systems.

4.3 Preprocessing Steps

Preprocessing plays a vital role in enhancing the performance of Machine Learning models. The key steps include:

Handling Missing Values: Missing values, often resulting from data corruption or improper recording, are addressed by removing rows with NaN values, negative infinity (-inf), and duplicates. **Feature Scaling**: Standardization is applied to rescale numerical features to have a mean of zero and a standard deviation of one, improving model accuracy and performance (Fki et al., 2024).

Label Encoding: Categorical features are converted into numerical values to make them suitable for Machine Learning models.

Class Imbalance Processing: The Synthetic Minority Over-sampling Technique (SMOTE) addresses class imbalance by generating synthetic examples for the minority class. It creates new instances in the feature space, improving the model's ability to classify minority classes (Ali et al., 2024a).

4.4 Tested Learning Algorithms

Using MLP, LSTM, and CNN architectures, we conducted 50 rounds with 10 epochs for each of the 10 clients. Through extensive testing, we determined that 10 epochs yielded the best results. Additionally, we split the dataset into 80% for training and 20% for testing. To optimize the proposed models, we use the Stochastic Gradient Descent (SGD) optimizer. This prominent algorithm in Machine Learning is particularly effective for training models such as neural networks. Unlike traditional gradient descent, which calculates the gradient using the entire dataset, SGD updates model parameters iteratively using small, randomly selected batches of data. This approach significantly enhances computational efficiency and speed. The frequent updates facilitated by mini-batches enable faster convergence and improve the model's ability to generalize, thereby reducing the risk of overfitting. Additionally, the stochastic nature of SGD helps the optimizer escape local minima, making it advantageous for handling large-scale datasets. Overall, SGD is a fundamental tool for scalable and robust model training (Sun et al., 2022).

4.5 Tested Feature Selection

Among the best features, we use XGBoost feature selection for identifying the most significant features within a dataset. XGBoost intrinsically evaluates feature importance by analyzing each feature's contribution to the model's predictive accuracy during training. It measures metrics such as gain, coverage, and frequency. This process enables the identification of the most influential features, facilitating dimensionality reduction, enhancing model performance, and lowering computational costs by focusing on the features that significantly impact prediction outcomes (Chen and Guestrin, 2016).

4.6 Evaluation Metrics

In anomaly detection, evaluating a model involves metrics like recall, precision, accuracy, and the F1 score to assess its ability to identify anomalies.

- Recall is the metric that evaluates a model's ability to correctly identify true anomalies, minimizing false negatives. Recall is calculated using the formula (1):

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{1}$$

- Accuracy measures the overall correctness of a model's predictions, considering both true posi-

tives and true negatives. The formula (2) for accuracy, considering true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN), is:

$$\text{Accuracy} = \frac{\text{TN} + \text{TP}}{\text{TP} + \text{FN} + \text{FP} + \text{TN}} \quad (2)$$

.

- The F1-Score, the harmonic mean of precision and recall, provides a balanced evaluation of model performance, especially in imbalanced datasets, by balancing false alarms and anomaly detection. The formula (3) calculates the F1-Score:

$$\text{F1-Score} = \frac{2 \times (\text{Recall} \times \text{Precision})}{\text{Recall} + \text{Precision}} \quad (3)$$

- Precision measures the proportion of flagged anomalies that are true anomalies, minimizing false positives (Ali et al., 2024b). The formula for precision is (4):

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (4)$$

5 EXPERIMENTS AND RESULTS

For validating the methodology explained in the previous section, we conducted various experiments as part of our research, using the SMOTE to balance the dataset and the SGD architecture to optimize the models' performance. We investigated three DL models: MLP, LSTM, and CNN, while also employing XGBoost for feature selection. We ran 50 rounds with 10 epochs for each of the 10 clients across the three combinations. Our Federated Learning-based experiments yielded the following summarized results: First, we present our experiments on the UNSW-NB15 dataset (Moustafa and Slay, 2015) and then, we compare our findings with those from other studies. As can be seen from the table 1, within the proposed Federated Learning architecture, the CNN model with SGD outperforms the other models (MLP and LSTM) using the same optimizer in terms of recall, accuracy, precision, and F1-Score. This proves that CNN model with SGD is very good at finding positive examples with low error rates. The CNN is slightly superior to the LSTM model with SGD in terms of precision and recall, but both models still perform well. With the lowest recall, precision, accuracy, and F1-Score among the models, the MLP model with SGD, on the other hand, performs worse than the other two models. In addition, we displayed some plotted curves 3, 4 and 5 for the tested combined approaches. In our Federated Learning setup, a "round" is a full cycle in which each client performs local training on its data and sends updates to the central server. The process is repeated 50 rounds, allowing each model to iteratively improve while Learning from decentralized data sources. In figures 3, 4 and 5, the X-axis represents the number of communication rounds (ranging from 1 to 50). The Y-axis represents the metric values (recall, precision, accuracy, F1-score, and loss) used.These metrics for each model are plotted to show how the model's performance changes with each subsequent round of training. From what we observed in general for figures 3, 4 and 5, as the rounds increase, recall improves because the model becomes more adept at identifying true positives, Learning from decentralized data from multiple clients. The MLP's recall begins low and improves marginally, indicating a struggle to consistently identify anomalies. CNN increases significantly over time, rapidly reaching high recall values, indicating its effectiveness in detecting anomalies. LSTM gradually evolves, but not as quickly or effectively as CNN, most likely due to its sequential data processing capabilities, which are less suited to this specific problem. Precision also tends to improve with rounds as the model reduces false positives, thus enhancing its ability to correctly classify anomalies. Precision increases slightly for the MLP model, but remains relatively low, indicating a challenge in reducing false positives. CNN achieves high precision quickly and maintains it throughout the rounds, demonstrating its strong ability in accurate anomaly detection. LSTM advances with each round but falls short of CNN's precision, implying that some false positives remain. Accuracy improves as the model becomes more effective at identifying both true positives and true negatives. MLP increases slightly but stands the lowest of the models, indicating lower overall performance. CNN achieves high accuracy early in the rounds, and maintains it throughout, making it the most effective model. LSTM improves steadily, achieving decent accuracy but still falling behind CNN. As precision and recall improve, the F1-score generally rises. The F1-score for the MLP algorithm is relatively low and increases slightly, indicating that the model faces recall and precision challenges. CNN quickly achieves a high F1-score, demonstrating a good balance between precision and recall. LSTM raises but does not match CNN, indicating unbalanced gains in precision and recall. Loss typically decreases as the model becomes more accurate, reducing the difference between predicted and actual results. MLP's loss decreases slightly but remains higher than that of other models, showing poorer performance. CNN promptly achieves low

loss values, indicating effective Learning and convergence. Loss decreases steadily with LSTM but does not reach the minimal values achieved by CNN. Among the three models, CNN with SGD optimization performs the best in anomaly detection within the proposed Federated Learning framework. It consistently achieves high recall, precision, accuracy, and F1-score while reducing loss. LSTM performs reasonably well but is less effective than CNN, most likely due to its sequential processing nature not being fully utilized in this particular context. MLP struggles the most, implying that it may not be appropriate for such complex anomaly detection tasks in a Federated interconnected setting.

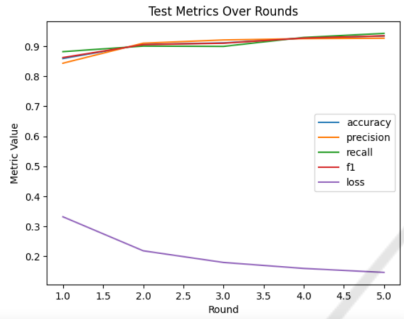

Figure 3: Test metrics over rounds for MLP architecture and SGD optimiser.

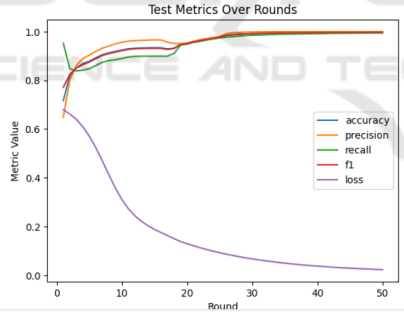

Figure 4: Test metrics over rounds for CNN architecture and SGD optimiser.

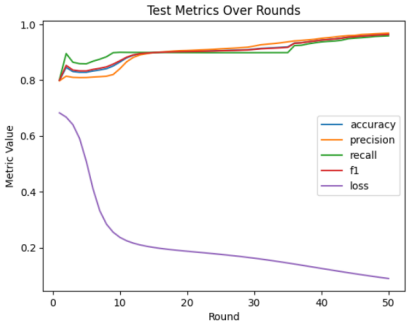

Figure 5: Test metrics over rounds for LSTM architecture and SGD optimiser.

Table 1: Experiments on the UNSW-NB15 Dataset.

Experiment	Recall	Precision	Accuracy	F1-Score
MLP+SGD	0.9085	0.8939	0.8903	0.9012
CNN+SGD	0.9939	0.9998	0.9965	0.9968
LSTM+SGD	0.9492	0.9610	0.9508	0.9550

- **Comparative Analysis with Related Works**

Table 2: Comparative Analysis with Related Works.

Experiment	Recall	Precision	Accuracy	F1-Score
SL (Priyadarshini, 2024)	0.9802	0.9812	0.9802	0.9811
MLP + Smote (Marfo et al., 2022)	0.9718	0.9809	0.9721	0.9763
CNN+SGD (our)	0.9939	0.9998	0.9965	0.9968

The table 2 compares our best proposed Federated Learning-based network anomaly detection (CNN+SGD) to various related previous approaches while considering the four common metrics recall, precision, accuracy, and F1-scores. The SL (Priyadarshini, 2024) solution performs well and consistently across all metrics, with a recall of 0.9802, precision of 0.9812, accuracy of 0.9802, and F1-score of 0.9811. SMOTE combined with the MLP model (Marfo et al., 2022) yields a recall of 0.9718, precision of 0.9809, accuracy of 0.9721, and F1-score of 0.9763. Our best solution combining the CNN, SGD, SMOTE with XGBoost for feature selection outperforms all other related existing studies, with a recall of 0.9939, precision of 0.9998, accuracy of 0.9965, and F1-score of 0.9968. These findings demonstrate that combining a CNN with SGD, adequate features using XGBoost, and SMOTE within a Federated Learning architecture is extremely effective.

6 CONCLUSION

This study compared the performance of various Deep Learning models in detecting network anomalies within a secure Federated Learning environment. To enhance model performance, we utilized XGBoost for feature selection, SMOTE for dataset balancing, and SGD for model optimization. The findings show that when it comes to recall, precision, accuracy, and F1-score, CNN combined with SGD, SMOTE, and XGBoost perform noticeably better than other possible combinations. In particular, the CNN model demonstrated almost flawless classification performance with recall of 0.9939, precision of 0.9998, accuracy of 0.9965, and F1-score of 0.9968. The performance of the LSTM and MLP models was enhanced by the addition of SMOTE, underscoring the

significance of correcting the dataset's class imbalance. However, the feature selection model using MLP, SGD, and SMOTE performed relatively poorly, indicating that this combination might not be as useful in the network anomaly specific context. Overall, the results indicate that feature selection with a CNN equipped with SGD, SMOTE, and XGBoost is very successful when applied to network anomaly detection tasks within a Federated Anomaly environment. To improve the results' generalizability, future research ought to investigate into other optimization techniques for these models and their applications across various domains. Furthermore, testing our experiments on different datasets would provide a broader overview of the model's performance.

ACKNOWLEDGEMENTS

The research leading to these results was supported by the Ministry of Higher Education and Scientific Research of Tunisia.

REFERENCES

Aburomman, A. A. and Reaz, M. B. I. (2016). A novel svm-knn-pso ensemble method for intrusion detection system. *Applied Soft Computing*, 38:360–372.

Ali, A. H., Charfeddine, M., Ammar, B., and Hamed, B. B. (2024a). Intrusion detection schemes based on synthetic minority oversampling technique and machine learning models. In *2024 IEEE 27th International Symposium on Real-Time Distributed Computing (ISORC)*, pages 1–8. IEEE.

Ali, A. H., Charfeddine, M., Ammar, B., Hamed, B. B., Albalwy, F., Alqarafi, A., and Hussain, A. (2024b). Unveiling machine learning strategies and considerations in intrusion detection systems: a comprehensive survey. *Frontiers in Computer Science*, 6:1387354.

Bharati, S., Mondal, M., Podder, P., and Prasath, V. (2022). Federated learning: Applications, challenges and future directions. *International Journal of Hybrid Intelligent Systems*, 18(1-2):19–35.

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794.

Doriguzzi-Corin, R. and Siracusa, D. (2024). Flad: adaptive federated learning for ddos attack detection. *Computers & Security*, 137:103597.

Fki, Z., Ammar, B., Fourati, R., Fendri, H., Hussain, A., and Ben Ayed, M. (2024). A novel iot-based deep neural network for covid-19 detection using a soft-attention mechanism. *Multimedia Tools and Applications*, 83(18):54989–55009.

Garg, S., Kaur, K., Batra, S., Kaddoum, G., Kumar, N., and Boukerche, A. (2020). A multi-stage anomaly detection scheme for augmenting the security in iot-enabled applications. *Future Generation Computer Systems*, 104:105–118.

Hdaib, M., Rajasegarar, S., and Pan, L. (2024). Quantum deep learning-based anomaly detection for enhanced network security. *Quantum Machine Intelligence*, 6(1):26.

Lakey, D. and Schlippe, T. (2024). A comparison of deep learning architectures for spacecraft anomaly detection. In *2024 IEEE Aerospace Conference*, pages 1–11. IEEE.

Lyu, L., Yu, H., Ma, X., Chen, C., Sun, L., Zhao, J., Yang, Q., and Philip, S. Y. (2022). Privacy and robustness in federated learning: Attacks and defenses. *IEEE transactions on neural networks and learning systems*.

Marfo, W., Tosh, D. K., and Moore, S. V. (2022). Network anomaly detection using federated learning. In *MILCOM 2022-2022 IEEE Military Communications Conference (MILCOM)*, pages 484–489. IEEE.

Moustafa, N. and Slay, J. (2015). Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set). In *2015 military communications and information systems conference (MilCIS)*, pages 1–6. IEEE.

Priyadarshini, I. (2024). Anomaly detection of iot cyberattacks in smart cities using federated learning and split learning. *Big Data and Cognitive Computing*, 8(3):21.

Rafique, S. H., Abdallah, A., Musa, N. S., and Murugan, T. (2024). Machine learning and deep learning techniques for internet of things network anomaly detection—current research trends. *Sensors*, 24(6):1968.

Saheb, M. C. P., Yadav, M. S., Babu, S., Pujari, J. J., and Maddala, J. B. (2021). A review of ddos evaluation dataset: Cicddos2019 dataset. In *International Conference on Energy Systems, Drives and Automations*, pages 389–397. Springer.

Sun, Y., Que, H., Cai, Q., Zhao, J., Li, J., Kong, Z., and Wang, S. (2022). Borderline smote algorithm and feature selection-based network anomalies detection strategy. *Energies*, 15(13):4751.

Torabi, H., Mirtaheri, S. L., and Greco, S. (2023). Practical autoencoder based anomaly detection by using vector reconstruction error. *Cybersecurity*, 6(1):1.

Zhao, Y., Chen, J., Wu, D., Teng, J., and Yu, S. (2019). Multi-task network anomaly detection using federated learning. In *Proceedings of the 10th international symposium on information and communication technology*, pages 273–279.