I. J. Computer Network and Information Security, 2026, 2, 78-93
Published Online on April 8, 2026 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2026.02.05

Securing Fog-assisted IoT: An Adaptable and Efficient Threat Identification Approach

Surya Pavan Kumar Gudla
Department of Computer Science and Engineering, NCR- PMEC Berhampur, Faculty of Engineering, Biju Patnaik University of Technology (BPUT), Rourkela, 769015, India
E-mail: pavan1980.mca@gmail.com
ORCID iD: https://orcid.org/0000-0002-6871-5841

Sourav Kumar Bhoi*
Department of Computer Science and Engineering, Parala Maharaja Engineering College (Govt.), Berhampur, 761003, India
E-mail: sourav.cse@pmec.ac.in
ORCID iD: https://orcid.org/0000-0002-5173-3453
*Corresponding author

Kshira Sagar Sahoo
Department of Computer Science and Engineering, SRM University-AP, Amaravati, 522502, India
E-mail: kshirasagar12@gmail.com
ORCID iD: https://orcid.org/0000-0002-6435-5738

Goluguri N. V. Rajareddy
Department of Chemical and Biological Engineering, University of Saskatchewan, Saskatoon, Canada S7H 5A2
E-mail: cre138@usask.ca
ORCID iD: https://orcid.org/0000-0003-2394-2693

Received: 16 October 2025; Revised: 04 December 2025; Accepted: 18 January 2026; Published: 08 April 2026

Abstract: The increase in cyber attacks leads to significant challenges to the security in Fog based IoT environments. Existing studies have been implementing machine learning (ML), ensemble learning (EL) and deep learning (DL) for security, in this study we opted deep ensemble learning (DEL) for detection of threats in fog based IoT environments. The proposed DEL model is build using Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and Gated Re-current Units (GRU) as base models, and it is augmented using a metalearner such as Logistic Regression (LR), Random Forest (RF), AdaBoost, XGBoost, CatBoost, and LightGBM and also with a Voting Classifier (VC) for f inding the best model. In our experimentation, the evaluation is performed with different datasets such as DDoS SDN, NSL-KDD, UNSW-NB-15, and IoTID20. In this work, DEL with RF achieved better performance than other models when performance metrics such as accuracy (Acy), precision (Prn), recall (Rcl), F1-Score (F1-S) and AUC-score are considered. For instance, DEL with RF achieved an Acy of 99.99%, Prn of 100%, Rcl of 99.96%, F1-S of 99.98% and AUC-score of 1.00 on IoTID20 dataset. Afterward, to analyze the network performance of the DEL models at fog, we have considered the metrics such as cost, energy, resource utilization, latency and service time. This work shows that DELmodels can improve the security of fog assisted IoT systems.

Index Terms: Security, Fog Computing, IoT, Deep Ensemble Learning, Meta-Learner.

1. Introduction

Nowadays, technology in the world is changing as per the user demands. New technologies in computing such as Internet of Vehicle Things (IoVT), Internet of Medical Things (IoMT), smart grids and Intelligent Transportation Systems (ITS) are emerging. This rapid growth in technology and the distributed nature of IoT within a cloud environment leads to security challenges. Despite of several cloud based solutions for the users, systems are still

This work is open access and licensed under the Creative Commons CC BY 4.0 License.

susceptible to intrusions and data breaches [1–3]. Fog computing has become a reliable solution to these problems. By providing services close to devices and easing the burden on centralized cloud servers, fog computing functions well at the edge of the network. The fog layer can reduce latency and bandwidth usage, but it is not resilient to security attacks. The security and privacy of, functioning in positional binary number system (PNS). In this aspect, the main directions of improving CS in PNS are increasing user productivity and, primarily, reliability of their functioning [4-7]. Ensuring the fault-tolerance property of data exchanged in the fog environment are at risk due to vulnerabilities like port scans, DDoS attacks, and Man-in-the Middle attacks [2–5]. Although, IoT devices use non-cellular network protocols for communication, the fog layer is still vulnerable to security attacks during data transfer. Therefore, a robust security mechanism is required to incorporate it in fog computing environment [2, 3, 6, 7].

With large data transmission and high usage of internet in fog based IoT environment, it is highly recommended to protect the system from attacks. If prior to data communication the attack is detected then the system can be protected well. Currently, the technology is shifting towards predicting the network behavior using ML, EL and DL models. ML algorithms are mainly used to find patterns and abnormalities on low volumes of data. To ensure the IoT security, the EL models combine the ML models which improves the prediction accuracy with high precision. Nowadays, DL models have emerged as the best models by working on larger volumes and high-dimensional data and finding the complex patterns in them dynamically. By Considering the relevance of these models to detect the attackers by their patter's prior communication, in this work we present an effective approach that integrates the EL and DL models to better detections of attacks in the fog based IoT environment. [1, 8–12].

The DEL algorithm is build using ensemble learning algorithms and deployed in fog computing environment. This DELmodel improves the security in fog computing by integrating the strength of various deep learning models [1, 8–18]. In this work, we built a base DEL model that comprises of CNN, LSTM, GRU in first stage, and in subsequent layer metalearner or voting classifier that supports in predicting and classifying security threats with high accuracy.

The major contributions in this work are given below.

- This work presents an approach for detecting attacks in the Fog assisted IoT systems, using DEL models.
- This approach is evaluated using CNN+LSTM+GRU as base DEL model with ML and ensemble models as meta learners (LR, RF, AdaBoost, XGBoost, CatBoost, LightGBM, and VC). Four different standard datasets (DDoS SDN, NSLKDD, UNSW-NB15, and IoTID20) are used for this result analysis.
- Extensive simulations are conducted to detect the attacks, and a comprehensive comparative analysis is done based on performance metrics such as Acy, Prn, Rcl, and F1-S.
- At last, the network performance of the models at fog are analyzed using cost, energy, resource utilization, latency and service time. From the results, it is observed that DEL with RF as metalearner, outperforms all other considered DEL models.

The subsequent sections of the study follow as: Section 2 provides a detailed review of the existing literature. Section 3 presents the problem statement, and in Section 4 we discuss the system layout and attack pattern in the operating environment. Section 5 focuses on the methodology of our work, in which we detail the workflow map and algorithms used. Section 6 presents the discussion about our findings. Finally, Section 7 summarizes our work with future scope.

2. Literature Study

Many research work focuses on the security improvements in fog computing networks using deep learning models. Some are highlighted as follows. Odeh Ammar, and Anas Abu Taleb [1] proposed models that uses three particular DL models: CNN, GRU, and LSTM. These models are entirely evaluated using NSL-KDD dataset. S Ahmad et al. [5] highlights the superior performance of DL that has emerged as the preferred computational approach within ML for cloud computing, robotics, and cybersecurity. R Lazzarini et al. [8] proposed a technique called DIS-IoT. DIS-IoT that is equipped for identifying attacks in an IoT environment using MLP, DNN, CNN and LSTM models while keeping low FP and FN rate on three datasets namely ToN IoT, CICIDS2017 and SWaT datasets.

VTomer, and S Sharma [9] proposed a way to deal with the troupe AI model choice assignment for attack prediction. The proposed approach has been tried on the NSL-KDD dataset. V Dutta et al. [10] presented different component designing techniques along with dimensionality reduction to achieve the highest proficiency. A blend of DNN and LSTM followed by a meta-classifier is used for attack detection. F Alanazi et al. [11] utilized DL-based ensemble models, combining architectures like CNN, LSTM, and GRU. These EL models over CICIDS2017 dataset have significantly improved the DDoS detection, with Acy of 99.77%. A Mohammed, and R Kora [12] discussed the strengths of EL models in deep learning.

TATangetal. [19] presented DL model for network interruption identification, and also evaluated intrusion detection in SDN. The proposed methodology shows that the outcome of the work is highly suitable for flow based anomaly detection events. F Khan et al. [20] in their work focused on the Internet of Medical Things (IoMT) which enhances healthcare efficiency and quality, this approach combines LSTM networks as individual learners, stacked with a decision tree to classify attacks. It is evaluated on the ToN-IoT dataset, by showing 4% improvement in precision over

baseline approaches. M P Ramkumar et al. [21] in their study introduces a hybrid optimization based ensemble classifier in fog computing platform. The fog layer utilizes the trained ensemble classifier for intrusion detection, achieving high precision (88%), recall (88.4%), F measure (88.2%), and accuracy (97.2%).

P Kumar et al. [22] proposed an ensemble intrusion detection model combining DL with efficient feature selection using RandomForests. It is evaluated on the UNSW-NB15 dataset and the model shows a high Acy of 97.5%, outperform ing several traditional models. Illy Poulmanogo, et al. [23] discussed increasing security threats in IoT applications and highlighted the need for effective Intrusion Detection Systems (IDS). It discusses about the EL which combines multiple algorithms for detection of intrusions.

From the above study, it is seen that despite advancement in the application of ML, EL, and DL models in various domains, significant research gaps remain, particularly in the context of threat detection within fog computing environ ments. Existing studies mostly carried out their work on single dataset. The existing studies on DEL models, motivated us to implement CNN+LSTM+GRU with different ensemble techniques like LR (legacy ML model), RF, AdaBoost, XGBoost, CatBoost, and LightGBM on the DDoS SDN, NSLKDD, UNSW-NB15, and IoTID20 datasets. Also, in this work a framework is presented that helps to detect the attacks using DEL prior to communication in cloud-fog based IoT environments.

3. Problem Statement

In the growing field of IoT and data analytics, the increasing number of security risks within fog-based IoT systems is a big challenge for data communication in the network. In this environment, we assume each communication between the IoT devices as a transaction that produces n records, comprising q malicious out of n records where (n-q) are normal. The main objective is to precisely identify the anomalies through binary classification using the DEL models. So, that prior to communication an IoT device can identify an attacker. The objective is to build the DEL models such as CNN+LSTM+GRU with LR, CNN+LSTM+GRU with RF, CNN+LSTM+GRU with AdaBoost, CNN+LSTM+GRU with XGBoost, CNN+LSTM+GRU with CatBoost, CNN+LSTM+GRU with LightGBM, and CNN+LSTM+GRU with VC and select the best model to be implemented at the fog devices for identification of attacker in the network.

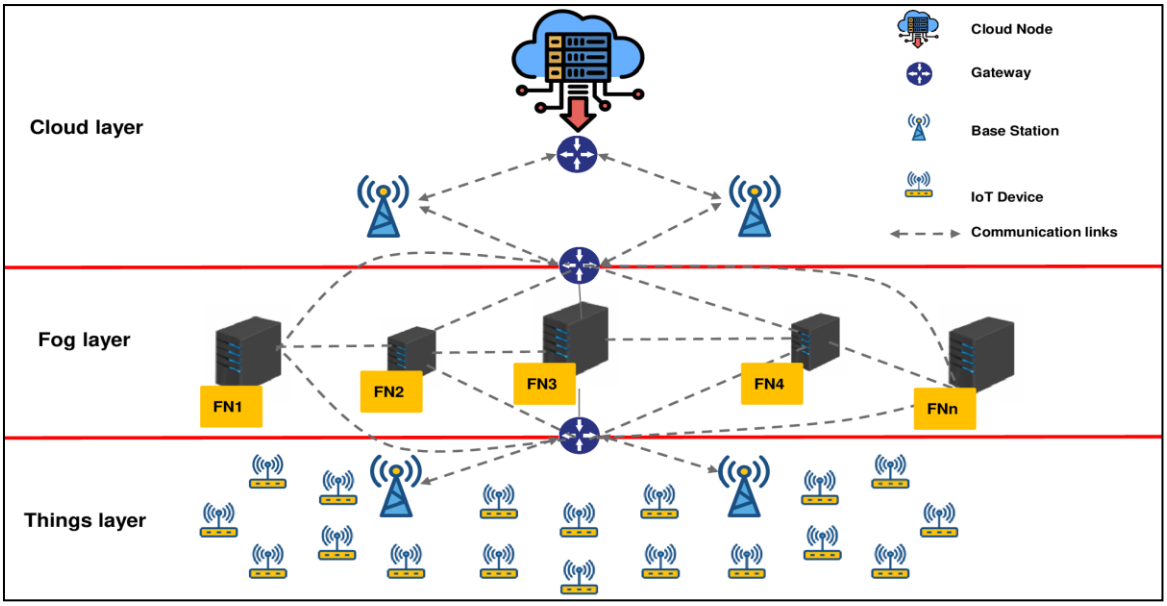


Fig.1. System layout

4. System Layout and Attack Pattern

The system architecture consists of three layers with cloud (CN), fog node (FN) and IoT devices. The IoT devices communicates with fog nodes and cloud using gateways (GW) and base stations (BS). We assume the IoT devices when communicates with each other use wireless communications (4G/5G/Wi-Max), and generates a communication transac tion. This transaction is then sent to the nearest fog node using wireless communications via BS and gateway to detect the attack at the fog level. The BS, gateway, fog and cloud also use wireless commmunications for data transmission. The network architecture is as depicted in Fig. 1 [2,3]. Here, we assume that attacks at the Fog layer can occur due to various factors, such as devices, protocols, etc. However, the primary vulnerability lies in the compromised IoT device, that is represented in Fig. 2.

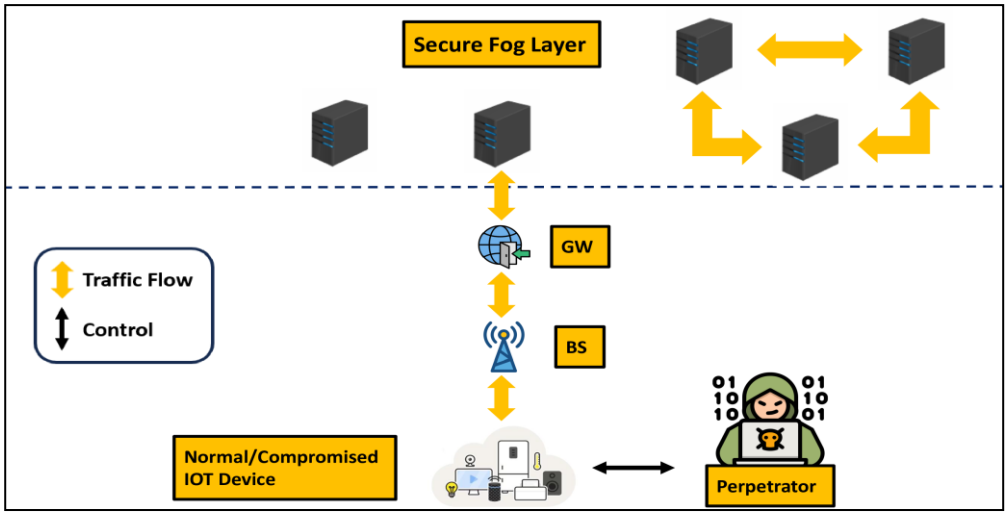


Fig.2. Attack pattern

5. Methodology

The methodology workflow consists of six stages as depicted in Fig. 3 and the step by step working of the proposed methodology is shown in Fig. 4.

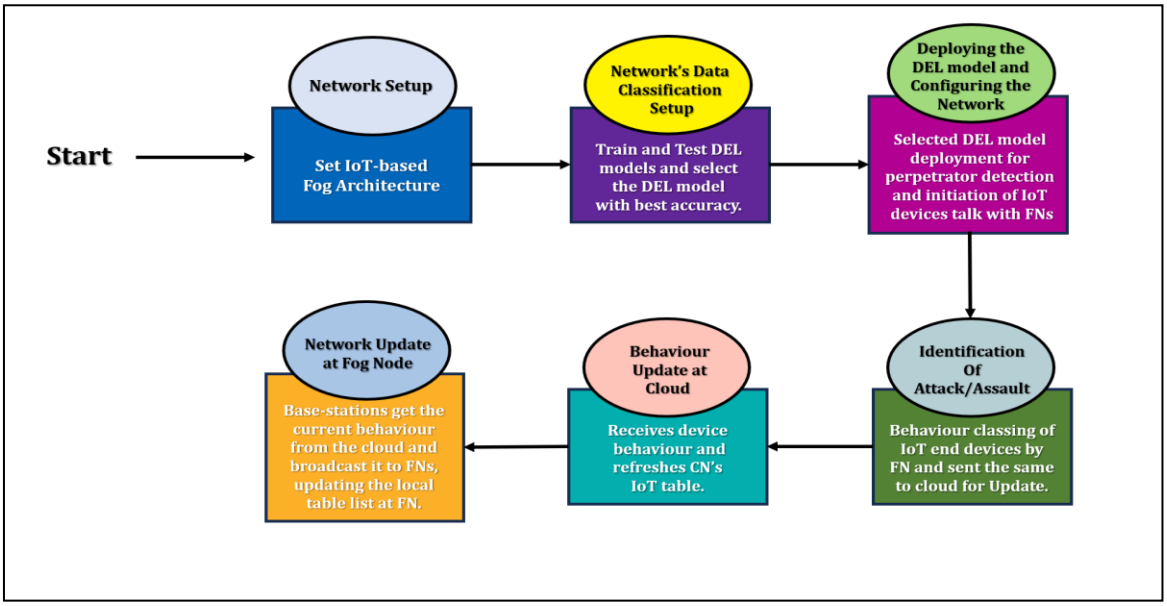


Fig.3. Workflow map

The stepwise approach of the methodology is shown in Fig. 4. In step-1 the IoT devices communicates and send the network flow data or communication data to the fog node through the base station and gateway for attack identification. This process is continuously done for storage of network flow at the fog node (step-2). In the second stage, FNs are equipped with various DEL algorithms to identify the behavior of IoT devices, that requires data preprocessing. Data preprocessing activities include removing null values, feature scaling techniques like standardization and one-hot encoding.

The evaluation procedure or to select the best DEL model is represented in Algorithm 2. At the fog, the models such as DEL with LR, RF, AdaBoost, XGBoost, CatBoost, LightBGM, and VC are trained and tested to determine the best DELmodel. The fog node is then deployed with the best DEL model and it is selected based on training and testing with standard datasets (step-3 and step-4). Algorithm 1 shows the steps to select the best model. As the fog node is trained with best DEL model, it can classify the attacker network flow from genuine IoT device network flow (step-5). Then the attacker information is updated at the cloud node and fog nodes (step-6 and step-7). As per the network architecture the fog nodes are connected to each other. So, each nodes shares its attack list to other for knowing the malicious IoT devices. Then, the fog nodes send the attack list to the IoT devices in its communication range for information, so that in future the IoT device prior to communication first identify a IoT device by looking its local

database of attack list. This is how the methodology works for the attack detection scheme.

In this study, the process for DEL model construction is shown in Fig. 5, as discussed in introduction 1, we have considered nine models. To build the base model in DEL, we utilized CNN, LSTM, and GRU, while LR, RF, Ad aBoost, XGBoost, Cat-Boost, and LightGBM were taken as metalearner one at a time in further layer as discussed in major contribution part of introduction which accepts the predictions of base model as input. For the VC, all metalearners were used. The models can be studied from [2, 3, 24–29].

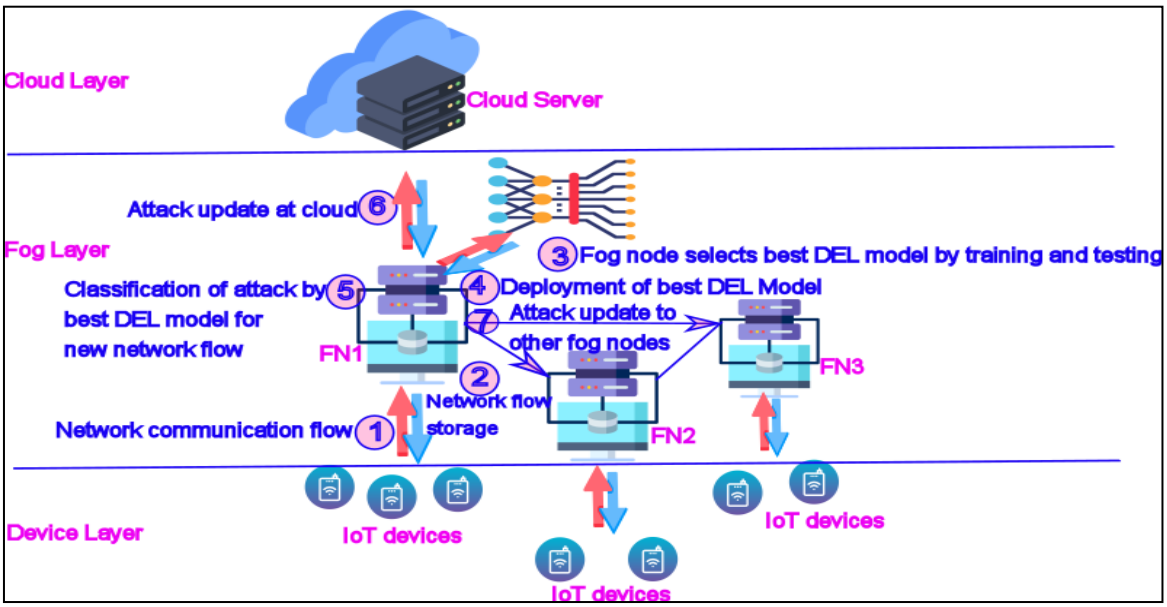


Fig.4. Stepwise approach of proposed methodology

The DLmodels used for construction of DEL model are discussed as follows [2,3]. CNN, LSTM, and GRU are three useful models for handling data with different patterns. CNN works by using small filters on the input to detect important features. The operation is shown as follows:

$$Q_{a,b} = \sum_{\mu=1}^{p}\sum_{\nu=1}^{q}\left(X_{a+\mu,b+\nu} \lozenge W_{\mu,\nu}\right) + \phi \tag{1}$$

where X is the input, W is the filter, ϕ is a bias term, and ⋄ is an element wise interaction.

Algorithm 1 IoT-Fog Communication and Attack Detection using DEL model (CNN+LSTM+GRU) with Meta learners (LR, RF, AdaBoost, XGBoost, CatBoost, LightGBM, and VC)

Input: IoT communication data (D_{IoT}), DEL model candidates (M_{DEL})

Output: Best DEL model (M_{best}) deployed at Fog node, attack updates

Begin
Step 1: Device Communication
$D_{IoT} \rightarrow Fog\ Node$
Generate network flow data F_{net}
Step 2: Data Storage
$F_{net} \rightarrow Fog\ DB$
Step 3: Model Training and Testing
Extract features $X \longleftarrow F_{net}$
for{ $M \in M_{DEL}$ } **do**
Train model M on X
Test model M
Compute evaluation metrics (*Acy*, *Prn*, *Rcl*, *F1-S*)
Step 4: Model Selection
$M_{best} \leftarrow \arg\max_{M \in M_{DEL}} \{Performance(M)\}$
Step 5: Model Deployment

11. Deploy $M_{best} \rightarrow Fog\ Node$

Step 6: Attack Detection

12. Detect attacks in real-time using M_{best}

Step 7: Cloud Reporting

13. Detected attacks $\rightarrow Cloud$.

Step 8: Synchronization

14. Cloud updates attack information
15. Cloud $\rightarrow$ All Fog nodes

Step 9: Fog Node Updates

16 Fog nodes enhance local detection with updated attack info
17. **END**

Algorithm 2 DEL Model with Meta Learners for Attack Classification

Input: Dataset D with features X and labels Y
Output: Performance metrics (Accuracy, Precision, Recall, F1-Score)

1. **Begin**

Step 1: Data Preprocessing

2. Split dataset D into training (D_{train}) and testing (D_{test}) sets
3. Normalize and standardize features in D_{train} and D_{test}

Step 2: Train Base Learners (CNN, LSTM, GRU)}

4. Train Convolutional Neural Network (CNN) model M_{CNN} on D_{train}
5. Train Long Short-Term Memory (LSTM) model M_{LSTM} on D_{train}
6. Train Gated Recurrent Unit (GRU) model M_{GRU} on D_{train}

Step 3: Generate Predictions

7. Obtain predictions from M_{CNN} , M_{LSTM} , and M_{GRU} on D_{train}
8. Concatenate predictions as feature matrix F_{train}
9. Obtain predictions on D_{test} to form F_{test}

Step 4: Train Meta Learners

10. Train Logistic Regression (LR) on F_{train}
11. Train Random Forest (RF) on F_{train}
12. Train AdaBoost on F_{train}
13. Train XGBoost on F_{train}
14. Train CatBoost on F_{train}
15. Train LightGBM on F_{train}
16. Train Voting Classifier (VC) with all meta-learners on F_{train}

Step 5: Evaluate Performance

17 Compute *Acy, Prn, Rcl, and F1-S* on F_{test} for each meta-learner
18. Output the best-performing meta-learner
19. **END**

An LSTM is a special kind of recurrent unit that uses memory and gates to decide what to keep or reject from the past. Its working is described as follows:

$$R_t = \psi(B_r \overset{o}{a}[CSt_{t-1}, Z_t] + \gamma_r) \tag{2}$$

$$A_t = \psi(B_a \text{ å } [CSt_{t-1}, Z_t] + \gamma_a) \tag{3}$$

$$M_t = (R_t \bullet M_{t-1}) + \left(A_t \bullet \tanh(B_m[Cst_{t-1}, Z_t] + \gamma_m)\right) \tag{4}$$

$$CSt_t \quad CSt_t = \psi(B_o \overset{o}{a}[CSt_{t-1}, Z_t] + \gamma_o) \bullet \tanh(M_t) \tag{5}$$

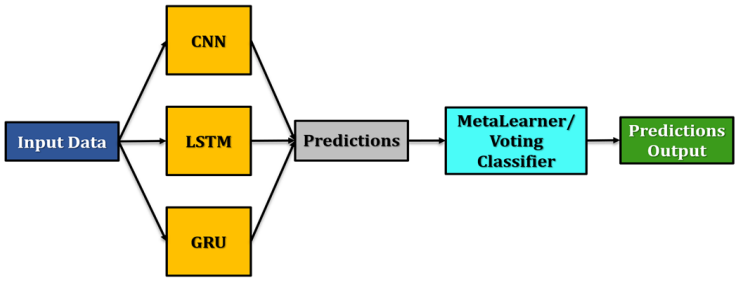


Fig.5. DEL model architecture

where R_t, A_t are gate activations, M_t is memory, CSt_t is the output, ψ is a nonlinear function, and • is a selective product.

A GRU is a simple model that combines the gates for efficiency. Its working is described as follows:

$$U_t = \psi(C_u \overset{o}{a}[CSt_{t-1}, Z_t]) \tag{6}$$

$$S_t = \psi(C_s \overset{o}{a}[CSt_{t-1}, Z_t]) \tag{7}$$

$$CSt_t = \tanh(C_h \overset{o}{a}[S_t \bullet CSt_{t-1}, Z_t]) \tag{8}$$

$$CSt_t = (1-U_t) \bullet CSt_{t-1} + U_t \bullet CSt_t \tag{9}$$

where U_t is the update factor, S_t is the selection factor, and CSt_t is the candidate state.

6. Results and Discussion

The performance of the models is analyzed using Python tool that is installed in a system that met the following specifications: AMD Ryzen 7 3750H processor, 16.0 GB of RAM, 1.5 TB of storage, and Windows 10 (64-bit) operating system. In this study, we evaluated the performance of various DEL models using different algorithms across four datasets: DDoS SDN, NSL-KDD, UNSW-NB-15, and IoTID20 as presented in Table 3. For each dataset total records, benign traffic, attacker traffic, total features and customized features are shown in Table 1. The base model for each DEL model comprised CNN, LSTM, and GRU, and in the subsequent stage with LR as the initial metalearner and remaining continues as discussed in previous sections. The CNN was built with a single convolution layer comprising 32 filters with kernel size 3 and ReLu activation function, followed by a fully connected dense layer with 64 neurons with dropout rate 0.5. The LSTM and GRUmodels consists of a single recurrent layer with 64 units with a Sigmoid output for binary classification. On all four datasets, the base model is initially trained for 10 epochs with batch size of 32 using Adam optimizer on learning rate 0.001. The predictions from the base model were fed to the metalearners for final classification. Each metalearner is trained with default configurations like 100 estimators for tree-based models, iterations=100 for CatBoost etc.

Table 1. Dataset statistics

Dataset	Total Samples	Normal Samples	Attack Samples	Features	Classes	Reference
DDoS SDN	104345	63561	40784	23	17	30
NSLKDD	148517	77054	71463	43	19	31
UNSW-NB15	257673	93000	164673	45	44	32
IoTID20	625783	40073	585710	86	68	33

6.1. Performance Analysis of DEL Models

To present a holistic view, we also show the performance metrics of the DL models in Table 2. These results serve as a basis for performance comparison with the DEL models performance during threat detection in fog-assisted environments. However, this study is for advancing DEL models which not only performed well but also surpass traditional DL models in all aspects.

For the DDoS SDN dataset, the DEL model with LR achieved an Acy of 98.77%, Prn of 98.23%, Rcl of 98.68%, and F1-S of 98.46%. Subsequently, the DEL model with RF achieved perfect performance metrics, all at 100%. Other

algorithms like AdaBoost, XGBoost, CatBoost, LightGBM, and the VC also demonstrated high performance across various metrics.

For the NSL KDD dataset, the DEL model with LR achieved an Acy of 98.26%, Prn of 98.11%, Rcl of 98.52%, and F1-S of 98.31%. Notably, the DEL model with RF achieved the best scores across all metrics, at 99.97%, 99.8%, 99.61% and 99.71% respectively. Similarly, the other algorithms also exhibited strong performance.

For the UNSW-NB-15 dataset, the DEL model with LR achieved an Acy of 97.45%, Prn of 98.16%, Rcl of 97.87%, and F1-S of 98.01%. Again, the DEL model with RF performed with near-perfect scores across all metrics at 99.99%. Other algorithms showcased consistent and high performance as well.

Lastly, for the IoTID20 dataset, the DEL model with LR achieved an Acy of 99.84%, Prn of 99.55%, Rcl of 96.6%, and F1-S of 98.05%. Similarly, the DEL model with RF demonstrated excellent performance with near-perfect scores across all metrics at 99.99%. Other algorithms also exhibited strong performance. This indicates the robustness of the DELapproach across different datasets.

In summary, the results shows that the DEL models use a combination of CNN, LSTM, and GRU as base models, along with various metalearners, consistently achieved high Acy, Prn, Rcl, and F1-S across the four considered datasets. This shows the effectiveness of ensemble learning techniques in enhancing the performance of attack detection systems in fog environments. However, among the DEL models, those combined with RF and XGBoost emerged as best performers. RF delivers best performance in every dataset.

Along with the performance metrics mentioned in Table 3, ROC curve with AUC score is obtained to address the imbalance nature of datasets. It is observed that DEL with RF outperformed all other in this scenario. As IoTID20 is novel dataset, in this study we included ROC-AUC scores as shown in Fig. 6. DEL model with other metalearners and VCshows less efficient than DEL with RF where DEL with XGBoost also consistently performed at best and at times it matched with DEL with RF in performance wise.

Table 2. DL models performance metrics

Dataset Name	ML/DL Model	Accuracy (%) Acy	Precision (%) Prn	Recall (%) Rcl	F1-Score (%) F1-S
DDOS SDN	MLP	99.46	99.25	99.4	99.32
	LSTM	**99.7**	**99.6**	**99.64**	**99.62**
	Bi-LSTM	99.63	99.78	99.29	99.53
	GRU	99.66	99.55	99.6	99.57
	HEM	97.87	99.17	97.28	98.22
	CNN+LSTM	96.43	95.71	95.27	95.49
NSLKDD	MLP	99.05	99.04	99.12	99.08
	LSTM	**99.12**	**99.22**	**99.08**	**99.15**
	Bi-LSTM	98.95	98.85	99.13	98.99
	GRU	98.43	98.41	98.56	98.48
	HEM	98.28	99.62	96.8	98.19
	CNN+LSTM	98.19	97.79	98.72	98.25
UNSW-NB15	MLP	93.41	95.74	93.51	94.61
	LSTM	**94.11**	**95.87**	**94.47**	**95.16**
	Bi-LSTM	94.04	95.86	94.43	95.14
	GRU	93.57	95.75	93.76	94.74
	HEM	94.05	92.08	92.37	92.22
	CNN+LSTM	92.85	94.37	94.05	94.21
IoTID20	MLP	99.84	99.7	97.84	98.76
	LSTM	**99.88**	**99.77**	**98.4**	**99.08**
	Bi-LSTM	99.86	99.57	98.27	98.92
	GRU	99.84	99.61	98.01	98.8
	HEM	99.84	99.83	99.99	99.92
	CNN+LSTM	99.76	99.7	96.58	98.11

6.2. Statistical Analysis

To evaluate whether the observed performance differences among models were statistically correct, we considered the accuracy values of six models such as MLP, LSTM, Bi-LSTM, GRU, CNN+LSTM, and CNN+LSTM+GRU (DEL) with Random Forest with four benchmark datasets such as DDoS-SDN, NSL-KDD, UNSW-NB15, and IoTID20. Since, same models were tested with multiple datasets, the Friedman test was applied. The test produced a Friedman $\chi 2$ statistic of 18.31 with a p-value of 0.0026 that clearly indicates that the performance differences among the models are

statistically significant at 1% level.

Table 3. DEL models performance metrics

Dataset Name	DEL model	Accuracy(%) Acy	Precision(%) Prn	Recall(%) Rcl	F1-Score(%) F1-S
DDoS SDN	DEL with LR	98.77	98.23	98.68	98.46
	DEL with RF	**100**	**100**	**100**	**100**
	DEL with AdaBoost	98.98	98.27	99.19	98.73
	DEL with XGBoost	99.97	99.93	100	99.1
	DEL with CatBoost	99.28	99.88	99.32	98.1
	DEL with LightGBM	99.56	99.32	99.56	99.45
	DEL with VC	99.14	98.81	99.04	98.93
NSL KDD	DEL with LR	98.26	98.11	98.52	98.31
	DEL with RF	**99.97**	**99.8**	**99.61**	**99.71**
	DEL with AdaBoost	98.034	97.75	98.46	98.1
	DEL with XGBoost	99.16	98.94	99.44	98.6
	DEL with CatBoost	98.55	98.4	98.8	98.6
	DEL with LightGBM	98.6	98.43	98.87	98.65
	DEL with VC	98.44	98.21	98.79	98.5
UNSW-NB-15	DEL with LR	97.45	98.16	97.87	98.01
	DEL with RF	**99.99**	**100**	**99.99**	**99.99**
	DEL with AdaBoost	97.11	98.07	97.42	97.75
	DEL with XGBoost	98.47	99.06	98.55	98.4
	DEL with CatBoost	97.95	98.76	98.04	98.4
	DEL with LightGBM	97.88	98.75	97.94	98.35
	DEL with VC	97.5	98.36	97.73	98.05
IoTID20	DEL with LR	99.84	99.55	96.6	98.05
	DEL with RF	**99.99**	**100**	**99.96**	**99.98**
	DEL with AdaBoost	99.78	99.45	97.13	98.28
	DEL with XGBoost	99.93	100	98.96	98.79
	DEL with CatBoost	99.84	99.6	97.99	98.79
	DEL with LightGBM	99.86	99.59	98.32	98.95
	DEL with VC	99.86	99.81	98.08	98.93

To analyze better, three visualizations are generated in Fig. 7, Fig. 8 and Fig. 9. The boxplot of accuracies highlighted that CNN+LSTM+GRU (DEL) with RF consistently achieves near perfect results with all datasets. Whereas, CNN+LSTM showed both lower accuracy and wider variability. The line plot across datasets further shows the dominance of DEL+RF, which remained the best in every case. While CNN+LSTM shows accuracy drop when UNSW-NB15 is considered. The heatmap provided a clear dataset wise overview, showing that DEL+RF achieved the maximum ac curacy in all datasets (100.00, 99.97, 99.99, and 99.99). Whereas CNN+LSTM consistently shows lowest values. These plots show DEL+RF shows higher accuracy and most stable model across all datasets.

To validate these results, a post hoc Nemenyi test is performed. The average ranks obtained across datasets were 1.00 for DEL+RF, 2.00 for LSTM, 3.50 for Bi-LSTM, 4.13 for GRU, 4.38 for MLP, and 6.00 for CNN+LSTM. With a critical difference value of approximately 1.73 at a 0.05 significance level. From the results, it is found that DEL+RF is statistically performing better. These results with the Friedman test strongly support that DEL+RF provides the most robust and reliable performance in all datasets.

6.3. Performance Analysis using Network Parameters

From above results it is seen that, DEL with RF performs better with respect to the parameters considered for all the datasets. However, when the results of DDoS SDN dataset are compared with that of other datasets, then DEL with RF performs better as shown in Table 3. So, we considered this scenario for evaluating the fog device performance where, number of IoT devices are varied from 100-500. It is assumed that these many numbers of network traffic flow happen between IoT devices and fog device. Each row in the dataset represents a network traffic flow. The performance is analyzed by increasing the number of IoT devices by studying parameters such as energy consumption (in Joules), cost estimation (USD), resource utilization, latency (seconds), and service time (seconds). It is also considered that the fog device operates at 200 Watt. The cost per watt second taken is \$0.00005.

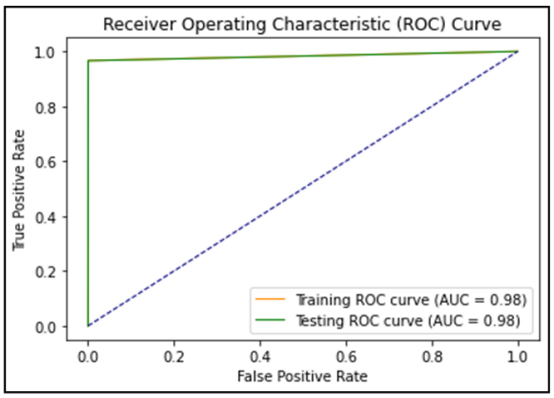


Fig.6(a). ROC-AUC for DEL with LR

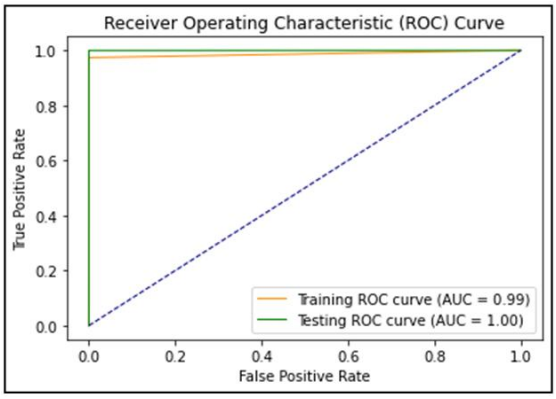


Fig.6(b). ROC-AUC for DEL with RF

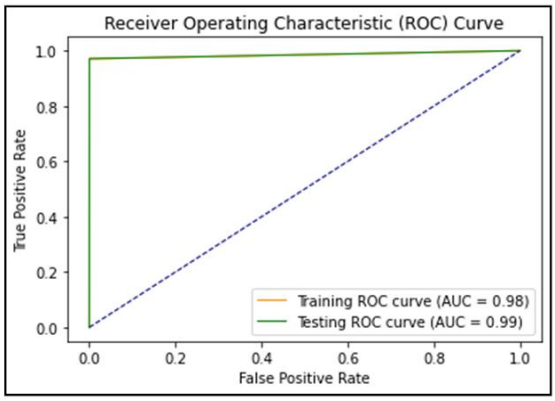


Fig.6(c). ROC-AUC for DEL with AdaBoost

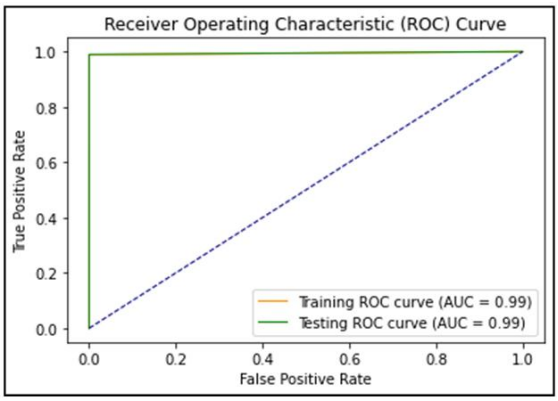


Fig.6(d). ROC-AUC for DEL with XGBoost

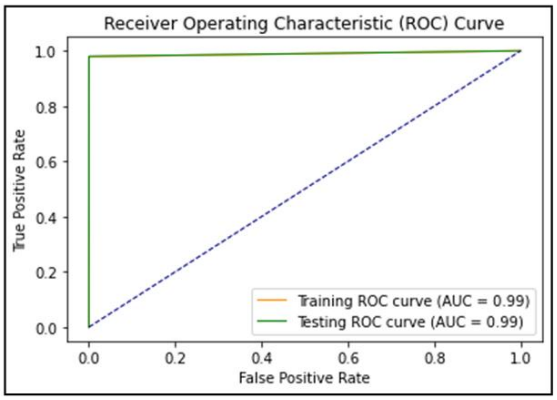


Fig.6(e). ROC-AUC for DEL with CatBoost

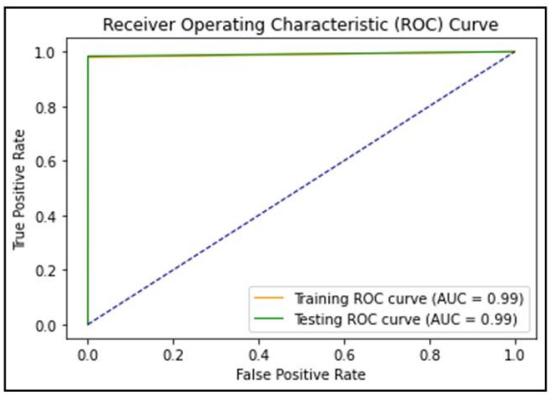


Fig.6(f). ROC-AUC for DEL with LightGBM

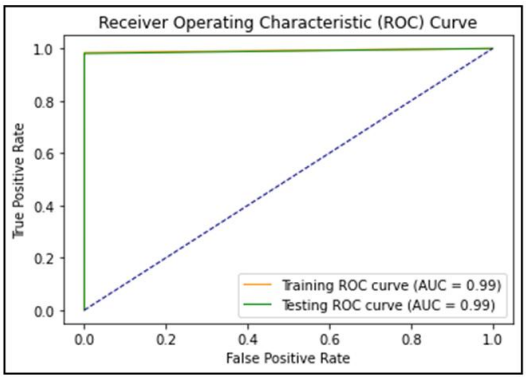


Fig.6(g). ROC-AUC for DEL with VC

Fig.6. Results of ROC-AUC for DEL models

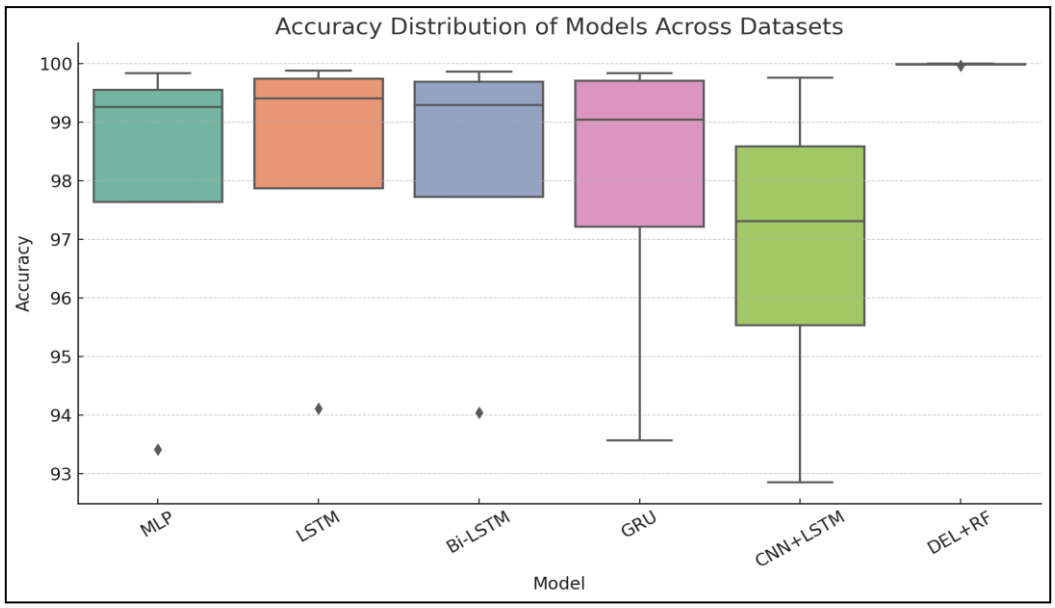


Fig.7. Accuracy distribution of models across datasets

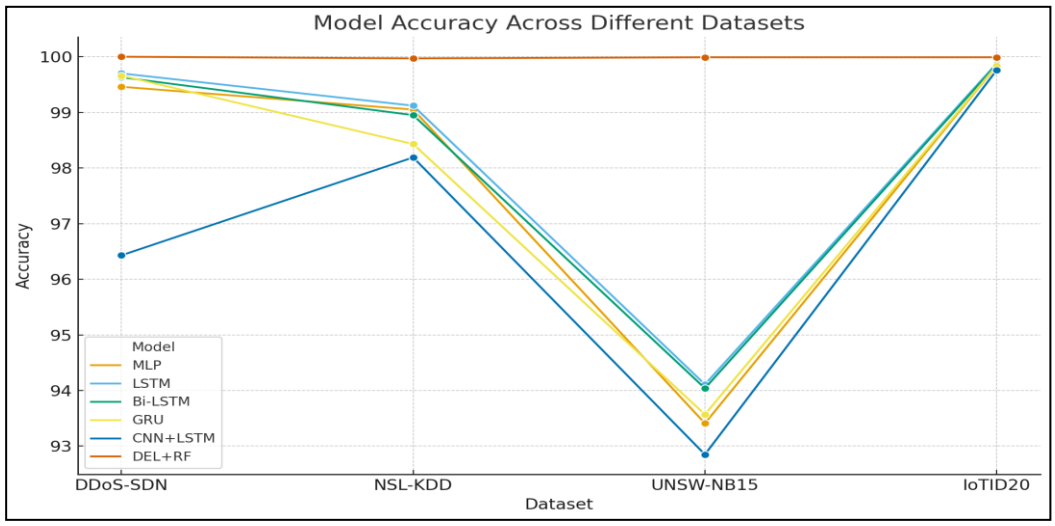


Fig.8. Model accuracy across different datasets

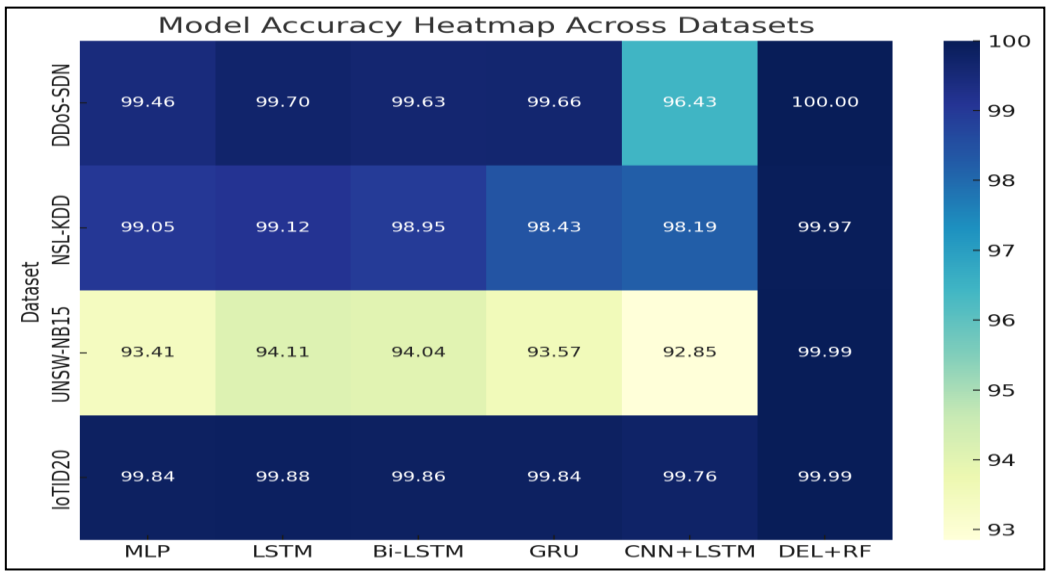


Fig.9. Model accuracy heatmap across datasets

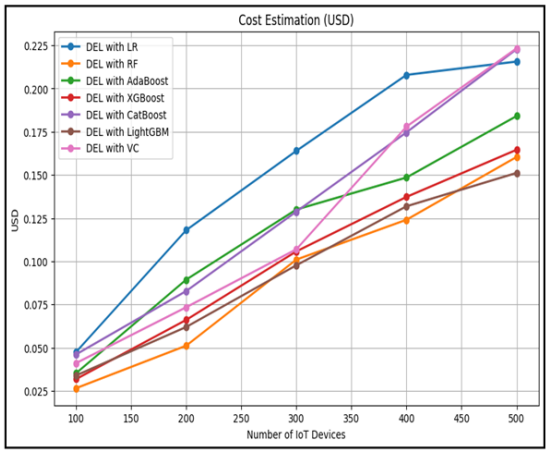


Fig.10(a). Cost Comparison

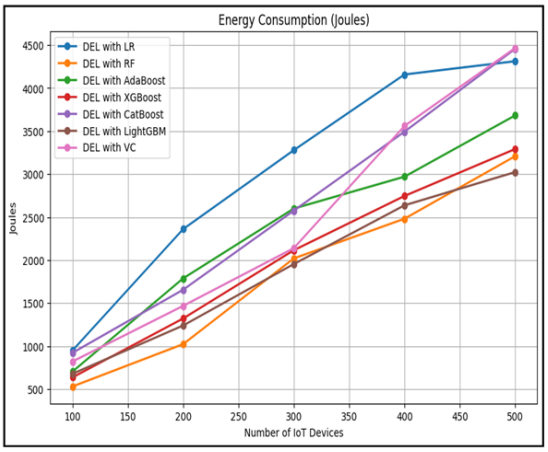


Fig.10(b). Energy Comparison

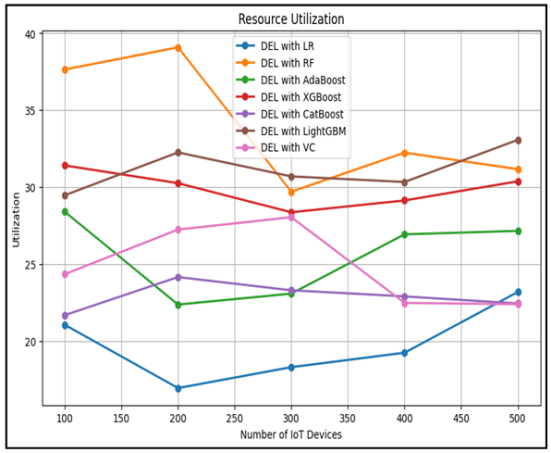


Fig.10(c). Resource Utilization

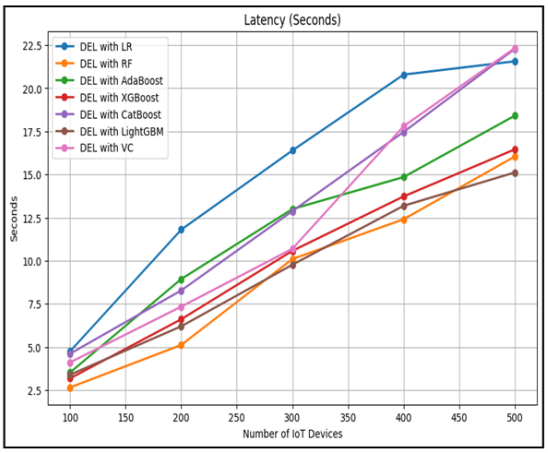


Fig.10(d). Latency Comparison

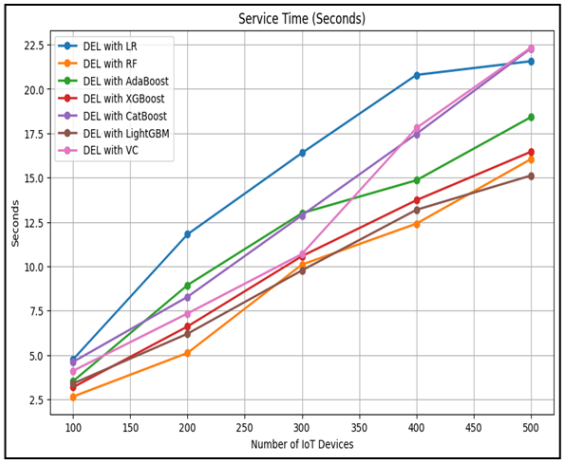


Fig.10(e). Service Time Comparison

Fig.10. Performance analysis of DEL models using network parameters

From Fig. 10a, it is observed that DEL with RF performs better than other DEL models by showing minimum cost. When the number of IoT devices increases the cost also increases. When the number of IoT devices varies from 100 to 500 (100,200,300,400, and 500) then DEL with RF shows cost (in USD) of 0.026, 0.051, 0.100, 0.124, 0.160. respectively. Similarly, DEL with LR shows 0.047, 0.118, 0.163, 0.207, 0.215 respectively. DEL with AdaBoost shows 0.035, 0.089, 0.129, 0.148, 0.184 respectively. DEL with XGBoost shows 0.031, 0.066, 0.105, 0.137, 0.164 respectively. DEL with CatBoost shows 0.046, 0.082, 0.128, 0.174, 0.222 respectively. DEL with LightGBM shows 0.033, 0.061, 0.097, 0.131, 0.151 respectively. DEL with VC shows 0.041, 0.073, 0.106, 0.177, 0.223 respectively.

From Fig. 10b, it is observed that DEL with RF performs better than other DEL models by showing minimum energy consumption. When the number of IoT devices increases the energy consumption also increases. When the number of IoT devices varies from 100 to 500 (100,200,300,400, and 500) then the energy consumption (in joules) is found as follows. DEL with LR shows 949.816, 2360.571, 3278.392, 4157.853, 4312.037 respectively. DEL with RF shows 531.198, 1023.233, 2019.705, 2480.892, 3208.072 respectively. DEL with AdaBoost shows 04.116, 1787.963,

2599.465, 2969.871, 3681.824 respectively. DEL with XGBoost shows 636.680, 1321.696, 2114.853, 2745.556, 3291.229 respectively. DEL with CatBoost shows 922.370, 1655.797, 2575.286, 3493.089, 4456.069 respectively. DEL with LightGBM shows 678.517, 1239.934, 1954.270, 2636.965, 3023.225 respectively. DEL with VC shows 821.508, 1468.209, 2139.030, 3559.108, 4465.632 respectively.

From Fig. 10c, it is observed that DEL with RF performs better than other DEL models by showing high resource utilization. When the number of IoT devices increases the utilization also increases. When the number of IoT devices varies from 100 to 500 (100,200,300,400, and 500) then the resource utilization are found as follows. DEL with LR shows 21.056, 16.945, 18.301, 19.240, 23.19089411 respectively. DEL with RF shows 37.650, 39.091, 29.707, 32.246, 31.171 respectively. DEL with AdaBoost shows 28.404, 22.371, 23.081, 26.937, 27.160 respectively. DEL with XGBoost shows 31.412, 30.264, 28.370, 29.137, 30.383 respectively. DEL with CatBoost shows 21.683, 24.157, 23.298, 22.902, 22.441 respectively. DEL with LightGBM shows 29.476, 32.259, 30.701, 30.337, 33.077 respectively. DEL with VC shows 24.345, 27.244, 28.050, 22.477, 22.393 respectively.

From Fig. 10d, it is observed that DEL with RF performs better than other DEL models by showing minimum latency. When the number of IoT devices increases the latency also increases. When the number of IoT devices varies from 100 to 500 (100,200,300,400, and 500) then the latency (in seconds) are found as follows. DEL with LR shows 4.749, 11.802, 16.391, 20.789, 21.560 respectively. DEL with RF shows 2.655, 5.116, 10.098, 12.404, 16.040 respectively. DEL with AdaBoost shows 3.520, 8.939, 12.997, 14.849, 18.409 respectively. DEL with XGBoost shows 3.183, 6.608, 10.574, 13.727, 16.456 respectively. DEL with CatBoost shows 4.611, 8.278, 12.876, 17.465, 22.280 respectively. DEL with LightGBM shows 3.392, 6.199, 9.771, 13.184, 15.116 respectively. DEL with VC shows 4.107, 7.341, 10.695, 17.795, 22.328 respectively.

From Fig. 10e, it is observed that DEL with RF performs better than other DEL models by showing minimum service time. When the number of IoT devices increases the service time also increases. When the number of IoT devices varies from 100 to 500 (100,200,300,400, and 500) then the service time (in seconds) are found as follows. DEL with LR shows 4.749, 11.802, 16.391, 20.789, 21.560 respectively. DEL with RF shows 2.655, 5.116, 10.098, 12.404, 16.040 respectively. DEL with AdaBoost shows 3.520, 8.939, 12.997, 14.849, 18.409. DEL with XGBoost: 3.183, 6.608, 10.574, 13.727, 16.456 respectively. DEL with CatBoost shows 4.611, 8.278, 12.876, 17.465, 22.280 respectively. DEL with LightGBM shows 3.392, 6.199, 9.771, 13.184, 15.116 respectively. DEL with VC shows 4.107, 7.341, 10.695, 17.795, 22.32816017 respectively. The latency and service times are identical as we assume the fog server processes the traffic immediately without adding any transmission or queuing delays.

From above analysis, it is observed that the DEL with RF model performs better than other DEL models in terms of cost, energy consumption, resource utilization, latency and service time. The scalability of the DEL models got evaluated by considering 100 to 500 IoT devices for the above mentioned network parameters are depicted in Fig. 10. While increased resource loads are expected under increased user-demand, the resource usage increase within the DEL models is manageable and can be effectively scaled to larger IoT deployments. This directly addresses the edge versus cloud tradeoff. Light tasks are done at the fog/edge nodes, while the heavy resource-demanding computations are done in the cloud, thus the system is cross-layer efficient and scalable. Besides evaluating the results on the four considered datasets, the proposed DEL framework is capable of being applied for practical fog environments. In smart healthcare, it can protect patient information from cyber intrusions by performing real-time detection at the fog nodes in a hospital. In Industrial IoT, it adds protection against attacks that can disrupt production systems, while in autonomous vehicles, it facilitates the on-board detection of anomalies with low latency to protect the passengers. These use-cases underscore the importance of our framework in securing multiple IoT environments.

7. Conclusions

In this study, the implementation of DEL techniques, leveraging a combination of CNN, LSTM, and GRU as base models, along with various metalearners, has demonstrated remarkable success in enhancing the performance of threat identification across diverse datasets. Through extensive evaluation, it is evident that the DEL models have consistently achieved the best generalization, high accuracy, precision, recall, and F1-Score metrics, showcasing their robustness and effectiveness in identifying threats. Also, at fog DEL with RF metalearner performs better than other DEL models in terms of cost, energy, resource utilization, latency, and service time. This study underscores the significance of DEL methodologies in bolstering cybersecurity measures, particularly in the context of evolving network threats in fog computing environments. Although the DEL models achieved the best results, they have limitations. Looking ahead, there are several domains for further exploration and enhancement in the realm of attack detection using DEL techniques. Firstly, future research could focus on investigating novel architectures and algorithms within the DEL framework to improve the model's ability to adapt to dynamic and evolving network environments. Additionally, efforts should be directed towards enhancing the scalability and efficiency of DEL models to accommodate larger datasets and real-time processing requirements. In large-scale deployments, latency overhead, the potential for false positives, and vulnerability to adversarial attacks continue to be issues. Our future work focus in addressing the above issues as well.

Author Contributions Statement

Surya Pavan Kumar Gudla – Conceptualization, Literature Survey, Methodology, Implementation, Result Analysis, Manuscript Writing and Editing
Sourav Kumar Bhoi– Problem Finding, Literature Survey, Methodology, Result validation, Supervision, Manuscript Writing and Editing
Kshira Sagar Sahoo – Model Training, Validation, and Performance Evaluation, Manuscript Writing and Editing
GNV Rajareddy – Result Analysis, Manuscript Editing

All authors have read and agreed to the published version of the manuscript.

Conflict of Interest Statement

The authors declare no conflicts of interest.

Funding Declaration

There is no funding information.

Data Availability Statement

The data is available on request.

Ethical Declarations

Not Applicable.

Acknowledgments

We sincerely thank BPUT Rourkela and PMEC Berhampur for providing the infrastructure to conduct this research work.

Declaration of Generative AI in Scholarly Writing

Not Applicable.

Abbreviations

The following abbreviations are used in this manuscript:
ML-Machine Learning
EL-Ensemble Learning
DL - Deep Learning
DEL-Deep Ensemble Learning
CNN-Convolutional Neural Network
LSTM-Long Short-Term Memory
GRU- Gated Re-current Units
LR-Logistic Regression
RF-Random Forest
VC-Voting Classifier
Acy- Accuracy
Prn-Precision
Rcl-Recall
F1-S- F1-Score
IoVT- Internet of Vehicle Things
IoMT-Internet of Medical Things
ITS- Intelligent Transportation Systems
CN-Cloud Node
FN-Fog Node
GW-Gateways
BS-Base Stations

References

[1] Ammar Odeh and Anas Abu Taleb. Ensemble-based deep learning models for enhancing IoT intrusion detection. Applied Sciences, 13(21):11985, 2023.

[2] Surya Pavan Kumar Gudla, Sourav Kumar Bhoi, Soumya Ranjan Nayak, Krishna Kant Singh, Amit Verma, and Ivan Izonin. A deep intelligent attack detection framework for fog-based IoT systems. Computational Intelligence and Neuroscience, 2022(1):6967938, 2022.
[3] SuryaPavanKumarGudla, SouravKumarBhoi,SoumyaRanjanNayak,andAmitVerma. Di-ads: Adeepintelligent distributed denial of service attack detection scheme for fog-based IoT applications. Mathematical Problems in Engineering, 2022(1):3747302, 2022.
[4] G Shruthi, Monica R Mundada, S Supreeth, and Bryan Gardiner. Deep learning-based resource prediction and mutated leader algorithm enabled load balancing in fog computing. International Journal of computer networks and information security, 15(4):84–95, 2023.
[5] Shahnawaz Ahmad, Iman Shakeel, Shabana Mehfuz, and Javed Ahmad. Deep learning models for cloud, edge, fog, and IoT computing paradigms: Survey, recent advances, and future directions. Computer Science Review, 49:100568, 2023.
[6] Dipti Prava Sahu, Biswajit Tripathy, and Leena Samantaray. Optimized intrusion detection system in fog computing environment using automatic termination-based whale optimization with elm. International Journal of Computer Network and Information Security, 16(2):79–91, 2024.
[7] SamaaYTarabay, AbeerTwakol, AhmedSSamrah, andYasserIbrahim. Asecure and efficient cryptography system based on chaotic maps for securing data image in fog computing. International Journal of Computer Network and Information Security, 15(1):64–80, 2023.
[8] Riccardo Lazzarini, Huaglory Tianfield, and Vassilis Charissis. A stacking ensemble of deep learning models for IoT intrusion detection. Knowledge-Based Systems, 279:110941, 2023.
[9] Vikas Tomer and Sachin Sharma. Detecting IoT attacks using an ensemble machine learning model. Future Internet, 14(4):102, 2022.
[10] Vibekananda Dutta, Michał Chora´ s, Marek Pawlicki, and Rafał Kozik. A deep learning ensemble for network anomaly and cyber-attack detection. Sensors, 20(16):4583, 2020.
[11] Fatmah Alanazi, Kamal Jambi, Fathy Eassa, Maher Khemakhem, Abdullah Basuhail, and Khalid Alsubhi. Ensem ble deep learning models for mitigating ddos attack in software-defined network. Intelligent Automation & Soft Computing, 33(2), 2022.
[12] Ammar Mohammed and Rania Kora. A comprehensive review on ensemble deep learning: Opportunities and challenges. Journal of King Saud University-Computer and Information Sciences, 35(2):757–774, 2023.
[13] Muhammad Usama Tanveer, Kashif Munir, Madiha Amjad, Syed Ali Jafar Zaidi, Amine Bermak, and Atiq Ur Rehman. Ensemble-guard IoT: A lightweight ensemble model for real-time attack detection on imbalanced dataset. IEEE Access, 2024.
[14] Ayoob Almotairi, Samer Atawneh, Osama A Khashan, and Nour M Khafajah. Enhancing intrusion detection in IoT networks using machine learning-based feature selection and ensemble models. Systems Science & Control Engineering, 12(1):2321381, 2024.
[15] Veeranna Gatate and Jayashree Agarkhed. Enhancing intrusion detection in wireless sensor networks through deep hybrid network empowered by sc-attention mechanism. Iran Journal of Computer Science, pages 1–12, 2024.
[16] Nurhadi Wijaya, Marselina Endah Hiswati, and Sarah Anjani. Deepidx: sophisticated ids model using the generative adversarial network (gan) algorithm. Iran Journal of Computer Science, 5(3):197–204, 2022.
[17] Shahid Ul Haq and Ash Mohammad Abbas. Siad: a traffic generation-based anomaly detection protocol with state improvement for internet of things using deep learning. Iran Journal of Computer Science, pages 1–15, 2024.
[18] Tamara Al-Shurbaji, Mohammed Anbar, Selvakumar Manickam, Iznan H Hasbullah, Nadia ALfriehate, Basim Ah mad Alabsi, Ahmad Reda Alzighaibi, and Hasan Hashim. Deep learning-based intrusion detection system for detecting IoT botnet attacks: A review. IEEE Access, 2025.
[19] TuanAnhTang,LotfiMhamdi,DesMcLernon,SyedAliRazaZaidi, MounirGhogho,andFadiElMoussa. Deepids: Deep learning approach for intrusion detection in software defined networking. Electronics, 9(9):1533, 2020.
[20] Fazlullah Khan, Mian Ahmad Jan, Ryan Alturki, Mohammad Dahman Alshehri, Syed Tauhidullah Shah, and Ateeq ur Rehman. A secure ensemble learning-based fog-cloud approach for cyberattack detection in iomt. IEEE Trans actions on Industrial Informatics, 19(10):10125–10132, 2023.
[21] MPRamkumar, T Daniya, P Mano Paul, and S Rajakumar. Intrusion detection using optimized ensemble classifica tion in fog computing paradigm. Knowledge-Based Systems, 252:109364, 2022.
[22] Prabhat Kumar, Govind P Gupta, and Rakesh Tripathi. A distributed ensemble design based intrusion detection sys tem using fog computing to protect the internet of things networks. Journal of ambient intelligence and humanized Computing, 12(10):9555–9572, 2021.
[23] Poulmanogo Illy, Georges Kaddoum, Christian Miranda Moreira, Kuljeet Kaur, and Sahil Garg. Securing fog-to things environment using intrusion detection system based on ensemble learning. In 2019 IEEE wireless communi cations and networking conference (WCNC), pages 1–7. IEEE, 2019.
[24] Leo Breiman. Classification and regression trees. Routledge, 2017.
[25] Leo Breiman. Random forests. Machine learning, 45:5–32, 2001.
[26] Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. Journal of computer and system sciences, 55(1):119–139, 1997.
[27] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, pages 785–794, 2016.
[28] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. Catboost: unbiased boosting with categorical features. Advances in neural information processing systems, 31, 2018.
[29] Guolin Ke, QiMeng,ThomasFinley, TaifengWang, WeiChen, WeidongMa, QiweiYe, andTie-YanLiu. Lightgbm: Ahighly efficient gradient boosting decision tree. Advances in neural information processing systems, 30, 2017.
[30] Singal Gaurav Mukhopadhyay Debajyoti Ahuja, Nisha. Ddos attack sdn dataset. mendeley Data, V1, 2020.
[31] Kiran Mahesh. Nsl-kdd dataset, 2020. Accessed: 2023-06-26.
[32] Nour Moustafa and Jill Slay. 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, 2015.

[33] Imtiaz Ullah and Qusay H Mahmoud. A scheme for generating a dataset for anomalous activity detection in IoT networks. In Canadian conference on artificial intelligence, pages 508–520. Springer, 2020.

Authors' Profiles



Surya Pavan Kumar Gudla is currently pursuing his Ph.D. from Department of Computer Science and En gineering, Biju Patnaik University of Technology (a state govt. university), Rourkela, India. He received his MCA and MTech in CSE Engg from Jawaharlal Nehru Technological University Kakinada (JNTUK), India. He is currently working as Asst.Prof. in Department of Computer Science and Engineering Aditya Institute of Technology and Management (AITAM), Tekkali, Andhra Pradesh. His interested areas are Computer Networks, Mobile Computing and DataMining. He has published 15 research papers in reputed international journals and conferences.



Sourav Kumar Bhoi (Senior Member, IEEE and Member, ACM) Sourav Kumar Bhoi received the Ph.D. degree and M.Tech degree from the Department of Computer Science and Engineering, National Institute of Technology (NIT), Rourkela, India, in 2017 and 2013, respectively. He is currently an Assistant Professor with the Department of Computer Science and Engineering, Parala Maharaja Engineering College (a Government Engineering College), Berhampur, India. He has nearly ten years of teaching and research experience. He also completed his Post Doc in CSIS programme during 2022-23 from SU, MAN. He also completed a three-month CSIR Summer Research Training Program in online mode from NEIST, Jorhat, Assam (Government of India), in August 2020. He acted as Project investigators for research projects sponsored by MHRD and AICTE. He is an India Book of Record Holder for his research work, in 2022. He has more than 140 research publications in reputed international journals, conferences, books, book chapters, technical articles, patents, and theses. His research interests include machine learning, the Internet of Things, edge and fog computing, ad hoc and sensor networks, and information security. He is a member of many professional bodies, such as a member of IAENG, a Life Member of CSI, an Associate Member of IEI, and a fellow of SIESRP. He acted as a member of TPC and the session chair of many international conferences. He received the prestigious IET Premium Award from IET Networks journal, in 2016. He also received many other awards and honors, such as the University Foundation Day Faculty Research Award in CSE and the Sadananda Memorial Award from the Institution of Engineers (India), in 2021 and 2020, respectively. He was a reviewer for many international journals and conferences. He delivers several invited talks in reputed organizations.



Kshira Sagar Sahoo (Senior Member, IEEE) received the Ph.D. degree in computer science and engineering from the National Institute of Technology (NIT), Rourkela, India, in 2019 and master's degree in information and communication technology from the Indian Institute of Technology (IIT), Kharagpur, India, in 2014. He is currently working as an Assistant Professor in the Department of Computer Science and Engineering at SRM University, Andhra Pradesh, India. He was a post doctoral Kempe fellow with the Autonomous Distributed Systems Laboratory, Department of Computing Science, Ume˚a University, Ume˚a Sweden from Oct 2022 to Sept 2024. His research interests include SDN, edge computing, IoT, Industrial IoT, 5G, urban anomaly detection, and machine learning. He has published more than 80 research papers in the leading international journals and conferences. He is a Senior Member of the IEEE Computer Society and an Associate Member of the Institute of Engineers India.



Dr. Goluguri N. V. Rajareddy is a Postdoctoral Research Fellow at the Department of Chemical and Biological Engineering, University of Saskatchewan, Canada. He received his Ph.D. in Computer Science and Engineering from the National Institute of Technology Silchar, India. His research focuses on Artificial Intelligence, Computer Vision, Edge Intelligence, and their applications in agriculture and food processing. He has authored over 17 research articles in SCI/Scopus-indexed journals and has contributed to interdisciplinary research in deep learning, digital twin systems, and intelligent sensing frameworks.

How to cite this paper: Surya Pavan Kumar Gudla, Sourav Kumar Bhoi, Kshira Sagar Sahoo, Goluguri N. V. Rajareddy, "Securing Fog-assisted IoT: An Adaptable and Efficient Threat Identification Approach", International Journal of Computer Network and Information Security(IJCNIS), Vol.18, No.2, pp.78-93, 2026. DOI:10.5815/ijcnis.2026.02.05