International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 – 5161
Available at http://inpressco.com/category/ijcet

Research Article

Anomaly Identification in Real-Time for Predictive Analytics in IoT Sensor Networks using Deep

Siddhesh Amrale*

Independent Researcher

Received 01 Dec 2024, Accepted 20 Dec 2024, Available online 23 Dec 2024, **Vol.14, No.6 (Nov/Dec 2024)**

Abstract

The proliferation of IoT devices has exposed networks to an increased risk of cyberattacks, as their number is increasing at an exponential rate. This has resulted in more sophisticated methods for identifying outliers. This research suggests a DL architecture for the Internet of Things (IoT) sensor network based on the Long Short-Term Memory (LSTM) model for anomaly detection. The ToN-IoT data on Kaggle was utilized and it was undergone through numerous preprocessing processes like missing values, label encoding, normalization and class balancing through the use of SMOTE. The number of IoT devices is growing at an exponential rate and this has augmented the susceptibility of networks to cyber threats. This has led to advanced techniques of detecting abnormalities. The results prove that the model is accurate and valid for identifying typical and non-standard network behavior. The proposed framework is an intelligent framework of detecting anomalies in an intelligent way to enhance the security of the IoT network on a scalable, high-quality, and real-time framework.

Keywords: *Internet of Things (IoT), Anomaly Detection, Cybersecurity, Machine Learning, Deep Learning*

Introduction

The IoT has grown rapidly, revolutionizing the industries today, including healthcare, agriculture, transportation, and manufacturing, through smart connectivity and automation of billions of sensor devices [1]. Such integrated mechanisms create enormous amounts of real-time information, which provides useful possibilities of forecast analytics and intelligent decision-making. The IoT poses unique challenges to network security due to the increased exposure of connected devices to cyber-attacks, illegal access, and aberrant system behavior [2][3]. Data integrity, operational reliability, and system security are ensured by the real-time detection of anomalies in IoT sensor networks.

Anomaly detection is the practice of finding data points that don't fit the pattern of expected occurrences. Abnormalities in the IoT setting can represent a potential cyberattack, hardware failures, or abnormal network activity [4]. Such abnormalities can be detected, ensuring a better reliability, security, and performance of the IoT systems. Traditional intrusion detection and rule-based approaches often fail due to the high dimensionality, streaming nature, and heterogeneity of data from the IoT, resulting in poor detection accuracy and high false alarm rates [5][6].

A combination of AI and DL has become an efficient method of making IoT networks more secure by offering intelligent and flexible detection devices [7]. Since its publication in 1956, AI has been developed to provide data-based solutions that have the capability of examining complicated associations and discovering hidden patterns in massive quantities of data [8][9]. Machine learning and deep learning are crucial to the development of AI-powered intelligent cyber protection systems. The ability of DL models to successfully detect abnormalities on real-time instances is based on their ability to offer effective temporal correlations among sequential IoT data [10]. Such models not only increase threat detection as well as prediction of attacks but also form the basis of proactive decision-making in predictive analytics.

Motivation and Contributions of the Study

The devices that are linked to the IoT ecosystem have grown exponentially as the ecosystem continues to gain momentum in the smart homes, industries, healthcare, and transportation sectors. This has contributed significantly to the inability to secure the network because IoT tools in most instances lack computing power, and are of low security configuration, hence an easy target for cyber-attacks. The temporal relationships, as well as the varied attacker patterns of the IoT information streams, do not fit the conventional rule-based and fixed machine

*Corresponding author's ORCID ID: 0000-0000-0000-0000

DOI: https://doi.org/10.14741/ijcet/v.14.6.15

learning algorithms. Therefore, there is an immediate need of an intelligent and dynamic anomaly detecting platform whereby it can have the ability to identify malicious activity within seconds. DL and LSTM networks, in particular, can be used to build an IoT network that is both more resilient and reliable through better means of anomaly and high-level sequential dependencies detection. The following are the major findings of the study:

- The ToN-IoT dataset is utilized for intrusion detection research since it contains a variety of realistic and different IoT assault scenarios.
- Development of a robust preprocessing pipeline to handle missing values, scaling, and categorical data encoding.
- Using SMOTE balancing to fix the data class imbalance in IoT intrusion reports.
- The LSTM based anomaly detector is introduced as an anomaly detector that can be applied to IoT sensor networks.
- Integrative review of performance based on numerous metrics that ensure credible evaluation, such as ACC, PRE, REC, F1, and ROC.

Novelty of the Study

The proposed framework is an innovation in itself as it integrates the state-of-the-art preprocessing and data balancing through application of SMOTE with the LSTM networks to be more accurate in identifying anomalies in IoT sensor networks. The way it works is by applying the gated memory architecture of LSTM to learn sequential structure in network traffic, which the traditional models fail to achieve in capturing temporal dependencies or unequal classes, resulting in high detection and low false alarm. The relevance of the model to the realistic environment is also justified by the fact that the ToN-IoT data that was used in the study involves a heterogenous stream of data of real-life sources of IoT and IIoT. The approach does not only improve the ACC and resiliency of anomaly detection, but it also shows an ability to extrapolate to a wide range of IoT devices and attack situations to offer an appropriate and scalable solution to real-time security in the context of the IoT network.

Organization of the Study

The paper is organized as follows: After the introductory part, Section II examines prior studies that focused on identifying anomalies in IoT networks. The methods are detailed in Section III, and the findings and comparisons are discussed in Section IV. Last but not least, Section V summarizes the study and provides future work directions.

Literature Review

A detailed review and critical analysis of the literature about AD in IoT networks in the context of this study contributed to its narrowing and the formation of the general direction of the research.

Kanwal et al. (2023) Automatic labeling and classification on IoT datasets can be achieved with the help of a hybrid method that combines clustering and classification techniques. There are two functions in the model. The initial step involves classifying dataset instances as normal or abnormal using k-means clustering. The second step in detecting irregularities in IoT networks is training a Random Forest model with tagged datasets. The outcomes demonstrate that the suggested model can identify irregularities in IoT networks with a REC of 98%, ACC of 98%, PRE of 98%, and F-measure of 0.98% [11].

Chevtchenko et al. (2023) explore an investigation into AD mapping for industrial gear that meets this need by utilizing IoT sensors and ML algorithms. This study gives a thorough overview of AD research by critically assessing 84 articles that are relevant to the topic and cover the years 2016–2023. The most popular algorithms, preprocessing methods, and sensor kinds are uncovered by research. Further, this analysis highlights potential areas of application and highlights both current and future research obstacles and opportunities [12].

Sarwar et al. (2023) built a method to detect suspicious activities in smart homes using machine learning and many classifiers. Execute assessments and tests utilizing the BoT IoT dataset obtained from UNSW. Using data collected from Internet of Things devices, machine learning models are constructed using four classifiers. While ANN has Weighted PRE of 0.98%, REC of 0.96%, and F1 of 0.96% on the Test dataset, decision tree, random forest, and AdaBoost all get scores of 1. As the results demonstrate, the suggested methodology is capable of producing results that are both very accurate and quite resilient [13].

Garg et al. (2022) investigate any discrepancies with previously widely used security protocols, like those for wired or wireless networks. The goal of this research is to evaluate how well three well-known ML algorithms—KNN, CNN, and NB, detect CoT outliers. The analysis of outliers on the BotIoT dataset, generated through the modeling of IoT devices and routers. With an F1 of 99.5%, CNN achieves an ACC of 99.94% [14].

Dubey, Pandey and Kumar (2021) Software failure, hardware failure, or a breakdown in communication could have caused the errors. This research presents a method for detecting software failures in WSNs that are enabled by the IoT and has taken into account a wide variety of defects in IoT-enabled WSNs, including offset, gain, stuck at, and data loss issues. The majority of the time, suggested system achieves a 99% ACC rate, as shown by the simulation results [15].

Sahu and Mukherjee (2020) aforementioned classification process is run on the entire 3.5 lakh dataset. A different approach would be to execute all classification algorithms without the "value" attribute, which can only have a value of 0 or 1. An initial dataset is generated for training purposes, while an additional dataset is generated for testing purposes. The data used for training is about 75% of the total, whereas the data used for testing is about 25%. In the first scenario, ANN achieves an ACC of 99.4%, and in the second case, the approach mentioned earlier achieves an ACC of 99.99% [16].

Upman and Goranin (2020) provide a smart system that safeguards IoT datasets against security breaches by detecting anomalies using the Radial Basis Function Network, a Neural Network Technique. The IoT-enabled systems are analyzed for anomalies and assaults using these astute methods. With a FPR of only 0.2%, the suggested approach achieves a test ACC of 99.3% [17].

Ngo et al. (2020) a variety of DNN models for anomaly identification, each tailored to a specific layer in the HEC using a bottom-up approach; the models' degree of complexity varies. For this reason, treat the model selection as a policy network reinforcement learning problem using a one-step Markov decision process. Compare the results of a HEC testbed constructed using the suggested method to those of the actual thing utilizing IoT datasets. In contrast to cloud-based detection jobs, the proposed method considerably decreases detection latency (e.g., by 71.4% for univariate dataset) while keeping ACC [18].

The most recent studies on AD in IoT networks are summarized in Table I. The table includes the studies' methodologies, results, limitations, and suggestions for further research.

Table 1 Recent Studies on Anomaly Detection in IoT Networks

Author	Application	Technique / Model Used	Result	Limitation / Future Work
Kanwal et al., (2023)	Identifying anomalies and automatically categorizing IoT networks	K-Means clustering and Random Forest classification are used in a hybrid technique.	REC: 98%, ACC: 98%, PRE: 98%, and F-measure: 0.98.	Restraining to static datasets; research in the future should investigate streaming data in real-time and adaptive clustering.
Chevtchenko et al., (2023)	Machines Used in Industrial IoT for Anomaly Detection (AD)	Systematic mapping study analyzing 84 research papers (2016–2023)	Classified the most popular ML models, preprocessing methods, and sensor kinds	Need for standardized datasets and frameworks; future research should address scalability and edge intelligence integration.
Sarwar et al., (2023)	IoT systems for smart homes: identifying anomalies	Decision Tree, ANN, Random Forest, and AdaBoost all make use of the UNSW-BoT IoT dataset.	RF, DT, AdaBoost achieved Weighted PRE, REC, F1 = 1; ANN = 0.98, 0.96, 0.96	Focused on limited IoT devices; future work should include larger and more diverse datasets to improve generalization.
Garg et al., (2022)	Security AD in CoT networks	KNN, CNN, and Naïve Bayes on BoT-IoT dataset	CNN achieved 99.94% ACC, 99.5% F1	Computational cost for deep models is high; future work should consider lightweight CNN architectures for edge deployment.
Dubey, Pandey, and Kumar (2021)	Software fault detection in IoT-enabled WSNs	Random Forest	Achieved 99% ACC in simulations	Focused only on software faults; future research should include hardware and communication-related faults with hybrid ML models.
Sahu and Mukherjee (2020)	Threat and anomaly identification in smart devices and IoT systems	ANN and other classification algorithms	ANN achieved 99.4% (full dataset) and 99.99% (reduced feature set) ACC	Model tested on limited features; future work should explore real-time IoT attack datasets and multi-feature integration.
Upman and Goranin (2020)	Security penetration detection in IoT-enabled systems	Radial Basis Function (RBF) Neural Network	99.3% test ACC, 0.2% FPR	Limited validation on small dataset; future research can focus on adaptive NN models for dynamic IoT environments.
Ngo et al., (2020)	Anomaly detection with adaptive algorithms based on hierarchical edge computing	Deep Neural Network (DNN) with Reinforcement Learning (Contextual Bandit)	Reduced detection delay by 71.4% without ACC loss	Implementation complexity; future work should optimize computational efficiency for real-time edge inference.

Research Methodology

The suggested method progresses from the Kaggle ToN-IoT dataset to the subsequent data preparation procedures, which include managing missing values, min-max scaling, and label encoding. A solution to the problem of class imbalance is to use SMOTE after splitting the dataset 80:20 between the training and testing sets. LSTMs are used for classification, and the efficacy of the model is shown by summarizing the findings and evaluating them using ACC, PRE, REC, and F1. Figure 1 shows how the suggested approach would work.

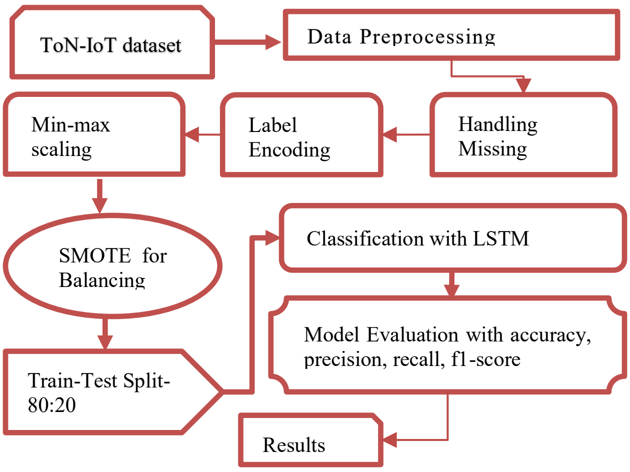

Fig.1 Proposed flowchart for Anomaly Detection in IoT Networks

AD in IoT Networks Flowchart Specifications.

Data Collection and Visualization

In this paper, analyzed a new dataset, ToN IoT, sourced from Kaggle. The ToN-IoT dataset aims to gather and assess IIoT and IoT combined data sources. The data is diverse and comes from a variety of places, including system logs (Windows and Linux), network traffic, and telemetry information from connected devices. Realistic networks provide the basis of the Internet of Things. Below is the dataset visualization:

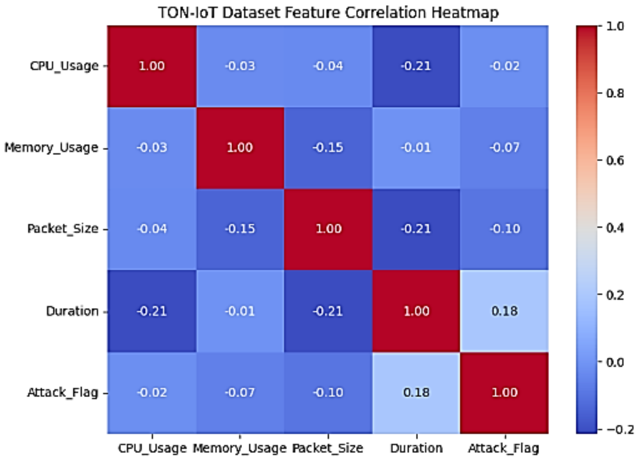

Fig.2 Correlation Heatmap

Figure 2 shows the **feature correlation heatmap** of the **TON-IoT dataset**, illustrating the linear relationships among key features including CPU_Usage, Memory_Usage, Packet_Size, Duration, and Attack_Flag. The intensity of the colors represents the degree of association. Near -1.0, and have a significant negative correlation, while closer to 1.0, show a high positive correlation. CPU_Usage and Duration show a weak negative correlation (−0.21), while Duration and Attack_Flag have a slight positive correlation (0.18), implying that longer durations are marginally associated with potential attacks.

Data Pre-processing

A dataset must undergo pre-processing in order to be cleaned and prepared for analysis [19]. It comprises of many obligations such as missing values, label encoding and min max normalization as discussed below:

- **Handling the missing values:** Determine and fill in the missing values of the dataset. This may be achieved by either filling the missing values through statistical tools or dropping the cases or the variables that have missing values, based on the proportion of missing data and its effect on the research.
- **Label Encoding:** The method of label encoder is applied to transform categorical features into numerical values. This method transforms all categorical numbers existing in the dataset to numbers.
- **Normalization:** Data normalization is done to manipulate the data to prevent the occurrence of negative values which may be harmful to the neural networks. All the data within the dataset are normalized to the range 0 to 1.

Data Balancing with SMOTE

A strategy for producing synthetic samples to equalize underrepresented classes, known as the SMOTE [20]. While SMOTE creates new samples by combining existing instances of the minority class, it can nevertheless provide data that is noisy or contains outliers on occasion. To make the dataset more balanced, SMOTE are used. Synthetic samples from the dataset with the minority class (normal packets) can be created using the SMOTE approach to fix the class imbalance problem. Thus, the ratio of benign to malicious packets is maintained.

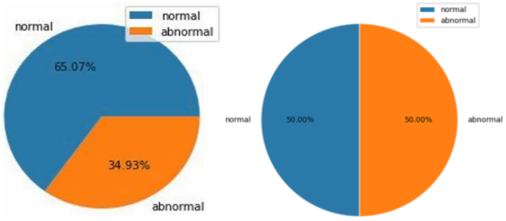

Fig.3 Imbalanced and Balanced Distribution of Classes

Figure 3 shows the Imbalanced and Balanced Distribution of Classes using SMOTE, displaying two pie charts illustrating different class distributions. The chart on the left shows an imbalanced distribution where the "normal" class accounts for 65.07% (blue) and the "abnormal" class accounts for 34.93% (orange). The chart on the right shows a perfectly balanced distribution, with the "normal" and "abnormal" classes each making up exactly 50.00% of the data.

Data Partitioning

The dataset was divided into two halves, one for testing and one for training, using an 80:20 split. Trained predictive models using the training set, and then tested them to see which one performed the best at detecting intrusions.

Classification with LSTM Model

The inability of RNN to memorize critical information from a sequence of inputs suggests the possibility of using an RNN extension known as LSTM. A complete data stream might be processed by such a network architecture. Unlike RNN, this design could have a longer memory [21]. The feedback connections that make up an LSTM allow the network to process both a single data point and the complete sequence of data. The responsibility of determining what data needs to be sent to the cell would fall on an LSTM network input gate. Previously defined and related equations describe the mathematical process, while Equation (1) describes the input gate.

$$i_t = \sigma(W_i * [h_{t-1}, x_t]) + b_i \tag{1}$$

The forget gate is responsible for determining whether or not to retain data from a prior memory state. An Equation (2) describing this process has been referenced as:

$$f_t = \sigma(W_i * [h_{t-1}, x_t]) + b_f (2)$$

The following Equations (3) and (4) illustrate the mathematical process of the update gate, which would update the information in each cell:

$$\tilde{c}_t = tanh(W_c * [h_{t-1}, x_t]) + b_c \tag{3}$$
$$C_t = (f_i * [c_{t-1}, i_t]) * \tilde{c}_t \tag{4}$$

The hidden layer's prior time step would be updated by the output gate. Furthermore, the data's output would be modified by this gate. Here are some Equations (5), (6) to help understand the process:

$$o_t = \sigma(W_v * [h_{t-1}, x_t]) + b_v \tag{5}$$
$$h_t = o_t * \tanh (c_t) \tag{6}$$

Where x_t is the cell's input and h_{t-1}, h_t and c_{t-1}, c_t are the hidden and cell states, respectively. The LSTM model would make use of the remaining variables to describe trainable weights and biases.

Performance metrics

Classification ACC, REC, PRE, and F1 were the performance indicators utilized to assess model's efficacy [22]. The ACC measures how many cases were properly identified out of all the examples in the dataset, which is a measure of the performance of a classification system. The accuracy of the model's false positive and actual positive detection rates is measured by ACC. Examining a model's REC also known as sensitivity or TPR is one approach to evaluate its efficacy. [23]. By merging REC and PRE into one metric, the F-score achieves a happy medium between the two. The four-parameter formulas are provided in Equations (7) through (10) below.

$$Accuracy = \frac{TP+TN}{P+N} \tag{7}$$
$$Precision = \frac{TP}{TP+FP} \tag{8}$$
$$Recall = \frac{TP}{TP+FN} \tag{9}$$
$$F1 = \frac{2 \times precision \times recall}{precision+recall} \tag{10}$$

The loss function measures the model's deviation from its targets during training. The goal of training is to minimize this loss, which is used as a performance metric for the model. Ransomware attack classification is one example of a multi-class classification problem that commonly employs cross-entropy loss.

A data point is considered to be anomalous if it returns a TP. If a data point is truly normal, it is a TN. A FP occurs when a normally occurring data point is mistakenly identified as an abnormality. A data point that is out of the ordinary but mistakenly labeled as normal is called a FN.

Results and Discussion

This section showcases the findings and analysis related to the model's implementation. The Python programming language was utilized in conjunction with the Jupyter Notebook software. For pre-processing data and executing the suggested model, utilized the scikit-learn and Keras packages. The recommended configuration makes use of a 1.6 GHz Intel Core i5 CPU, 8 GB of RAM, and a MacBook Air. Table II shows how well the LSTM model did in detecting suspicious behavior in IoT networks. Overall, the model has a success rate of 98.08% in identifying suspicious network activity, demonstrating its great reliability in distinguishing between regular and aberrant traffic. With an ACC of 90.14%, the model was able to detect the majority of actual data anomalies; a REC of 89.21% confirms that the majority of the cases labeled as anomalies were genuine. F1 further confirms that PRE and REC are balanced at 89.58%, which demonstrates the LSTM model's power and relevance to the IoT environment for early anomaly detection.

Table 2 Performance Results of LSTM Model for Anomaly Detection in IoT Networks

Models	LSTM
Accuracy	98.08
Precision	90.14
Recall	89.21
F1-Score	89.58

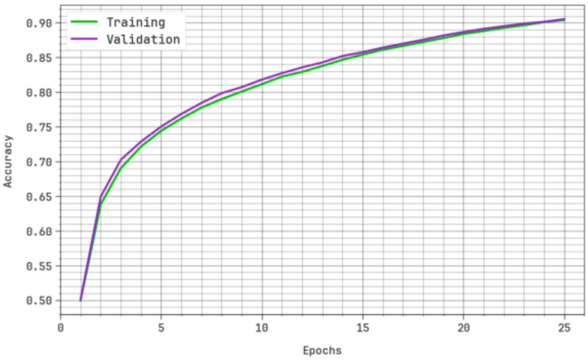

Fig.4 Accuracy curve graph of LSTM Model

The ACC of the Training dataset (green line) and the Validation dataset (purple line) for over 25 epochs are shown in Figure 4 of the LSTM Model's ACC curve graph. The graph indicates that the two accuracies are approximately 0.50 and they increase with increasing the number of epochs, which means that the model is learning properly. The Validation ACC closely tracks the Training ACC throughout the process, hovering just slightly below it initially but then overlapping and continuing to climb together towards the end, reaching a final ACC close to 0.90 at epoch 25.

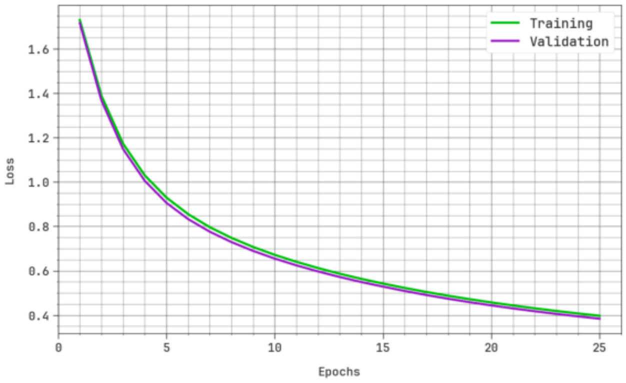

Fig. 5 Loss graph of LSTM model

Figure 5 serves as a reference point for the plot of the LSTM model loss against the epochs on the x-axis, which pertains to the Training dataset (green line) and the Validation dataset (purple line). Loss for both datasets drops sharply in the first few epochs, reaching a peak of about 1.7 before leveling out. Throughout the 25 epochs, the Training and Validation loss curves closely follow each other, suggesting that the model is performing well in terms of generalization and is not overfitting. By epoch 25, both loss values have converged to a low point, approximately 0.4, suggesting the model has successfully minimized the error.

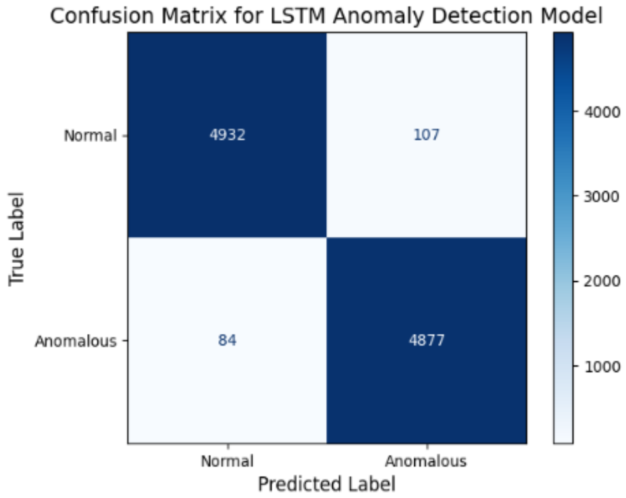

Fig.6 Confusion Matrix of the LSTM Model

Figure 6, An LSTM Model Confusion Matrix. This is how well the LSTM model classified data when it came to identifying IoT anomalies. It accurately classified 4932 Normal and 4877 Anomalous cases with only a few misclassifications of 107 False Positives and 84 False Negatives which reflects good perfection and confidence of distinguishing normal and anomalous network behaviors.

Comparative Analysis

Table III provides a comparative study of different models used in anomaly detection of IoT sensor networks. The LSTM model with an ACC of 98.08% was the most effective model because it could learn the temporal dependency and differentiate between typical and out-of-the-ordinary sensor data. The competitive performance of MLP model was also 97.8 and Voting ensemble model with 96.63 indicates that it performs strongly and just slightly worse in generalization. On the other hand, the FFNN and the Autoencoder models achieved relatively lower accuracies of 69.3 and 68.24, respectively, meaning that they failed to produce complicated temporal and contextual associations that exist in IoT information. Altogether, the LSTM model performed better than any other and, thus, it is the obvious selection to rely upon the unquestionable anomaly detection application within an IoT environment.

Table 3 Performance Comparison of Models of anomaly detection in IoT sensor network

Models	Accuracy
MLP [24]	97.8
Voting [25]	96.63
FFNN [26]	69.3
Autoencoder[27]	68.24
LSTM	98.08

The proposed LSTM model is useful to identify anomalies in an IoT sensor network due to its sensitivity to temporal relations and dynamic tendencies of sensor music in sequence. Gated memory architecture of it allows stable learning and adaptation to time-varying network behavior. This model's great accuracy in detection, low false alarm rate, and ability to generalize to various IoT environments make it ideal for real-time AD and bolstering the security and reliability of networks in general.

Conclusion and Future Study

The environment is being transformed by the IoT due to the exponential growth of the number of small-scale devices that are connected to the Internet. The present paper has constructed a valuable IoT sensor network anomalous detection framework on LSTM model. Kaggle ToN-IoT data was utilized, which was preprocessed, normalized, and balanced with the help of SMOTE to enhance the quality of learning data and models. The LSTM model was identified to perform better and ACC of 98.08, PRE of 90.14, REC of 89.21, and F1 of 89.58 than the traditional ML and neural models. These results confirm the high adaptability of the model to replicate the time-based dependencies of the traffic within the IoT networks, appropriately differentiate the normal and unusual trends, as well as adjust to changing conditions. Its memory-gated architecture enables the ability to handle sequential data, which makes it very reliable and able to produce fewer false alarms, as a result in improved robustness and security of IoT networks.

This study can be extended in the future by using hybrid deep learning neural networks such as CNN-LSTM or Transformer-based networks, real-time detection structures, and federated learning technologies in order to scale, improve privacy, and interpretability in distributed IoT systems.

References

[1] S. Natha, “A Systematic Review of Anomaly Detection using Machine and Deep Learning Techniques,” *Quaid-e-Awam Univ. Res. J. Eng. Sci. Technol.*, vol. 20, no. 1, pp. 83–94, Jun. 2022, doi: 10.52584/QRJ.2001.11.

[2] P. Rana and B. P. Patil, “Cyber security threats in IoT: A review,” *J. High Speed Networks*, vol. 29, no. 2, pp. 105–120, 2023, doi: 10.3233/JHS-222042.

[3] B. R. Cherukuri, “Future of cloud computing: Innovations in multi-cloud and hybrid architectures,” *World J. Adv. Res. Rev.*, vol. 1, no. 1, pp. 068–081, Feb. 2019, doi: 10.30574/wjarr.2019.1.1.0002.

[4] Z. H. Ali, H. A. Ali, and M. M. Badawy, “Internet of Things (IoT): Definitions, Challenges, and Recent Research Directions,” *Int. J. Comput. Appl.*, vol. 128, no. 1, 2015.

[5] M. Yang and J. Zhang, “Data Anomaly Detection in the Internet of Things: A Review of Current Trends and Research Challenges,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 9, pp. 1–10, 2023, doi: 10.14569/IJACSA.2023.0140901.

[6] A. R. Bilipelli, “Forecasting the Evolution of Cyber Attacks in FinTech Using Transformer-Based Time Series Models,” *Int. J. Res. Anal. Rev.*, vol. 10, no. 3, pp. 383–389, 2023.

[7] V. Shah, “Analyzing Traffic Behavior in IoT-Cloud Systems : A Review of Analytical Frameworks,” *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, vol. 9, no. 3, pp. 877–885, 2023.

[8] C. Dietz *et al.*, “IoT-Botnet Detection and Isolation by Access Routers,” in *2018 9th International Conference on the Network of the Future (NOF)*, IEEE, Nov. 2018, pp. 88–95. doi: 10.1109/NOF.2018.8598138.

[9] B. R. Cherukuri, “Ethical AI in cloud: Mitigating risks in machine learning models,” *World J. Adv. Eng. Technol. Sci.*, vol. 1, no. 1, pp. 096–106, Dec. 2020, doi: 10.30574/wjaets.2020.1.1.0018.

[10] V. Shah, “Securing the Cloud of Things : A Comprehensive Analytics of Architecture , Use Cases , and Privacy Risks,” vol. 3, no. 4, pp. 158–165, 2023, doi: 10.56472/25832646/JETA-V3I8P118.

[11] R. Kanwal, R. Kanwal, U. Noor, and Z. Rashid, “A Hybrid Learning Approach for Automatic Data Labelling and Anomaly Detection in IoT Networks,” in *2023 3rd International Conference on Artificial Intelligence (ICAI)*, IEEE, Feb. 2023, pp. 238–241. doi: 10.1109/ICAI58407.2023.10136687.

[12] S. F. Chevtchenko *et al.*, “Anomaly Detection in Industrial Machinery Using IoT Devices and Machine Learning: A Systematic Mapping,” *IEEE Access*, vol. 11, pp. 128288–128305, 2023, doi: 10.1109/ACCESS.2023.3333242.

[13] N. Sarwar, I. S. Bajwa, M. Z. Hussain, M. Ibrahim, and K. Saleem, “IoT Network Anomaly Detection in Smart Homes Using Machine Learning,” *IEEE Access*, vol. 11, pp. 119462–119480, 2023, doi: 10.1109/ACCESS.2023.3325929.

[14] U. Garg, H. Sivaraman, A. Bamola, and P. Kumari, “To Evaluate and Analyze the Performance of Anomaly Detection in Cloud of Things,” in *2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, IEEE, Oct. 2022, pp. 1–7. doi: 10.1109/ICCCNT54827.2022.9984316.

[15] K. Dubey, S. Pandey, and S. Kumar, “A Fault Detection Scheme for IoT-enabled WSNs,” in *2021 International Conference on Computational Performance Evaluation (ComPE)*, IEEE, Dec. 2021, pp. 667–670. doi: 10.1109/ComPE53109.2021.9751912.

[16] N. K. Sahu and I. Mukherjee, “Machine Learning based anomaly detection for IoT Network: (Anomaly detection in IoT Network),” in *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, IEEE, Jun. 2020, pp. 787–794. doi: 10.1109/ICOEI48184.2020.9142921.

[17] V. Upman and N. Goranin, “Investigation of RBFN Application for Anomaly-Based Intrusion Detection on IoT Networks,” in *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*, IEEE, Jul. 2020, pp. 103–109. doi: 10.1109/WorldS450073.2020.9210293.

[18] M. V. Ngo, T. Luo, H. Chaouchi, and T. Q. S. Quek, “Contextual-Bandit Anomaly Detection for IoT Data in Distributed Hierarchical Edge Computing,” in *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*, IEEE, Nov. 2020, pp. 1227–1230. doi: 10.1109/ICDCS47774.2020.00191.

[19] D. Patel, “Zero Trust and DevSecOps in Cloud-Native Environments with Security Frameworks and Best Practices,” *Int. J. Adv. Res. Sci. Commun. Technol.*, vol. 3, no. 3, 2023.

[20] R. Patel, “Automated Threat Detection and Risk Mitigation for ICS (Industrial Control Systems) Employing Deep Learning in Cybersecurity Defence,” *Int. J. Curr. Eng. Technol.*, vol. 13, no. 06, pp. 584–591, Dec. 2023, doi: 10.14741/ijcet/v.13.6.11.

[21] P. Mobtahej, X. Zhang, M. Hamidi, and J. Zhang, “An LSTM-Autoencoder Architecture for Anomaly Detection Applied on Compressors Audio Data,” *Comput. Math. Methods*, vol. 2022, pp. 1–22, Sep. 2022, doi: 10.1155/2022/3622426.

[22] Y. Wei, J. Jang-Jaccard, W. Xu, F. Sabrina, S. Camtepe, and M. Boulic, “LSTM-Autoencoder-Based Anomaly Detection for Indoor Air Quality Time-Series Data,” *IEEE Sens. J.*, vol. 23, no. 4, pp. 3787–3800, Feb. 2023, doi: 10.1109/JSEN.2022.3230361.

[23] R. Patel, “Artificial Intelligence-Powered Optimization of Industrial IoT Networks Using Python-Based Machine Learning,” *J. Eng. Technol. Adv.*, vol. 3, no. 4, pp. 138–148, 2023, doi: 10.56472/25832646/JETA-V3I8P116.

[24] T. M. Booij, I. Chiscop, E. Meeuwissen, N. Moustafa, and F. T. H. D. Hartog, “ToN_IoT: The Role of Heterogeneity and the Need for Standardization of Features and Attack Types in IoT Network Intrusion Data Sets,” *IEEE Internet Things J.*, vol. 9, no. 1, pp. 485–496, 2022, doi: 10.1109/JIOT.2021.3085194.

[25] Y. Alotaibi and M. Ilyas, “Ensemble-Learning Framework for Intrusion Detection to Enhance Internet of Things’ Devices Security,” *Sensors*, vol. 23, no. 12, 2023, doi: 10.3390/s23125568.

[26] F. Anwar and S. Saravanan, “Comparison of Artificial Intelligence Algorithms for IoT Botnet Detection on Apache Spark Platform,” *Procedia Comput. Sci.*, vol. 215, pp. 499–508, 2022, doi: 10.1016/j.procs.2022.12.052.

[27] H. Nizam, S. Zafar, Z. Lv, F. Wang, and X. Hu, “Real-Time Deep Anomaly Detection Framework for Multivariate Time-Series Data in Industrial IoT,” *IEEE Sens. J.*, vol. 22, no. 23, pp. 22836–22849, 2022, doi: 10.1109/JSEN.2022.3211874.