

An Adaptive and Interpretable Federated Deep Learning Architecture for Privacy-Aware Network Threat Classification

Dr. M. Nagaraju Naik¹ | Prof. M. Padmavathamma²

¹ Research Scholar, Department of Computer Science,
Sri Venkateswara University, Tirupati, Andhra Pradesh, India

² Research Supervisor, Department of Computer Science & Dean, Faculty of Sciences, Sri Venkateswara University, Tirupati,
Andhra Pradesh, India

Galla Narayana ³

³ Research Scholar, Department of Computer Science,
Sri Venkateswara University, Tirupati, Andhra Pradesh, India

Abstract—The increasing distribution of enterprise computing across cloud platforms, edge devices, branch networks, and Internet-of-Things environments has made the collection and centralized analysis of network telemetry increasingly difficult. Conventional deep-learning intrusion detection approaches commonly require the movement of traffic records to a common training repository, which can expose sensitive operational information and create substantial communication and storage demands. This study introduces AX-FDL, an adaptive and interpretable federated deep-learning architecture designed to classify network traffic while retaining the underlying records at participating sites. The proposed architecture combines dimensionality reduction, decentralized model optimization, and post-training feature attribution. An ensemble-based selection procedure reduces the original 79 CIC-IDS2017 flow attributes to 20 attributes before model training. Under the experimental configuration reported in this study, this reduction corresponds to a 74.68% decrease in model-update communication overhead. The resulting classifier is evaluated using centralized, federated IID, federated non-IID, and label-poisoning scenarios. With IID data, the framework records 99.89% accuracy, a 0.9988 macro F1-score, and a 0.9997 ROC-AUC. Under a Dirichlet non-IID configuration with $\alpha = 0.5$, accuracy remains 99.54%. When 20% of participating clients are subjected to label-flipping poisoning, the reported accuracy is 98.12%. SHAP-based attribution is additionally used to identify the flow characteristics that contribute to individual predictions and aggregate model behavior. The findings indicate that the combination of feature reduction, federated optimization, and model interpretation can provide a communication-conscious approach to distributed network threat classification.

Keywords - Federated learning; network threat classification; privacy-aware machine learning; explainable artificial intelligence; SHAP; feature reduction; CIC-IDS2017; cybersecurity.

I. INTRODUCTION

Enterprise information systems are no longer confined to a single network boundary. Cloud services, branch offices, edge computing

platforms, and connected devices continuously generate network-flow information. This expansion increases the volume and diversity of observations available to security monitoring systems, while also increasing the difficulty of collecting and processing such information centrally.

Traditional signature-oriented intrusion detection depends on previously identified patterns. Such systems remain useful for recognized threats, but their effectiveness is constrained when the observed traffic differs substantially from known signatures. Learning-based detection therefore provides an alternative in which statistical patterns can be learned from historical traffic. Deep neural networks, convolutional architectures, recurrent models, and related methods have demonstrated strong classification capabilities on network-flow datasets.

A centralized learning arrangement, however, requires traffic observations from multiple locations to be assembled at a common training location. Network-flow records may contain information about internal communication structures, service usage, host behavior, and other operational characteristics. Moving these observations between administrative or organizational boundaries can consequently introduce privacy, governance, storage, and communication concerns.

Federated learning provides a different training arrangement. Instead of moving the original training records to a common repository, participating sites perform optimization locally and communicate model updates to an aggregation coordinator. This approach changes the information exchanged during training, although it does not by itself eliminate all communication costs or all security risks.

Network intrusion detection introduces two additional difficulties. First, the CIC-IDS2017 representation contains 79 flow-level attributes. A neural model trained on the complete feature representation can therefore involve unnecessary input dimensions and corresponding parameter-update traffic. Second, traffic distributions are not necessarily identical across sites. A corporate

office, manufacturing environment, branch network, and edge installation can observe different mixtures of services and attacks. Such non-identical distributions can create client drift during federated optimization.

Interpretability is another operational requirement. High classification performance alone does not explain why an individual flow was assigned to a particular class. Security analysts may require information about the flow attributes contributing to an alert before deciding whether further investigation is warranted. SHAP provides a feature-attribution mechanism that can be applied to the trained model to obtain such information.

This work combines these three requirements—feature reduction, distributed training, and interpretability—within a single architecture. The proposed AX-FDL framework uses an ensemble-based feature-selection procedure to reduce the input space from 79 to 20 attributes, applies Federated Averaging (FedAvg) to coordinate local DNN training, and uses SHAP for post-training interpretation.

The principal contributions reported in the original study are reformulated as follows:

- A feature-reduction procedure is used to identify 20 high-importance flow attributes from the original 79-dimensional CIC-IDS2017 representation.
- A federated training architecture is defined in which raw traffic observations remain at participating nodes while model parameters are exchanged for aggregation.
- The framework is assessed under centralized, IID federated, non-IID federated, and label-poisoning conditions.
- SHAP attribution is incorporated to provide both aggregate and individual-flow explanations.
- Communication savings are quantified using the reported 79-to-20 feature configuration.

II. BACKGROUND AND RELATED STUDIES

A. Learning-Based Network Intrusion Detection

Early intrusion detection research relied on statistical and conventional machine-learning classifiers including Naïve Bayes, support vector machines, nearest-neighbor methods, and decision trees. These methods can be computationally practical, but complex network traffic may contain nonlinear relationships that are difficult to represent with shallow decision boundaries.

Deep learning expanded the range of models used for intrusion detection. Multilayer perceptrons, convolutional neural networks, long short-term memory models, and autoencoder-based systems have been applied to network-security classification. Studies using benchmark collections such as KDD Cup 99, NSL-KDD, UNSW-NB15, and CIC-IDS2017 have demonstrated the usefulness of learned representations for distinguishing normal and malicious traffic.

The CIC-IDS2017 dataset was developed to represent contemporary network traffic and attack behavior. Its flow representation provides numerous numerical attributes that can be used by supervised classifiers. The same richness, however, introduces the possibility of redundant or weakly informative inputs.

B. Federated Learning for Cybersecurity

Federated learning was introduced as a decentralized optimization paradigm in which multiple participants jointly train a model without placing their original training data in a central repository. McMahan et al. described Federated Averaging as a communication-efficient approach for aggregating locally updated neural models.

Cybersecurity applications are particularly relevant because network telemetry can be sensitive. Previous work has investigated federated learning for IoT monitoring and distributed intrusion detection, including scenarios in which traffic remains at edge locations.

A remaining issue is the amount of information associated with model updates. Reducing the dimensionality of the input representation can reduce the size of the associated model components and therefore the communication burden considered in this study. Another issue is statistical heterogeneity: local traffic distributions can differ substantially among participants.

C. Explainability for Security Analytics

Deep neural classifiers may provide strong predictive performance without directly exposing the reasoning associated with an individual output. Explainable artificial intelligence methods address this issue by estimating relationships between input variables and predictions.

SHAP is based on Shapley-value concepts from cooperative game theory. It assigns contribution values to input features for a model output. In the present architecture, these values are used to identify the attributes associated with predicted network-traffic classes.

The combination of federated optimization and post-training feature attribution is therefore relevant to distributed security monitoring. AX-FDL places feature reduction before distributed model training and uses SHAP after model inference.

III. DATASET AND DATA PREPARATION

A. CIC-IDS2017

The experiments use the CIC-IDS2017 benchmark produced by the Canadian Institute for Cybersecurity. The dataset contains flow statistics generated from packet captures using CICFlowMeter and represents both benign traffic and multiple contemporary attack categories. The flow representation contains 79 continuous numerical attributes.

The evaluation subset specified in the study consists of four captures:

- Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX.csv
- Friday-WorkingHours-Morning.pcap_ISCX.csv
- Thursday-WorkingHours-Afternoon-Infiltration.pcap_ISCX.csv
- Thursday-WorkingHours-Morning-WebAttacks.pcap_ISCX.csv

These records provide the traffic classes used in the reported experiments.

B. Data Cleaning

Records containing NaN, positive infinity, or negative infinity are removed. The study reports that 2,871 anomalous rows, representing less than 0.15% of the total samples, were excluded.

The original labels are consolidated into five operational categories:

- BENIGN

- PortScan
- Bot
- Infiltration
- Web Attack

The Web Attack category combines Brute Force, Cross-Site Scripting (XSS), and SQL Injection attacks according to the preprocessing description.

C. Numerical Scaling

Continuous variables are transformed using Min-Max normalization:

$$x' = (x - x_{\min}) / (x_{\max} - x_{\min})$$

where x is the original attribute value and $x_{\min}$ and $x_{\max}$ represent the minimum and maximum values calculated for the training partition. This transformation places the continuous inputs on a common numerical scale and prevents variables with large numerical ranges from disproportionately affecting optimization.

D. Federated Data Construction

Five participating nodes are used in the experimental simulation.

For the IID condition, the cleaned dataset is randomly shuffled and divided into five equal subsets with comparable class distributions.

For the non-IID condition, a Dirichlet distribution with concentration parameter $\alpha = 0.5$ is used to generate heterogeneous class proportions among the five participants. This configuration is intended to represent environments in which different sites observe substantially different traffic compositions.

IV. AX-FDL ARCHITECTURE

The proposed architecture contains three connected processing stages:

- Stage 1: feature reduction
- Stage 2: local neural optimization and federated aggregation
- Stage 3: model interpretation using SHAP

The overall sequence is:

Distributed participants $\rightarrow$ *local feature reduction* (79 $\rightarrow$ 20) $\rightarrow$
local DNN optimization $\rightarrow$ *model-update transmission* $\rightarrow$
aggregation coordinator $\rightarrow$ *FedAvg* $\rightarrow$ *global model distribution* $\rightarrow$
local SHAP interpretation

The architecture does not require the original traffic records to be transferred to the aggregation coordinator.

A. Feature Reduction Module

The original 79-dimensional flow representation is reduced before federated model training. The reported feature-selection mechanism combines tree-ensemble importance with Gini-based scoring.

For feature f_j , the importance is represented as:

$$\text{Importance}(f_j) = (1/T) \sum_t \sum_{n \in N_t, v(n)=f_j} \Delta I(n, t)$$

where T is the number of trees, N_t represents the internal nodes of tree t , $v(n)$ identifies the feature used at node n , and $\Delta I(n, t)$ denotes the corresponding reduction in Gini impurity.

The twenty highest-scoring attributes form the reduced feature representation. The study reports the following principal attributes and importance values:

Rank	Feature	Importance
1	Destination Port	0.1425
2	Init_Win_bytes_backward	0.1182
3	Init_Win_bytes_forward	0.1041
4	Bwd Packet Length Std	0.0894
5	Average Packet Size	0.0762
6	Packet Length Variance	0.0651
7	Bwd Packet Length Max	0.0583
8	Flow IAT Mean	0.0492
9	Total Length of Fwd Packets	0.0431
10	Flow Duration	0.0385

TABLE I. PRINCIPAL FEATURE IMPORTANCE VALUES

The remaining ten selected attributes collectively account for the reported importance value of 0.2154. The reduction from 79 to 20 attributes removes 59 inputs. Under the communication calculation reported in the study, the resulting configuration corresponds to a 74.68% reduction in update traffic.

B. Local Neural Classifier

Each participant maintains an identical DNN architecture. The input layer contains 20 units corresponding to the selected attributes and is followed by batch normalization.

The hidden structure is:

- Input: 20 units
- Hidden layer 1: 128 neurons, ReLU, dropout 0.3
- Hidden layer 2: 64 neurons, ReLU, dropout 0.2
- Hidden layer 3: 32 neurons, ReLU
- Output: 5 neurons with Softmax activation

For a local mini-batch, the categorical cross-entropy objective is:

$$L(\theta) = -(1/M) \sum_{i=1..M} \sum_{c=1..C} y_{i,c} \log(\hat{y}_{i,c})$$

where M denotes mini-batch size, $C = 5$ is the number of classes, $y_{i,c}$ is the binary target indicator, and $\hat{y}_{i,c}$ is the predicted probability for class c .

C. Federated Optimization

Training proceeds for 30 global communication rounds.

At the beginning of a round, the aggregation coordinator distributes the current global parameter vector to the participating nodes. Each node then performs five local epochs using the Adam optimizer with learning rate $\eta = 0.001$.

For participant k , let D_k denote its local dataset and $n_k = |D_k|$. After local optimization, the updated parameter vector is transmitted to the coordinator.

The new global model is calculated using weighted averaging:

$$\theta^{(t+1)} = \sum_{k=1..K} (n_k/N) \theta_k^{(t+1)}, \text{ where } N = \sum_{k=1..K} n_k$$

The weighting scheme gives each participant influence proportional to its local sample count.

D. SHAP Interpretation

Following model training and prediction, SHAP is used to quantify feature contributions. For an input x , the contribution assigned to feature i is represented by:

$$\phi_i(x) = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(|F|-|S|-1)!/|F|!}{|F|!} [f_x(S \cup \{i\}) - f_x(S)]$$

Here, F denotes the selected feature set, S is a subset excluding feature i , and $f_x(S)$ represents the model output conditioned on the subset. Because the optimized feature set contains 20 attributes, SHAP explanations operate on this reduced representation. The resulting attribution values can be inspected for individual alerts or summarized across the evaluation set.

V. EXPERIMENTAL CONFIGURATION

A. Computing Environment

- Intel Xeon E5-2680 v4 processor at 2.40 GHz
- 64 GB RAM
- NVIDIA RTX 3090 GPU with 24 GB VRAM
- Python 3.10, PyTorch 2.1, Scikit-Learn 1.3, PySyft/Flower federated-learning frameworks, and SHAP 0.42

B. Training Parameters

Parameter	Value
Number of participants	5
Global rounds	30
Local epochs per round	5
Mini-batch size	64
Learning rate	0.001
Selected attributes	20
Original attributes	79

TABLE II. PRINCIPAL TRAINING PARAMETERS

The same reduced input representation is used throughout the federated experiments.

C. Evaluation Measures

Performance is assessed using accuracy, precision, recall, macro F1-score, and ROC-AUC.

Accuracy is:

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN)$$

Precision is:

$$\text{Precision} = TP / (TP + FP)$$

Recall is:

$$\text{Recall} = TP / (TP + FN)$$

The reported F1 formulation is:

$$F1 = 2(\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})$$

TP, TN, FP, and FN represent true-positive, true-negative, false-positive, and false-negative outcomes. Communication efficiency is evaluated by comparing the update size associated with the 20-feature configuration against the corresponding unpruned 79-feature arrangement.

VI. RESULTS

A. Comparative Performance

The study evaluates four configurations: a centralized baseline, federated IID training, federated non-IID training, and federated training with poisoned clients.

Under IID conditions, AX-FDL achieves 0.9989 accuracy compared with 0.9991 for the centralized reference configuration. The reported difference is small while the federated arrangement retains training data at participating locations.

B. Effect of Statistical Heterogeneity

The non-IID experiment uses a Dirichlet concentration parameter of $\alpha = 0.5$. Accuracy decreases from 99.89% under IID conditions to 99.54% under the specified heterogeneous distribution.

This change indicates that statistical differences among local datasets affect federated optimization. Nevertheless, the reported accuracy remains above 99.5% in the evaluated configuration.

C. Label-Poisoning Experiment

The poisoning scenario introduces corrupted client updates corresponding to 20% of the participating nodes. The reported accuracy is 98.12%, with precision of 98.20%, recall of 98.12%, macro F1 of 98.10%, and ROC-AUC of 98.95%.

The experiment demonstrates the behavior of the stated FedAvg configuration when some participating training sources provide corrupted labels. The result should be interpreted as an evaluation of this particular experimental setup rather than as a general guarantee of Byzantine robustness.

D. Communication Reduction

The study reports that reducing the input representation from 79 to 20 attributes decreases total communication across 30 global rounds and five participants from 1.84 GB to 0.46 GB.

The corresponding reduction is reported as 74.68%.

This result illustrates the communication benefit associated with the selected reduced representation. The measured saving is specific to the parameterization and transmission accounting used in the reported experiment.

E. SHAP-Based Interpretation

SHAP analysis identifies the relative contribution of the selected flow attributes to the model's predictions. The principal global attributes reported in the study include Destination Port, Init_Win_bytes_backward, Bwd Packet Length Std, Average Packet Size, and Flow Duration.

For PortScan predictions, Destination Port, Init_Win_bytes_backward, and Flow IAT Mean are reported as important contributors in individual explanations.

Such attribution information can supplement the class label by indicating which flow characteristics were associated with a particular model output. This can provide analysts with additional information when reviewing alerts.

Condition	Accuracy	Precision	Recall	Macro F1	ROC-AUC	Saving
Centralized baseline	0.9991	0.9991	0.9991	0.9991	0.9998	0%
AX-FDL, IID	0.9989	0.9989	0.9989	0.9988	0.9997	74.68%
AX-FDL, non-IID, $\alpha=0.5$	0.9954	0.9958	0.9954	0.9953	0.9982	74.68%
AX-FDL, 20% poisoned	0.9812	0.9820	0.9812	0.9810	0.9895	74.68%

TABLE III. COMPARATIVE EVALUATION RESULTS

VII. DISCUSSION

The experimental results demonstrate three principal properties of the proposed configuration.

First, reducing the input representation before federated optimization substantially decreases the communication quantity reported by the experiment. This is particularly relevant to distributed environments in which repeated transmission occurs over multiple global rounds.

Second, the federated classifier maintains high classification performance under both IID and the specified non-IID configuration. The decrease from 99.89% to 99.54% indicates a measurable effect from heterogeneous data, but the selected feature representation continues to provide strong classification performance in the evaluated dataset.

Third, SHAP adds an interpretation layer to the prediction pipeline. Instead of reporting only a class label, the system can associate the prediction with feature-level contributions. This distinction is important because prediction quality and prediction interpretability address different operational requirements.

The poisoning experiment also highlights an important limitation of conventional averaging. Although the model retains 98.12% accuracy in the stated scenario, the performance reduction relative to clean IID training demonstrates that malicious or corrupted participants can affect the global model. Consequently, federated learning should not be interpreted as automatically providing immunity from adversarial model updates.

The results should also be interpreted within the scope of the experimental dataset and configuration. CIC-IDS2017 is a benchmark dataset, and the experiments use four specified captures, five simulated participants, 30 global rounds, and the stated DNN configuration. Performance in a live multi-tenant network may differ because traffic distributions, attack prevalence, feature quality, network conditions, and participant behavior can vary.

VIII. CONCLUSION

This paper presented a restructured description of AX-FDL, an adaptive and interpretable federated deep-learning framework for distributed network threat classification. The architecture combines three mechanisms: reduction of the original CIC-IDS2017 feature space, decentralized neural optimization using FedAvg, and SHAP-based post-training interpretation.

The experimental configuration reduces the feature representation from 79 attributes to 20 selected attributes. The reported communication analysis shows a 74.68% reduction in update traffic, with total communication decreasing from 1.84 GB to 0.46 GB over the stated training configuration.

For IID federated learning, the framework achieves 99.89% accuracy, 0.9988 macro F1, and 0.9997 ROC-AUC. Under the non-IID Dirichlet configuration with $\alpha = 0.5$, accuracy is 99.54%. With 20% poisoned participating clients, the reported accuracy is 98.12%.

The inclusion of SHAP provides a complementary interpretability layer by assigning feature contributions to predictions. This allows the output of the classifier to be examined in terms of the selected network-flow attributes rather than being treated solely as an unexplained class label.

Future development identified in the study includes the incorporation of homomorphic encryption and differential privacy into model aggregation and the extension of evaluation to multi-tenant 5G/6G edge environments. Additional experiments with stronger aggregation defenses, broader traffic sources, and

operational deployments would further clarify the framework's behavior beyond the benchmark configuration used here.

REFERENCES

- [1] M. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new dataset for intrusion detection systems-testbed architecture and attacks," in Proc. Int. Conf. Inf. Syst. Secur. Privacy (ICISSP), 2018, pp. 108–116.
- [2] B. McMahan, E. Moore, D. Ramage, D. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in Proc. Artif. Intell. Statist. (AISTATS), 2017, pp. 1273–1282.
- [3] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 30, 2017, pp. 4765–4774.
- [4] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," IEEE Signal Process. Mag., vol. 37, no. 3, pp. 50–60, May 2020.
- [5] V. Mothukuri, R. Parizi, S. Pouriyeh, Y. Huang, A. Dehghantanha, and G. Srivastava, "A survey on security and privacy of federated learning," Comput. Netw., vol. 192, p. 108064, Jun. 2021.
- [6] R. Zhao, Y. Wang, and Z. Zhang, "Deep learning for network intrusion detection: A systematic review," IEEE Trans. Netw. Service Manag., vol. 18, no. 4, pp. 4210–4228, Dec. 2021.
- [7] A. Rezaei and X. Liu, "Explainable artificial intelligence (XAI) in cybersecurity: Methodologies, challenges, and opportunities," Comput. Security, vol. 118, p. 102728, Jul. 2022.
- [8] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," IEEE Access, vol. 7, pp. 41525–41550, Apr. 2019.
- [9] D. Preuveneers, V. Rimmer, I. Tsingenopoulos, J. Spooren, W. Joosen, and E. Ilie-Zudor, "Chained anomaly detection models for federated learning: An evaluation: The case of cybersecurity event tracking," Appl. Sci., vol. 8, no. 12, p. 2693, Dec. 2018.
- [10] M. Marino, P. Wickramasinghe, and C. S. Wickramasinghe, "An adversarial approach for explainable AI in intrusion detection systems," in Proc. IEEE Int. Conf. Big Data, 2020, pp. 3215–3222.