



ISSN(Online): 2320-9801

ISSN (Print): 2320-9798

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 10, October 2015

## Distributed DNN-based Middleware for Cyberattack Detection in the Smart IOT Ecosystem: A Novel Framework and Performance Evaluation Technique

Sakshi Sharma<sup>1\*</sup>, Natasha Dutta<sup>2</sup>

Salesforce Administrator Jul, KONI Hospitality, Chula Vista, CA<sup>1\*</sup>

Project Engineer, Online Micro Services, India<sup>2</sup>

**ABSTRACT:** In the current digital environment, cyberattacks continue to pose a serious risk and difficulty. Devices connected to the Internet of Things (IoT) are becoming more vulnerable due to security problems like ransomware, malware, insufficient encryption, and IoT botnets. These flaws may result in ransom demands, data manipulation, unauthorised access, and system threats. Creating strong cybersecurity protocols for contemporary smart environments is necessary to address these problems. This method uses proactive network traffic monitoring to find possible risks in the Internet of Things environment. Our plan is to increase smart environments' awareness of threats and security. Two IoT gateways were used to examine the effectiveness and performance of a deep neural network (DNN) model. The results were promising: the model caused an average increase of less than 30 kb/s in network bandwidth and a mere 2% rise in CPU usage. Additionally, memory and power consumption were minimal, with 0.42 GB and 0.2 GB of memory usage for NVIDIA Jetson and Raspberry Pi devices, respectively, and an average 13.5% increase in power consumption per device. The machine learning models achieved nearly 93% detection accuracy and a 92% F1 score on the datasets used. Our framework demonstrates an effective and efficient method for detecting malware and attacks in Smart Environments.

**KEYWORDS:** IoT, Ecosystem, Network, Cyber Attacks, Machine Learning, Malware and Attacks, IoT security, Artificial Neural Network.

### I. INTRODUCTION

In the rapidly evolving landscape of the Internet of Things (IoT), the proliferation of smart devices and interconnected systems has significantly amplified the complexity of cybersecurity challenges. The sheer volume and diversity of IoT devices introduce new vulnerabilities that traditional security measures often fail to address effectively. These vulnerabilities, including inadequate encryption, susceptibility to malware, ransomware, and the formation of IoT botnets, create opportunities for cybercriminals to exploit and compromise critical systems. As these devices increasingly play pivotal roles in various applications, from smart homes and industrial automation to healthcare and urban infrastructure, the need for advanced, proactive security solutions becomes ever more pressing [1]. In response to these challenges, our study introduces a novel framework that leverages distributed deep neural networks (DNNs) as a middleware solution for detecting cyber-attacks within the Smart IoT ecosystem. This innovative approach employs AI-driven techniques to enhance the detection and prevention of threats across a broad range of scenarios, ensuring robust protection for interconnected devices. By implementing this framework, we aim to proactively monitor and analyze network traffic, identifying potential security breaches with high accuracy and efficiency. Our approach not only addresses the current limitations of conventional cybersecurity practices but also adapts to the dynamic and distributed nature of IoT environments. Through rigorous performance evaluation and testing, including deployment on various IoT gateways, we assess the framework's effectiveness, focusing on key metrics such as detection accuracy, resource consumption, and system impact. The results demonstrate the framework's capability to significantly improve cybersecurity in Smart Environments, providing a critical safeguard against evolving threats and ensuring the integrity and resilience of interconnected systems.



# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 10, October 2015

Smart Environments that incorporate IoT infrastructure face a range of cybersecurity challenges. One major issue is the difficulty of implementing traditional endpoint protection measures, such as antivirus software, intrusion detection systems, and firewalls, in IoT environments where devices are often resource-constrained and require energy-efficient solutions. The AT&T Alien Labs™ recently discovered Botena Go malware that exposed millions of IoT devices [2]. A group of hackers accessed and controlled thousands of Verkada security cameras and exposed user credentials publicly on the internet [3]. IoT malware attacks in particular increased by 6%, with routers being the most targeted devices [4]. Additionally, the need for real-time communication is hindered by the asynchronous nature of many IoT systems, and the vast diversity of IoT devices makes it challenging to apply a one-size-fits-all security solution. Existing studies have explored the use of AI models for cybersecurity, but many have focused on limited datasets or specific types of attacks. To address these gaps, our study proposes a novel framework that leverages AI to detect malware attacks across a wide range of IoT devices in Smart Environments. Our approach employs a multi-agent network of AI models, where the most computationally intensive models are trained in the Cloud, while less demanding models are trained in Fog/Dew environments and deployed on Edge devices. This setup ensures efficient use of resources and effective threat detection. Key contributions of our research include: (a) the development of a new method for identifying malware and attacks on IoT devices using AI, (b) the ability to monitor live network traffic for real-time threat detection, (c) the capability to pinpoint security issues and affected devices, which helps minimize maintenance efforts, and (d) performance and concurrency testing that confirms the framework's practicality for real-world deployment in Smart Environments. Several existing studies [5–8] have proposed AI models for cybersecurity; however, the majority of them have considered only a portion of the dataset or targeted only a few attacks. Therefore, in this study, we have proposed an approach with a framework to discover malware attacks on IoT devices using AI-enabled approaches covering diverse and distributed scenarios in Smart Environments. In our work, the choice of hardware for setting up the IoT network is representative of typical industrial use and is available off the shelf. Our approach will utilize a multi-agent network of AI models, where the most cumbersome will be trained in the Cloud environment, and the rest can be trained in Fog/Dew and subsequently deployed on Edge devices. The findings suggest that our approach is well-suited for efficient, in-production implementation, providing robust cybersecurity for diverse and distributed IoT ecosystems.

As the Internet of Things (IoT) network systems continue to expand and become more complex, the integration of machine learning with IoT has become increasingly prevalent. The shift towards data-driven infrastructure has driven research to focus more on machine learning applications within the IoT domain. Today, machine learning techniques are applied across various fields, from healthcare—where they assist in interpreting ECGs, detecting diseases through X-rays, analyzing genomic patterns, automating cancer detection, and modeling brain signals—to aerospace, where they help in defect detection through complex methods like eddy current testing, as demonstrated by D'Angelo et al. The growing complexity of IoT systems, however, has introduced significant vulnerabilities. Security breaches and anomalies in IoT devices have become common, highlighting the need for enhanced security measures and robust detection mechanisms to safeguard these increasingly intricate networks.

## II. LITERATURE REVIEW

In this article, we introduce an innovative AI-based middleware and model designed to detect attacks in diverse Smart Environments. Our approach involves a four-step process to enhance malware and attack detection through data-driven, multi-agent systems. The first step involves collecting network traffic data from various IoT devices, such as Arduino, Raspberry Pi, and NVIDIA Jetson devices. In the second phase, we apply various AI and deep learning models to classify malware and attacks at multiple levels, focusing on minimizing overhead and latency in the IoT components. The third step involves deploying these AI models in different configurations within our local smart environments. Finally, the fourth step evaluates the performance and concurrency of the models, measuring their impact on electrical power, network bandwidth, and memory usage to assess their efficiency in enhancing IoT cybersecurity. Our experimental results, derived from diverse IoT malware and attack datasets, demonstrate that AI can be a highly effective tool for safeguarding smart environments against cybersecurity threats [9].



ISSN(Online): 2320-9801

ISSN (Print): 2320-9798

# International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 10, October 2015**

The challenge of detecting sophisticated and persistent botnet attacks on connected IIoT systems is particularly complex and critical, given the potentially catastrophic consequences. To address this challenge, our research paper introduces AttackNet, a powerful deep learning model designed to detect and classify various botnet attacks in IIoT environments. This model utilizes an adaptive CNN-GRU architecture and has been rigorously tested with recent datasets and standard performance metrics. AttackNet demonstrates exceptional performance, achieving a testing accuracy of 99.75%, a loss of 0.0063, and precision and recall scores of 99.75% and 99.74%, respectively. Notably, our model excels within the N\_BaIoT dataset, achieving an impressive 99.75% accuracy across ten classes, which significantly outperforms existing techniques by 3.2% to 16.07%. AttackNet surpasses current anomaly detection systems in IIoT in terms of accurately identifying and classifying botnet attacks, especially when evaluated on real-time IoT device datasets [10].

Smart Environment is a technology-enabled circumstance that offers better, userfriendly and efficient IoT infrastructure with a focus on greener and more sustainable future [11]. Used devices, components, and generated data are subject to the user's needs with sustainability and adaptation as major targets [12,13]. To defend the IoT infrastructure against known cyber-attacks, various open-source and commercial software solutions, such as anti-viruses, firewalls, anti-pattern detection approaches, and security protocols, help to enhance cybersecurity.

Our proposed model is based on a denoising autoencoder, which helps extract features resilient to the heterogeneous nature of IoT environments. Experimental results using real-world IoT datasets demonstrate that our framework significantly enhances the accuracy of detecting malicious data compared to existing IoT-based anomaly detection models. Internet of Things (IoT) systems have become integral to various industries and government services. However, these systems are highly vulnerable to security attacks that compromise data integrity and service availability. The diversity of data from different IoT devices and the disturbances within these systems make it more difficult to detect anomalies and compromised nodes compared to traditional IT networks. Consequently, there is an urgent need for effective and reliable anomaly detection to ensure that malicious data is identified and excluded from IoT-driven decision support systems. The demand for internet data traffic is rapidly increasing for different data-driven Smart Environment applications. The network traffic predictions focus on anticipating future traffic, utilizing previous traffic data [14]. Using IoT malware network traffic data, Bendiab et al. [15] proposed an AI-enabled detection approach at the package level, reducing the time of detection using deep learning methods. Their network data consist of 1000 pcap files of normal and malware traffic collected from different network traffic sources.

Types of attacks and anomalies such as Denial of Service, Data Type Probing, Malicious Control, Malicious Operation, Scanning, Spying, and Incorrect Configuration can lead to significant failures in IoT systems. This paper evaluates and compares the performance of several machine learning models for accurately detecting these attacks and anomalies in IoT systems. The machine learning algorithms considered include Logistic Regression (LR), Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), and Artificial Neural Network (ANN). The models are assessed using various performance metrics, including accuracy, precision, recall, F1 score, and the area under the Receiver Operating Characteristic Curve. The results show that Decision Tree, Random Forest, and ANN models each achieved a test accuracy of 99.4%. While these models have the same accuracy, the Random Forest model demonstrated superior performance based on other evaluation metrics.

A general IoT ecosystem normally includes IoT nodes, end-point devices with limited computational capabilities (CPU MHz) that are used to collect data, send measurements and often work using batteries or solar panels. The IoT gateways are portable devices, having the functionality of low-end personal computers (CPU GHz), performing data processing and aggregation tasks. Moreover, the devices follow different proprietary and open communication protocols, unique data storage standards, operational logic [9], different operating systems, and dependencies. From the cybersecurity perspective, data can be protected on the Linux-based IoT gateway using tools available for Unix such as ClamAV (Clam AntiVirus) for malicious software detection, encryption available for Linux and RPiDS (Raspberry Pi IDS) [16] for an intrusion detection system (IDS). However, the application of such measures on IoT end-nodes is extremely limited. There is no OS, yet rather firmware that defines a strict routine of initialization function SETUP() and the iterative function LOOP() [17]. The only cybersecurity solution that is available and being tested for AVR is the



# International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 10, October 2015**

Arduino Crypto library, designed to protect the information by application of various standard encryption methods [18]. Therefore, it is necessary to establish an understanding of what kind of data analytics for security can be run on IoT nodes and what should be moved to the IoT gateway for the sake of ensuring primary services availability and data protection [19]. Our previous work provides a framework for the data-driven cyberattack prediction method using several intelligent methods. The method analyzes the complexity of power consumption and bandwidth in deploying AI models to IoT devices.

### III. METHODOLOGY

The methodology consists of several key stages designed to ensure comprehensive and effective threat detection. Initially, we set up a diverse array of IoT devices, including sensors, actuators, and gateways, across a simulated Smart Environment to generate a wide range of network traffic data. This data is crucial for training and evaluating our deep learning models. The core of our framework involves a multi-agent system where different components of the deep neural network (DNN) are trained and deployed across various network layers. We use a distributed approach, with the most computationally intensive models being trained in a cloud environment, while lighter models are trained in Fog/Dew environments and deployed on Edge devices. This distribution leverages the strengths of each environment—cloud for heavy processing, Fog/Dew for intermediate tasks, and Edge devices for real-time, localized processing. Our methodology begins with data aggregation, where we collect multi-level network traffic data from IoT devices across different scenarios, including normal operations and various attack vectors. The collected data undergoes preprocessing to handle issues like noise and missing values, ensuring that the dataset is clean and representative. Next, we apply advanced deep learning techniques to train the DNN models. This includes using convolutional neural networks (CNNs) for feature extraction and recurrent neural networks (RNNs), specifically gated recurrent units (GRUs), for capturing temporal patterns in the data. The trained models are then deployed in a distributed manner: the cloud environment handles the bulk of model training and updates, while the Fog/Dew environments and Edge devices use these models to perform real-time monitoring and attack detection. To validate the effectiveness of our framework, we perform extensive performance and concurrency testing. This involves evaluating metrics such as detection accuracy, false positives, false negatives, and system resource usage, including network bandwidth, CPU and memory consumption, and power usage. We also test the framework's scalability and robustness under different network loads and attack scenarios to ensure it can handle real-world conditions. The final step includes deploying the models in operational environments to observe their performance in live settings, making adjustments as necessary to improve accuracy and efficiency.

The overall framework consists of several distinct processes, as illustrated in Fig. 1. The first step involves dataset collection and observation, where the dataset is carefully gathered and examined to identify the types of data present. Following this, data preprocessing is carried out, which includes several key stages: data cleaning, data visualization, feature engineering, and vectorization. These preprocessing steps transform the raw data into feature vectors.

The dataset is divided into training and testing sets using an 80–20 split. The training set is utilized to train various machine learning algorithms, while the testing set is reserved for evaluating the final model. The training process involves different optimization techniques depending on the classifier used. For instance, Logistic Regression employs coordinate descent, while Support Vector Machine (SVM) and Artificial Neural Networks (ANN) use conventional gradient descent methods. Decision Trees (DT) and Random Forest (RF) do not require an optimizer as they are non-parametric models. Once the models are trained, the final model is assessed using the testing set and evaluated with various performance metrics to determine its effectiveness.



# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 10, October 2015

Table 1: Frequency distribution of considered attacks.

Attacks	Frequency Count	% of Total Data	% of Anomalous Data
Denial of Service	5780	01.61%	57.70%
Data Type Probing	342	00.09%	03.41%
Malicious Control	889	00.24%	08.87%
Malicious Operation	805	00.22%	08.03%
Scan	1547	00.43%	15.44%
Spying	532	00.14%	05.31%
Wrong Setup	122	00.03%	01.21%

## IV. RESULTS

In the Data Analysis section, various machine learning techniques were applied to the dataset, and five-fold cross-validation was performed with each technique. Figures 1(a) and (b) illustrate how the accuracy results stabilized after this cross-validation. The findings indicate that Random Forest (RF) and Artificial Neural Network (ANN) achieved the highest accuracy for both training and testing. Decision Tree (DT) showed performance similar to RF and ANN during training but exhibited greater variability during testing, initially performing poorly before aligning closely with RF and ANN in the later folds. Support Vector Machine (SVM) and Logistic Regression (LR) underperformed compared to other techniques in training. However, in the first two folds of testing, SVM and LR initially outperformed other methods, with Logistic Regression performing the best among them. Nonetheless, their performance declined in the final three folds. It provides various evaluation metrics for each technique trained on the dataset. It reveals that DT and RF outperformed the other techniques in terms of accuracy, precision, recall, and F1 score, with RF being slightly more accurate than ANN. While LR and SVM also performed reasonably well, they did not match the performance of DT, RF, or ANN.

The confusion matrices help determine the most optimized technique. The results indicate that RF is the most effective method, correctly classifying nearly all classes except Denial of Service (DoS) and Normality. RF misclassified 403 out of 1178 DoS samples as Normal and 18 out of 69,571 Normal samples as DoS. DT's confusion matrix is similar to RF's but also misclassified 18 Normal samples as DoS and two as Spying. ANN's performance was comparable to DT, misclassifying one more sample than DT. ANN correctly predicted six out of eight labels but misclassified 403 DoS samples as Normal and 18 Normal samples as DoS, with additional misclassifications in Spying and Malicious Control. LR and SVM performed poorly overall. LR misclassified numerous samples across various categories, including all remaining DoS samples as Normal. SVM also struggled, misclassifying data from several categories as Normal, with notable misclassifications in DoS and other classes.

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 10, October 2015

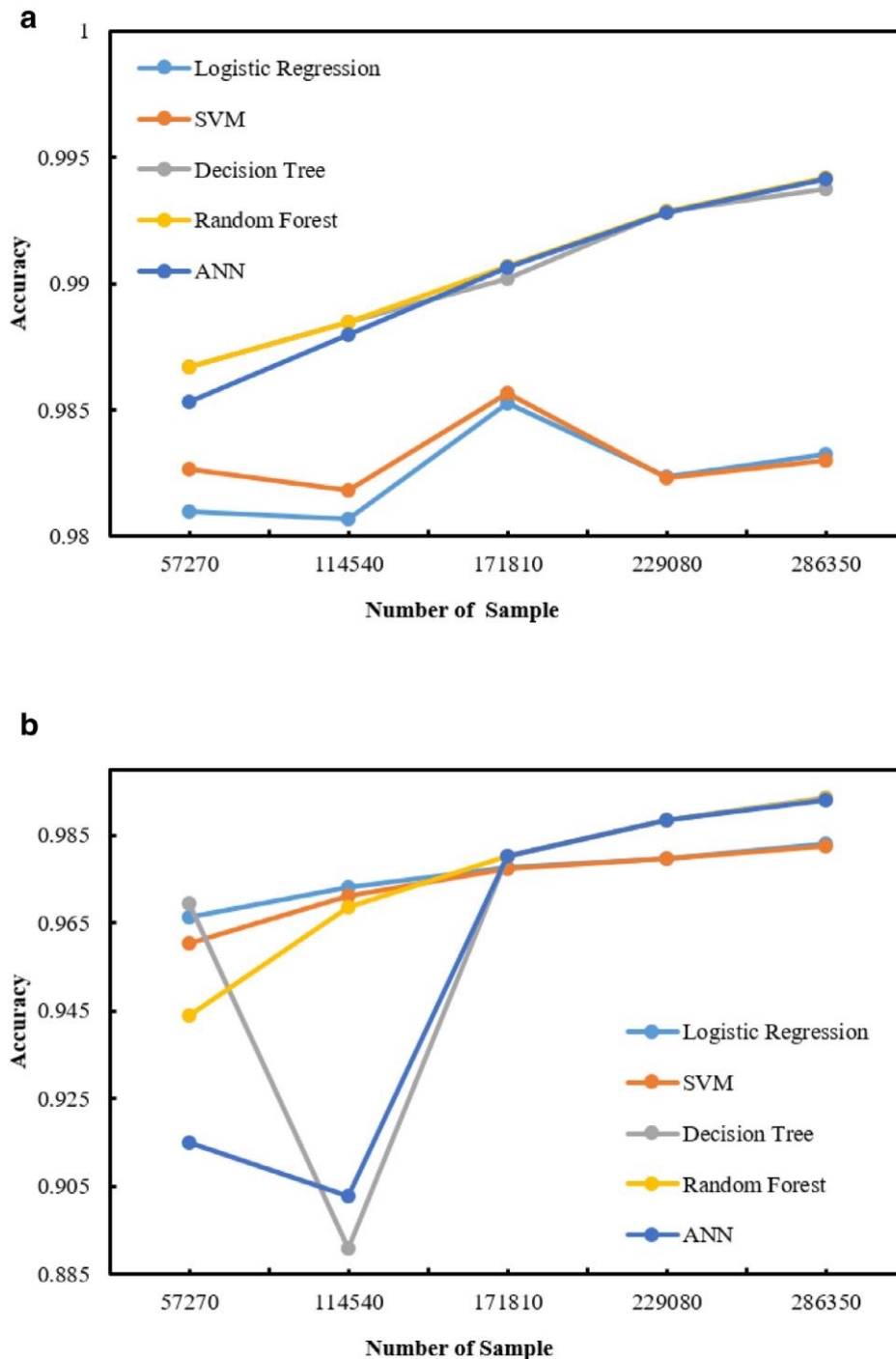


Fig. 1. (a) Training accuracy for different techniques for 5 fold cross validation (b) Testing accuracy for different techniques for 5 fold cross validation.

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 10, October 2015

Finally, Fig. 2 displays the Receiver Operating Characteristic (ROC) Curves for LR, SVM, DT, RF, and ANN. The area under the ROC curve for DT, RF, and ANN approaches one, indicating high accuracy. In contrast, LR and SVM only achieved a value of one for DoS and Wrong Setup categories.

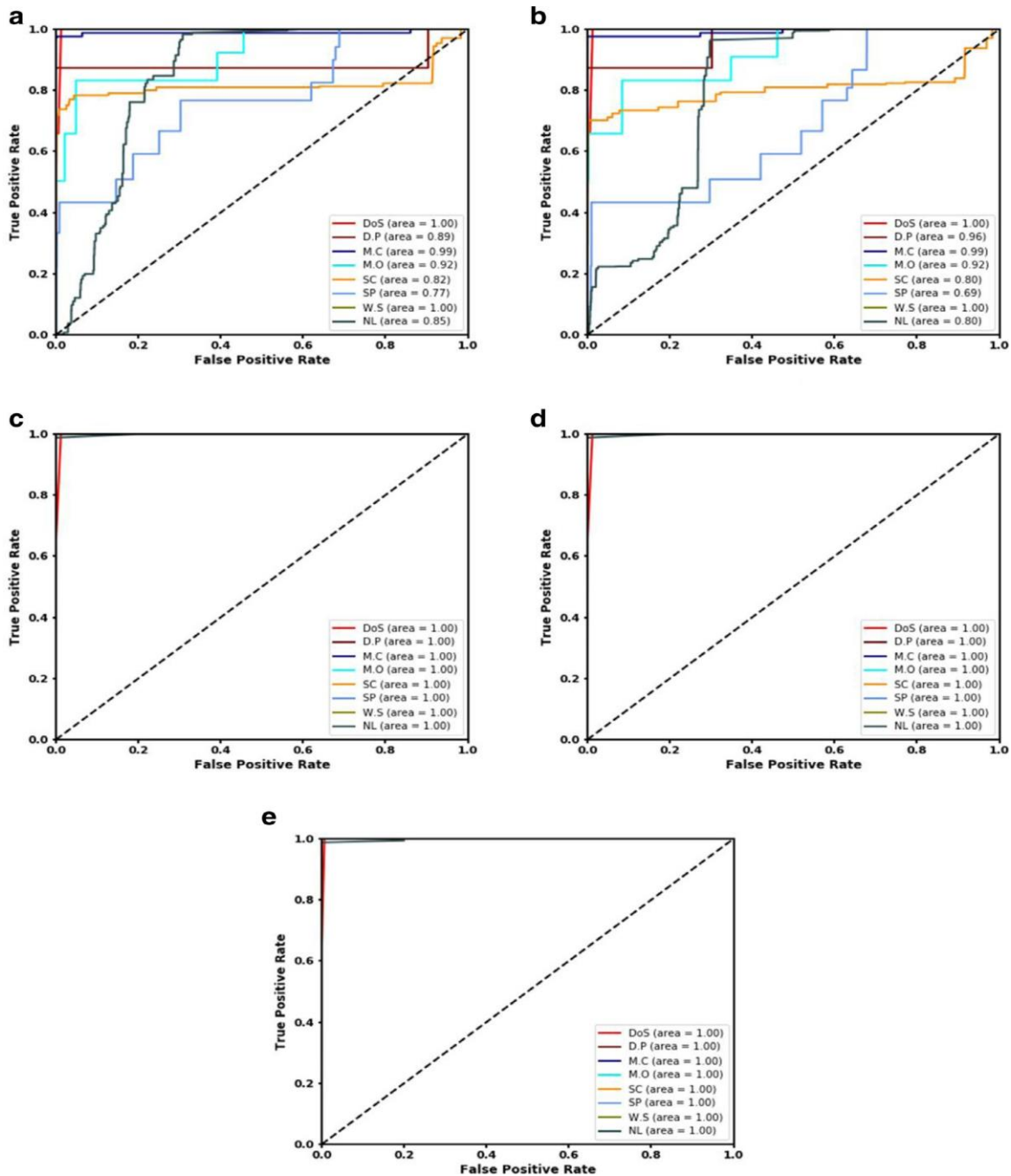


Fig. 2. ROC Curve of (a) Logistic Regression (b) Support Vector Machine (c) Decision Tree (d) Random Forest (e) Artificial Neural Network.



# International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 10, October 2015**

## V. DISCUSSION

The results highlight significant differences in performance among various machine-learning techniques for detecting attacks in IoT systems. The use of five-fold cross-validation provided a robust evaluation of the models, revealing that Random Forest (RF) and Artificial Neural Networks (ANN) consistently performed well in both training and testing phases. RF and ANN achieved high accuracy, precision, recall, and F1 scores, demonstrating their effectiveness in identifying malicious activities and anomalies within IoT environments.

Decision Tree (DT) showed comparable performance to RF and ANN during training, but its performance fluctuated during testing. Initially, DT struggled but eventually aligned closely with RF and ANN in the latter folds of cross-validation. This variability suggests that while DT can be effective, its performance may be more sensitive to the specific characteristics of the dataset and the validation process.

Support Vector Machine (SVM) and Logistic Regression (LR) generally lagged behind the other techniques. SVM and LR exhibited lower accuracy and other metrics in training, although LR showed better performance than SVM in some testing folds. Despite these occasional advantages, both methods ultimately fell short of the performance levels achieved by RF and ANN. The confusion matrices further highlight that LR and SVM had significant difficulties in accurately classifying various attack types, with numerous misclassifications across different categories. The confusion matrices also underscored the strengths and limitations of each model. RF was particularly effective, with only a few misclassifications in the DoS and Normality classes. ANN's performance was close to that of RF but showed slightly more misclassifications, particularly in the Normal and DoS categories. DT's performance was similar to RF's but exhibited some inconsistencies in classifying Normal samples. In contrast, LR and SVM had numerous misclassifications, reflecting their lower effectiveness in accurately detecting and categorizing attacks.

The ROC curves further corroborated these findings. RF, DT, and ANN achieved high areas under the curve, indicating strong performance in distinguishing between classes. However, LR and SVM only achieved optimal performance in a limited number of categories, reflecting their overall lower effectiveness in this context. The results suggest that RF and ANN are the most suitable techniques for detecting attacks in IoT systems due to their high accuracy and robust performance across various metrics. DT also shows promise but with some variability, while LR and SVM are less effective, particularly in accurately classifying different types of attacks. These findings provide valuable insights into selecting and optimizing machine learning models for cybersecurity applications in IoT environments.

## VI. CONCLUSION

The study concluded that the Random Forest (RF) technique is particularly effective for detecting cyberattacks in IoT networks when using the dataset analyzed. RF demonstrated superior performance in accurately predicting several types of attacks, including Data Probing (D.P), Malicious Control (M.C), Malicious Operation (M.O), Scanning (SC), Spying (SP), and Wrong Setup (W.S), outperforming other machine learning methods. It also showed better accuracy in predicting Denial of Service (DoS) and Normal samples compared to other techniques. Based on these results, RF is deemed the most suitable technique for this specific dataset and problem. However, it's important to note that the study only applied traditional machine learning methods and did not introduce any new algorithms. Consequently, further research is necessary to develop a more robust detection algorithm and to thoroughly analyze the entire framework. Additionally, the study utilized data from a virtual environment, which may not fully capture the complexities of real-time scenarios. In real-world applications, different issues may arise, such as variations in the behavior of microservices over time, leading to anomalies in IoT services. To address these concerns, future research should focus on empirical studies using real-time data and explore how different factors might affect the performance of RF and other techniques. Although RF achieved an accuracy of 99.4% in this study, its effectiveness in handling larger datasets and addressing unforeseen challenges remains uncertain, highlighting the need for continued investigation.





ISSN(Online): 2320-9801

ISSN (Print): 2320-9798

# International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 10, October 2015**

## REFERENCES

1. Augusto, J.C. Past, Present and Future of Ambient Intelligence and Smart Environments. In Proceedings of the Agents and Artificial Intelligence; Filipe, J., Fred, A., Sharp, B., Eds.; Springer: Berlin/Heidelberg, Germany, 2010; pp. 3–15. [CrossRef]
2. Augusto, J.C.; Nakashima, H.; Aghajan, H. Ambient Intelligence and Smart Environments: A State of the Art. In Handbook of Ambient Intelligence and Smart Environments; Nakashima, H., Aghajan, H., Augusto, J.C., Eds.; Springer: New York, NY, USA, 2010; pp. 3–31. [CrossRef]
3. R. Phalnikar, P.A. Khutade, Survey of QoS based web service discovery, in: 2012 World Congress on Information and Communication Technologies, IEEE, 2012, pp. 657–661.
4. C. Pautasso, E. Wilde, RESTful web services: principles, patterns, emerging technologies, in: Proceedings of the 19th International Conference on World Wide Web, 2010, pp. 1359–1360.
5. W. Rong, K. Liu, A survey of context aware web service discovery: from user's perspective, in: 2010 Fifth Ieee International Symposium on Service Oriented System Engineering, IEEE, 2010, pp. 15–22.
6. V.X. Tran, H. Tsuji, A survey and analysis on semantics in QoS for web services, in: 2009 International Conference on Advanced Information Networking and Applications, IEEE, 2009, pp. 379–385.
7. Asuvaran & S. Senthilkumar, "Low delay error correction codes to correct stuck-at defects and soft errors", 2014 International Conference on Advances in Engineering and Technology (ICAET), 02-03 May 2014. doi:10.1109/icaet.2014.7105257.
8. Aziz A., Hanafi S., and Hassanien A., "Multi-Agent Artificial Immune System for Network Intrusion Detection and Classification," in Proceedings of International Joint Conference SOCO'14-CISIS'14-ICEUTE'14, Bilbao, pp. 145-154, 2014.
9. B. Kitchenham, P. Brereton, M. Turner, M. Niazi, S. Linkman, R. Pretorius, D. Budgen, The impact of limited search procedures for systematic literature reviews—A participant-observer case study, in: 2009 3rd International Symposium on Empirical Software Engineering and Measurement, IEEE, 2009, pp. 336–345.
10. Senthilkumar Selvaraj, "Semi-Analytical Solution for Soliton Propagation In Colloidal Suspension", International Journal of Engineering and Technology, vol, 5, no. 2, pp. 1268-1271, Apr-May 2013.
11. J. Kopecky, T. Vitvar, C. Bournez, J. Farrell, Sawsdl: Semantic annotations for wsdl and xml schema, IEEE Internet Comput. 11 (6) (2007) 60–67.
12. A. Renuka Devi, S. Senthilkumar, L. Ramachandran, "Circularly Polarized Dualband Switched-Beam Antenna Array for GNSS" International Journal of Advanced Engineering Research and Science, vol. 2, no. 1, pp. 6-9; 2015.
13. M. Malaimalavathani, R. Gowri, A survey on semantic web service discovery, in: 2013 International Conference on Information Communication and Embedded Systems, ICICES, IEEE, 2013, pp. 222–225.
14. Aziz A., Salama M., Hassanien A., and Hanafi S., "Detectors Generation Using Genetic Algorithm for A Negative Selection Inspired Anomaly Network Intrusion Detection System," in Proceedings of Federated Conference on Ensemble Voting based Intrusion Detection Technique using Negative Selection Algorithm 157 Computer Science and Information Systems, Wroclaw, pp. 597-602, 2012.