Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X Volume 4, Issue 6, 2026

EDGE AI-BASED MODELS FOR DETECTING SPOOFING ATTACK IN RESOURCE-CONSTRAINED IOT NETWORKS

Abrar Akram[*1], Khalid Hussain[2], Shoaib Ahmad Hashmi[3], Anam Irshad[4]

[1,2,3,4]*Faculty of Computer Science and Information Technology, the Superior University Lahore, Pakistan*

[*1]abrar.akram61@gmail.com, [2]khalidhussain.fsd@superior.edu.pk, [3]shoaibhashmi7860@gmail.com, [4]anamirshad364@gmail.com

DOI: https://doi.org/10.5281/zenodo.20637659

Keywords
IoT security, Edge AI, Spawning attack detection, Intrusion detection system, Logistic Regression, UNSW-NB15, Feature selection, Lightweight machine learning.

Article History
Received: 11 April 2026
Accepted: 23 May 2026
Published: 11 June 2026

Copyright @Author
Corresponding Author: *
Abrar Akram

Abstract
The Internet of Things (IoT) has created an immense new space for the cyber bad guys to attack in such sectors as healthcare, industry automation and smart infrastructure. A profiling attack is particularly concerning in the IoT environment, since it can mimic the typical behavior of a legitimate device, thereby enabling the attacker to gain access to the network and access the information without triggering the security alerts. Traditional IDSs are not appropriate for IoT because they are centralized systems, require huge amount of computation resources, and most of the IoT end-points don't have those resources. A light-weight Edge AI based IDS system is proposed in this paper, which is specially designed for detecting the spoofing attack in resource-constrained IoT network. A structured machine learning pipeline is applied to the standard dataset UNSW-NB15, which includes data cleaning of duplicated data, encoding labels, data normalisation using the StandardScaler, feature selection using correlation-based feature selection to select 15-25 most important features and binary classification using LogisticRegression with L2 regularization parameter of 0.1. A Random Forest (RF) classifier is used to evaluate the accuracy of detection and computational cost of the proposed model. The results from experiments indicate that the accuracy of Logistic Regression model is 95.46%, precision is 95.53%, recall is 96.26% and F1 score is 95.89%. Despite its simplicity, the model is still very competitive in terms of detection performance and suitability in the edge environment in terms of memory, latency and processing requirements (vs. Random Forest with 93.12% accuracy). These results show that with proper optimization of lightweight models, processing and feature engineering, it is possible to obtain a dependable real-time IDS with low computational requirements. This framework offers a workable and scalable security solution for use at the network edge in today's modernistic environment of IoTs.

I. INTRODUCTION

Internet of Things (IoT) has gone beyond the conceptual stage and is becoming a key building block in the present digital infrastructure. These days, billions of devices are diversified, continuously generate, share and react to data without man's interference. This universal connectivity has ushered in a new level of

efficiency in many industries, but it has also made it possible to exploit most of them, as the attackers have a huge and complex target to choose from (Al-Fuqaha et al., 2015).

Because of their limited computational power, IoT devices are bound to have a limited lifespan. Most of the IoT endpoints have a limited amount of processing power, memory and energy.Most IoT endpoints are constrained in terms of their processing power, memory and energy. These are not accidental but intended – they allow devices to be small, cheap and usable in situations where conventional computing devices would be impractical. The same constraints, however, make it extremely challenging to put strong, real-time security into the devices. This leads to a lot of IoT deployments that depend on perimeter security solutions, which are ineffective against high-level (network level) attacks. (Roman et al., 2013)

Spoofing attacks are a particularly hazardous form of cyber attack against IoT systems, given the variety of cyber taxonomy attacks against such systems. In a spoofing attack, the adversary is trying to pretend to be a legitimate network entity by creating a fake device identifier (IP, MAC or authentication credentials). Once the attacker is able to disguise themselves, they could be able to intercept sensitive communications, input malicious commands, steal data or act as a pivot point to further network penetration. Spoofing can be difficult to detect by signature-based or anomaly-based detection systems, and can be insidious because it can infiltrate the traffic stream without being noticed (Sicari et al., 2015).

Traditional Intrusion Detection Systems (IDS) are good in enterprise infrastructures, but fall short in terms of structure for IoT needs. Thus, signature-based systems are not able to identify new attack forms and anomaly-based systems need significant compute power for real-time baseline modeling. Generally, both paradigms use centralized architectures and send raw traffic data to remote servers for processing, which is undesirable because it causes unacceptable latency, high bandwidth usage and creates single points of failure (Neshenko et al., 2019).

Edge Artificial Intelligence (Edge AI) is an intriguing paradigm. Edge AI allows for threat detection to be done locally, near to the edge node (such as a gateway, a router or a proximate computing device), with low latency, and without relying on cloud architecture. This not only satisfies data privacy requirements, which have become increasingly important in many industries, including healthcare and industrial IoT, but also provides security for data during its application (Shi et al., 2016).

In this paper following contributions are made: (i) A systematic preprocessing pipeline especially for IoT network traffic datasets for IoT security applications; (ii) A correlation-based feature selection strategy which is able to reduce the dimensionality of network traffic data without compromising the discriminative power of the data; (iii) A lightweight Logistic Regression Classifier which is optimized with L2 regularization for edge deployment; and (iv) A comparative evaluation with Random Forest to show that simple classification can be as effective as complex classification in IoT security applications.

This paper is organized as follows: The existing work related to IoT security, IDS architectures, machine learning based detection and Edge AI are discussed in Section II. In Section III the proposed methodology will be described. Experimental Results and discussion are given in Section IV. This paper is concluded in Section V, with an indication of what can be done in the future.

II. RELATED WORK

A. IoT Security Challenges

The security posture of IoT environments is fundamentally undermined by three intersecting factors: device heterogeneity, resource scarcity and the lack of standardised security frameworks. The main vulnerability vectors are consistently identified as weak or absent authentication mechanisms, unencrypted communication channels and infrequent firmware updates (Andrea et al., 2015). IoT networks are typically physically distributed and publicly accessible and therefore vulnerable to remote cyber attacks as well as physical tampering, a threat profile qualitatively different from that of traditional enterprise networks (Frustaci et al., 2018).

B. Intrusion Detection in IoT Environments

Signature-based IDS methods are computationally efficient and explainable, however, they are inherently reactive and can only detect threats that are known to have attack signatures, and cannot detect zero-day exploits or variants of attacks. Anomaly-based systems improve generalisation by modelling normal behaviour and flagging deviations, but suffer from high false positive rates when deployed in dynamic, non-stationary traffic environments typical of IoT networks (Diro & Chilamkurti, 2018). Notably, the most current IDS solutions are designed for centralised, resource-rich environments and cannot be applied directly to IoT endpoints without a significant architectural rethinking (Meidan et al., 2018).

C. Machine Learning in Intrusion Detection

Machine learning has become the dominant paradigm for next-generation IDS, which is able to learn complex non-linear decision boundaries from labelled traffic data. Methods that involve ensemble such as Random Forest and Gradient Boosting have been shown to have high accuracy on standard benchmark datasets (Ferrag et al., 2020). Deep learning architectures, such as Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNNs), have shown promise for sequence-based traffic analysis (Kim & Suh, 2019). However, these approaches usually demand high memory and computational resources which makes them impractical to directly deploy on resource-constrained IoT hardware. A common shortcoming in the literature is the focus on detection accuracy at the cost of computational viability, resulting in a disconnect between laboratory performance and real-world deployability (Hindy et al., 2020).

D. Data optimisation and feature selection

Feature selection was found to be very important for lightweight IDS models. Correlation based methods, Principal Component Analysis (PCA) and wrapper based methods have all shown the ability to reduce the input dimensionality significantly without proportional reductions in detection accuracy (Tan et al., 2014). This study shows that the training of models using carefully selected subsets of features can not only expedite the process but also enhance their generalisation to unseen traffic patterns, reducing the risk of overfitting which poses a major challenge in high dimensional network traffic data (Bhatt et al., 2020).

E. Edge AI for IoT Security

Cloud-based security architectures have been recognised as a structural solution for latency and privacy limitations (Bonomi et al., 2012). Edge AI refers to the deployment of AI inference at the edge of the network, enabling on-the-spot threat detection with low round trip communication lag and less reliance on persistent cloud connectivity. Studies have shown that edge-based detection systems can reduce latency and bandwidth consumption significantly (Hu et al., 2018). But the challenge of developing AI models small enough to run within the memory and power envelopes of edge hardware remains a research frontier.

F. Gap in research

Significant progress has been achieved in the above constituent areas, yet there is currently no single lightweight, accurate, edge-deployable solution that has been rigorously tested against spoofing attacks in IoT environments. Existing techniques tend to optimise for one or two of these properties at the expense of others. The aim of the present work is to fill this gap by proposing an integrated framework that gives priority to computational efficiency besides detection accuracy, validated on a broadly accepted benchmark dataset with spoofing-relevant attack categories.

III. METHODOLOGY

A. Dataset

The main data source for this study is the UNSW-NB15 dataset which is created at the University of New South Wales using the IXIA PerfectStorm tool to generate network traffic. The dataset consists of network traffic records that cover normal activity and nine classes of synthetic attack traffic, including Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance,

Shellcode, and Worms, many of which display behavioural characteristics that are relevant to spoofing-based intrusions (Moustafa & Slay, 2015). The dataset contains about 80,000 records and 45 features divided into four categories: basic flow statistics (number of packets, byte sizes, connection duration), content features (payload properties, session behaviour), temporal features (time between packets, session duration), and statistical summaries (mean, variance of traffic flow metrics). The binary target label represents the normal traffic (class 0) and malicious traffic (class 1).The UNSW-NB15 dataset was selected over legacy benchmarks such as KDD Cup 99 due to its representation of contemporary attack patterns and its broader adoption in recent IDS research (Sharafaldin et al., 2018).

B. Data Preprocessing

Datasets of raw network traffic always need extensive preprocessing before they can be used to train machine learning models. The pre-processing pipeline implemented in this study consists of the following steps:

Data Cleaning

All records were checked for missing values, inconsistencies and duplicate entries. To avoid data leakage and reduce redundancy, duplicate records were removed. Missing values were handled by targeted imputation or deletion of records, depending on the amount and pattern of missingness.

- **Categorical Encoding:** Features like protocol type, network service, and connection state in the UNSW-NB15 dataset are encoded as categorical text labels. They were transformed into numerical representation by applying label encoding, where each category was assigned a unique integer, thus maintaining the compact representation of the feature.'
- **Feature Scaling:**

The numerical features have different scales across the dataset, which could bias the learning algorithms sensitive to the gradient towards features with higher absolute value. All numerical features were standardised using StandardScaler from the Scikit-learn library, which makes the features have zero mean and unit variance. Thus all features contribute equally to the learning process.

- **Train-Test Partitioning:**

The preprocessed dataset was divided into 80% training and 20% holdout testing set, using stratified random sampling in order to preserve class distribution across the two partitions. The model was only trained on the training set, and the test set was held out until the final evaluation.

C. Feature Selection

The raw dataset contains 45 input features, which creates a high-dimensional learning problem, increasing the computational cost and risk of overfitting, both of which are critical for edge deployment. A correlation-based feature selection approach was employed to identify and retain the most discriminative feature subset. First the pairwise Pearson correlation analysis was performed on all features. Feature pairs exhibiting a correlation coefficient above a defined threshold were identified, and the less predictive member of each redundant pair was eliminated.Second, we correlated each kept feature against the binary target label and removed those that showed negligible relationships with the target's predictive power. This two-stage filtering process resulted in a feature space of 15-25 highly informative attributes, thus preserving the discriminative structure of the data and substantially reducing the computational overhead.

D. Model Selection and Training

Two machine learning algorithms were evaluated as potential classifiers:

Logistic Regression with L2 Regularizer : Logistic regression is a classical linear classifier that estimates the probability of binary class membership by applying a logistic (sigmoid) transform to a weighted linear combination of input features. Its main benefits in this application are its small computational footprint, with low memory consumption and near-instantaneous inference, and its natural interpretability, which allows one to directly inspect the contribution of each feature to the classification decision. L2 regularisation (ridge penalty) with a regularisation

parameter C = 0.1 was used which constrains the magnitude of the coefficients and reduces the over-fitting of the model on the high dimensional UNSW-NB15 feature space.

Random Forest: For comparison, we trained a Random Forest ensemble classifier on the same preprocessed and feature-selected dataset. Random Forest creates a collection of decision trees on subsamples of bootstrapped data, and combines their predictions using majority voting. While this method is typically highly accurate and noise tolerant, it does come with significantly increased memory and inference-time overhead compared to Logistic Regression - a crucial factor affecting edge deployment feasibility.

E. Evaluation Metrics

Model performance was assessed using the following metrics, derived from the confusion matrix:

Table I. Evaluation Metrics Definitions

Metric	Definition
Accuracy	(TP + TN) / (TP + TN + FP + FN) — Overall proportion of correct classifications
Precision	TP / (TP + FP) — Proportion of predicted attacks that are genuine attacks
Recall	TP / (TP + FN) — Proportion of actual attacks correctly identified
F1-Score	2 x (Precision x Recall) / (Precision + Recall) — Harmonic mean of Precision and Recall

IV. RESULTS AND DISCUSSION

A. Performance Comparison

Table II summarizes the performance of both classifiers on the holdout test set following preprocessing and feature selection.

Table II. Comparative Performance of Evaluated Classifiers

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression (L2, C=0.1)	95.46%	95.53%	96.26%	95.89%
Random Forest	93.12%	92.87%	93.45%	93.16%

B. Logistic Regression Performance Analysis

Logistic Regression (LR) consistently and robustly performed across all evaluation metrics. The model accurately detected the huge majority of the instances of both normal traffic and spoofing-related malicious activity, with a classification accuracy of 95.46% on the holdout test set. The 95.53% accuracy means that the false alarm rate is very low. If the system detects an intrusion it is very probably a real threat. This property is of practical importance since too many false positives will erode operator trust and waste processing power on devices with limited capacity.

The recall performance of 96.26% indicates that the model successfully identified almost all actual attack instances which is an important requirement in IoT security, where missing a real attack can have serious operational or safety consequences. The F1-score of 95.89% shows that the model has a good trade-off between precision and recall, which confirms the reliable and consistent detection capability of the model in the inherently imbalanced setting of network intrusion detection.

C. Comparative Analysis with Random Forest

The Random Forest model achieved an accuracy of 93.12% with a precision of 92.87%, recall of 93.45% and F1-score of 93.16%, which are considerably lower than the Logistic Regression model. This is a directionally counter-intuitive result: one might expect that the greater model

capacity of an ensemble method would produce better classification performance . This observed result can be explained by the efficiency of the preprocessing and feature selection pipeline. After removing redundant and low discriminative features, the classification problem was sufficiently linear that dominant decision boundaries can be captured by a logistic model, without the need for the non-linear partitioning capability of an ensemble.

Critically, Random Forest has a significantly worse computational profile for edge deployment. The memory needed to store a trained ensemble of decision trees is orders of magnitude greater than that of a single logistic model. Random Forest also has much higher inference time due to the sequential traversal of several trees. These differences are operationally decisive in IoT edge environments where memory is measured in kilobytes and real-time detection latency is a hard constraint. Hence, Logistic Regression is the best option not only on detection metrics but on all the criteria relevant to edge-deployable IoT security.

D. Impact of Feature Selection

The feature selection process using correlation reduced the feature space from 45 to 15-25 attributes without any measurable drop in classification performance. This dimensionality reduction helped speed up model training and inference, and reduced memory requirements. It also helped the model generalise strongly by removing noise and redundancy from the input representation. These findings validate the known fact that feature selection is a key enabler of light-weighted accurate IDS models for resource constrained environments (Bhatt et al., 2020).

E. Edge Deployment Suitability

The Logistic Regression model has a computational profile that makes the proposed system suitable for edge deployment. The parameter storage is minimal, a vector of weights proportional to the number of selected features, and inference is a single dot product followed by a sigmoid activation. These operations are well within the computational envelope of modern IoT gateways and edge processors. This allows the system to perform classification in real time with inference latency of less than a millisecond on commodity edge hardware without the need for cloud connectivity or off-loading the data to remote servers.

V. CONCLUSION

In this paper, we proposed a lightweight Edge AI-based Intrusion Detection System to detect spoofing attacks in resource-constrained IoT environments. By implementing a pipeline of data cleaning, categorical encoding, feature normalisation, correlation based feature selection and L2 regularised Logistic Regression classification, the proposed framework achieved an accuracy, precision, recall and F1-score of 95.46%, 95.53%, 96.26% and 95.89% respectively on the UNSW-NB15 benchmark dataset.

The experimental comparison with Random Forest shows that rigorous preprocessing and feature selection are not a prerequisite for high detection accuracy in terms of model complexity. The small computational footprint, interpretability and fast inference time of the Logistic Regression model makes it particularly suitable for deployment on IoT edge nodes where memory, power and latency constraints are binding.

The proposed work has important implications for IoT system architects and security engineers looking for low-cost, privacy-preserving, scalable alternative solutions to cloud-centric security architectures. By pushing threat detection intelligence to the network edge, sensitive traffic data doesn't need to pass through external networks, inherently lowering interception risk and enabling data-sensitive domains to meet regulatory compliance.

It is important to note a number of limitations. The UNSW-NB15 dataset is broadly representative, but it was not constructed solely from the traffic of IoT devices, and performance may vary on deployments with highly device-specific traffic profiles. Moreover, the real-time streaming deployment conditions are not considered in the current evaluation.

Future work may focus on testing the proposed model on additional IoT-based datasets such as N-BaIoT to evaluate its performance in different network environments. Further improvements can also be made by applying federated learning techniques, which allow multiple edge devices to update the model collaboratively without sharing sensitive raw data. In addition, future studies may explore optimized deep learning approaches, including lightweight and compressed neural network models, to improve edge deployment efficiency. Another important direction is the practical implementation and performance evaluation of the proposed system on real IoT edge hardware in order to analyze its real-time detection capability, latency, and resource consumption in real-world scenarios.

REFERENCES

Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of Things: A survey on enabling technologies, protocols, and applications. IEEE Communications Surveys & Tutorials, 17(4), 2347-2376. https://doi.org/10.1109/COMST.2015.2444095

Andrea, I., Chrysostomou, C., & Hadjichristofi, G. (2015). Internet of Things: Security vulnerabilities and challenges. Proceedings of the IEEE Symposium on Computers and Communication (ISCC), 180-187. https://doi.org/10.1109/ISCC.2015.7405513

Bhatt, S., Anand, P., & Bhatt, Y. (2020). Novel features for network intrusion detection system. Proceedings of the International Conference on Inventive Computation Technologies (ICICT), 806-811. https://doi.org/10.1109/ICICT48043.2020.9112548

Bonomi, F., Milito, R., Zhu, J., & Addepalli, S. (2012). Fog computing and its role in the Internet of Things. Proceedings of the 1st Edition of the MCC Workshop on Mobile Cloud Computing, 13-16. https://doi.org/10.1145/2342509.2342513

Diro, A., & Chilamkurti, N. (2018). Distributed attack detection scheme using deep learning approach for Internet of Things. Future Generation Computer Systems, 82, 761-768. https://doi.org/10.1016/j.future.2017.08.043

Ferrag, M. A., Maglaras, L., Ahmim, A., Derdour, M., & Janicke, H. (2020). RDTIDS: Rules and decision tree-based intrusion detection system for Internet-of-Things networks. Future Internet, 12(3), 44. https://doi.org/10.3390/fi12030044

Frustaci, M., Pace, P., Aloi, G., & Fortino, G. (2018). Evaluating critical security issues of the IoT world: Present and future challenges. IEEE Internet of Things Journal, 5(4), 2483-2495. https://doi.org/10.1109/JIOT.2017.2767291

Hindy, H., Brosset, D., Bayne, E., Seeam, A., Tachtatzis, C., Atkinson, R., & Bellekens, X. (2020). A taxonomy of network threats and the effect of current datasets on intrusion detection systems. IEEE Access, 8, 104650-104675. https://doi.org/10.1109/ACCESS.2020.2999Cecil

Hu, P., Dhelim, S., Ning, H., & Qiu, T. (2018). A survey of intrusion detection on industrial control systems. International Journal of Distributed Sensor Networks, 14(8). https://doi.org/10.1177/1550147718794411

Kim, T., & Suh, B. (2019). Toward an interpretable deep learning model for mobile malware detection and family identification. Computers, 8(3), 58. https://doi.org/10.3390/computers8030058

Meidan, Y., Bohadana, M., Mathov, Y., Mirsky, Y., Shabtai, A., Breitenbacher, D., & Elovici, Y. (2018). N-BaIoT: Network-based detection of IoT botnet attacks using deep autoencoders. IEEE Pervasive Computing, 17(3), 12-22. https://doi.org/10.1109/MPRV.2018.03367731

Moustafa, N., & Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems. Proceedings of the Military Communications and Information Systems Conference (MilCIS), 1-6. https://doi.org/10.1109/MilCIS.2015.7348942

Neshenko, N., Bou-Harb, E., Crichigno, J., Kaddoum, G., & Ghani, N. (2019). Demystifying IoT security: An exhaustive survey on IoT vulnerabilities and a first empirical look on Internet-scale IoT exploitations. IEEE Communications Surveys & Tutorials, 21(3), 2702-2733. https://doi.org/10.1109/COMST.2019.2910750

Roman, R., Zhou, J., & Lopez, J. (2013). On the features and challenges of security and privacy in distributed Internet of Things. Computer Networks, 57(10), 2266-2279. https://doi.org/10.1016/j.comnet.2012.12.018

Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP), 108-116. https://doi.org/10.5220/0006639801080116

Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge computing: Vision and challenges. IEEE Internet of Things Journal, 3(5), 637-646. https://doi.org/10.1109/JIOT.2016.2579198

Sicari, S., Rizzardi, A., Grieco, L. A., & Coen-Porisini, A. (2015). Security, privacy and trust in Internet of Things: The road ahead. Computer Networks, 76, 146-164. https://doi.org/10.1016/j.comnet.2014.11.008

Tan, Z., Jamdagni, A., He, X., Nanda, P., & Liu, R. P. (2014). A system for denial-of-service attack detection based on multivariate correlation analysis. IEEE Transactions on Parallel and Distributed Systems, 25(2), 447-456. https://doi.org/10.1109/TPDS.2013.146

IEER
Institute for Excellence in Education & Research