Internet of Things (IoT) (CSC-544-UT1)

Project Title: Anomaly Detection in IoT Networks

Team Members

Lalitha Priya Bijja – 101168225
Venkata Surya Deepak Lakshimpalli – 101143451

Abstract—As smart devices and the Internet develop, the Internet of Things (IoT) technologies have become an important factor in our life. IoT helps manufacturing companies to monitor the status of every machine in real time, the quality of products, and the environment variables within the factory. This not only allows managers to reduce the risk of damages and losses but also helps to make decisions from a higher overall standpoint. In addition, IoT has changed people's lives and behavior. People are now relying on IoT devices and services more than ever. However, anomalies can cause security and safety issues for an IoT network. It is important to detect anomalies and alert users to prevent damages or losses. In this paper, we propose using Machine Learning and Deep Learning methods to detect anomalies in a network. The experiments were performed on the IoT-23 dataset. The performance and time cost for these models are compared to give us the best algorithm with high performance in less time.

Index Terms—Internet of Things, security, malicious node, anomaly detection, Machine Learning, Deep Learning.

I. INTRODUCTION

Internet of Things (IoT) is a revolution to the global information industry after the Internet. The IoT is a smart network that allows devices to exchange information and communicate with each other through the internet. With IoT, humans can achieve the purpose of tracking, monitoring, locating, identifying, and managing things. Since the revolution of the Internet and mobile devices, IoT has become an evolving and hot research topic within the computer science industry. The number of IoT devices on the Internet is increasing every year and in every sector such as: Smart Healthcare, Smart Transportation, Smart Governance, Smart Agriculture, Smart Grid, Smart Home, Smart Supply chain, etc.

Because of the convenience brought by IoT, the behavior of humans has also changed. People of younger generations are more used to using services from IoT devices such as smart bulbs, smart ovens, smart refrigerators, AC, temperature sensors, smoke detectors, etc. However, as IoT develops, the concerns of privacy and security issues have increased among users. As all the devices are connected to the internet and each other, this leads to more number of ways for the attacker to access the information possible. The connected devices collect data with personal information and store it. Most of the users do not have knowledge about IoT technology, and the hackers can steal information from the users or even control the smart devices of the users. This not only reduces the advancement of IoT technology but also slows down the development of IoT infrastructure. Therefore, providing security and privacy for

these constantly and heavily connected devices has become a major challenge. Another key issue for providing security and privacy to these devices is managing the huge amount of data generated by them, which is quite difficult using general data collection, storage, and processing techniques.

With the development of Machine Learning (ML) and Deep Learning (DL), learning algorithms can learn from the results of trained data and adapt to increase performance to make informed and intelligent decisions. A learning algorithm that has been trained by the data can establish the difference between regular benign traffic in the model with malicious traffic. In other words, it can detect when there is abnormal behavior in the network, thereby preventing unauthorized access. Learning algorithms are basically classified into two categories, which are Supervised Learning and Unsupervised Learning. We try to use the lightweight machine learning methods and neural networks for accuracy improvement on detecting malicious nodes. The central unit in the model captures IoT traffic data and sends the data to a selected trained Machine Learning or Deep Learning model. Multiple trained Machine Learning and Deep Learning models are tested. The reason for choosing multiple models is to fit the individual needs for different users or groups. In other words, it is important to find the efficient model for a different type of user.

This large data in the IoT network and the heterogeneity of the data make it too difficult to improve security and to meet all the requirements such as cost-effectiveness, reliability, performance, etc. In some cases, if one of the features is improved then it may affect the performance of other features. For example, an increase in the number of security checks and protocols in all data transfer then it may result in the increase in cost and latency of that particular application making it unsuitable for certain users. Also, the increase in the number of devices connected increases the chance for an attacker to gain access to the network by accessing the node or device that has a weak link, for example, a device like a smart bulb. Most of the devices that are available in the market as of now do not have the security features like firewalls, antivirus, etc. As the IoT devices are resource-constrained it is important for these devices to detect an intrusion with less complexity and time. So, the use of Machine learning (ML) and Deep Learning (DL) techniques helps to reduce this complexity as these models learn from the trained data. It is important for the central unit to classify the message's integrity. The privacy and security issues of IoT motivate researchers for developing a framework for automatic IoT sensors attack and anomaly detection.

In this project, we proposed to use ML/DL algorithms such as Support Vector Machines, Decision Trees, Naive Bayes, and Convolutional Neural Networks for anomaly detection and based on their accuracy and time cost, the better algorithm to use can be concluded. And we used the IoT-23 dataset for the implementation of ML/DL methods. The paper goes as follows, in Section II literature review is discussed, in Section III methodology is explained, in Section IV results are discussed with evaluation metrics and comparison. In sections V, we concluded the paper with a few suggestions of future work. At last, references for this study are included.

II. LITERATURE REVIEW

In this section, all the different anomaly detection algorithms and methodologies are briefly discussed. There are a number of different mechanisms to improve the safety and privacy of IoT devices. For example, in [12], a chaos-based encryption technique is used to generate symmetric keys to provide secured data transmission between the server and the IoT device, which guarantees data integrity and authenticity. According to [13], a mechanism with low computational complexity has been proposed by using random hopping sequence and random permutations to hide valuable information. Moreover, in [14], Doshi presented a method to detect DDoS attacks in the network layer with a low-cost machine learning approach, including KNN, LSVM, NN, Decision Tree, and Random Forest. This method can detect which node is attacking the central unit with an IP address. This method was reported to achieve high testing accuracy for all five machine learning algorithms. In [21], the detection of anomaly is done using fog computing, which clusters the different types of anomalies present in the sensor layer or edge nodes without performing computation on both the cloud and sensor layer but in the fog layer of the network. By using the fog computing method it has become easier to detect an anomaly. In [17], the author tries to implement a malware detection system by using different classifiers of k-NN and random forest to build the model. The device filters TCP packets and selects important features such as frame numbers, length, labels, etc. The k-NN algorithm assigns traffic to the class while the random forest classifier builds decision trees to detect the malware. The authors have proposed a new methodology in [22] which uses game theory and Nash equilibrium to help the resource-constrained IoT devices to detect an anomaly using Intrusion Detection System(IDS), activating it only when needed. When an attack occurs the attack pattern (signature) is stored and then the model is trained and whenever the pattern repeats it is identified as an attack.

In another approach [15], the authors used the Gated Recurrent Unit (GRU) deep learning model to detect different types of network attacks on an IoT network. The model uses an attention mechanism that helps to learn significant features from the input data. The GRU model performed better compared to the traditional machine learning models. Also, in [16], the authors proposed a deep learning approach for intrusion detection system for the IoT network using Convolutional Neural Networks (CNNs). The CNN model is trained and tested on the CICIDS2017 dataset. The model is able to detect attacks in the IoT network with high accuracy. In [18], the authors have proposed an approach to use the Ensemble Learning and Multi-Classifier system to detect IoT network attacks. The authors have selected six machine learning classifiers such as J48, RandomForest, KNN, NaiveBayes, Multilayer Perceptron, and SMO, to build an ensemble learning system. The experimental results show that the ensemble model performs better than the individual classifiers. In [19], the authors proposed a novel intrusion detection system (IDS) for the IoT network using an unsupervised learning technique called autoencoder. The autoencoder model is trained and tested on the UNSW-NB15 dataset. The experimental results show that the proposed IDS system is able to detect known and unknown attacks in the IoT network.

In [20], the authors have proposed a deep learning-based IDS for the IoT network using a Long Short Term Memory (LSTM) model. The model is trained and tested on the CICIDS2017 dataset. The LSTM model is able to capture the temporal dependencies of the input data and detect attacks with high accuracy. In [23], the authors have proposed a deep learning approach for intrusion detection in the IoT network using a Recurrent Neural Network (RNN). The RNN model is trained and tested on the UNSW-NB15 dataset. The experimental results show that the proposed RNN model outperforms the traditional machine learning models. In [24], the authors have proposed an anomaly detection system for the IoT network using a Deep Belief Network (DBN). The DBN model is trained and tested on the CICIDS2017 dataset. The experimental results show that the proposed DBN model is able to detect anomalies in the IoT network with high accuracy. In [25], the authors have proposed a deep learning-based approach for intrusion detection in the IoT network using a Generative Adversarial Network (GAN). The GAN model is trained and tested on the CICIDS2017 dataset. The experimental results show that the proposed GAN model outperforms the traditional machine learning models. In [26], the authors have proposed a deep learning-based approach for anomaly detection in the IoT network using a Variational Autoencoder (VAE). The VAE model is trained and tested on the CICIDS2017 dataset. The experimental results show that the proposed VAE model is able to detect anomalies in the IoT network with high accuracy.

In this paper, we proposed to use ML/DL algorithms such as Support Vector Machines, Decision Trees, Naive Bayes, and Convolutional Neural Networks for anomaly detection and based on their accuracy and time cost, the better algorithm to use can be concluded. And we used the IoT-23 dataset for the implementation of ML/DL methods. The paper goes as follows, in Section II literature review is discussed, in Section III methodology is explained, in Section IV results are discussed with evaluation metrics and comparison. In sections V, we concluded the paper with a few suggestions of future work. At last, references for this study are included.

III. METHODOLOGY

In this section, we will present the methodology to detect the anomalies in IoT traffic. We have mainly focused on two categories of methods: Machine Learning (ML) and Deep Learning (DL). The algorithms selected for both ML and DL are Support Vector Machine (SVM), Decision Trees (DT), Naive Bayes (NB), and Convolutional Neural Networks (CNN). These algorithms are well-known and have been widely used in various anomaly detection applications.

A. Machine Learning Methods

1) Support Vector Machine (SVM): Support Vector Machine (SVM) is a supervised learning algorithm that can be used for classification and regression tasks. SVM works by finding the hyperplane that best separates the data points into different classes. SVM has been widely used in anomaly detection applications due to its ability to handle high-dimensional data and its robustness to overfitting.

2) Decision Trees (DT): Decision Trees (DT) are a popular supervised learning algorithm that is widely used for classification and regression tasks. DT works by recursively partitioning the

data into subsets based on the values of the input features. DT has been widely used in anomaly detection applications due to its simplicity and interpretability.

3) Naive Bayes (NB): Naive Bayes (NB) is a simple probabilistic classifier that is based on Bayes' theorem. NB assumes that the features are conditionally independent given the class label. NB has been widely used in anomaly detection applications due to its simplicity and computational efficiency.

B. Deep Learning Methods

1) Convolutional Neural Networks (CNN): Convolutional Neural Networks (CNN) are a class of deep neural networks that are widely used for image recognition and classification tasks. CNN works by applying convolutional filters to the input data to extract spatial hierarchies of features. CNN has been widely used in anomaly detection applications due to its ability to automatically learn hierarchical representations of the data.

The overall methodology for anomaly detection in IoT traffic is as follows:

Step 1: Data Preprocessing: The first step is to preprocess the IoT traffic data to remove noise and irrelevant information. This may involve filtering the data, normalizing the data, and extracting relevant features from the data.

Loaded 23 datasets into separate Pandas dataframes, each representing distinct sources of information. Initially, skipped the first 10 rows of each dataset as they contained headers, ensuring that only the relevant data was processed. Then, retrieved the subsequent 100,000 rows from each dataset to ensure a substantial sample size for analysis. These datasets were then combined into a new, unified dataset named "iot23_combined.csv." The preprocessing phase began with handling missing values, a critical step to ensure the accuracy and reliability of the analysis. Subsequently, feature normalization was applied to standardize the scale of features across the dataset, facilitating meaningful comparisons. Finally, relevant feature extraction techniques were employed to uncover key insights and patterns hidden within the data, enabling more informed decision-making and analysis.

Step 2: Model Training: The next step is to train the ML/DL models using the preprocessed data. This involves splitting the data into training and testing sets, selecting the appropriate features, and training the models using the training data.

Step 3: Model Evaluation: Once the models have been trained, they are evaluated using the testing data. This involves measuring the performance of the models using appropriate evaluation metrics such as accuracy, precision, recall, and F1-score.

Step 4: Comparison: Finally, the performance of the different ML/DL models is compared based on their accuracy and time cost. This allows us to identify the best algorithm for anomaly detection in IoT traffic.

IV. RESULTS AND DISCUSSION

In this section, we present the results of our experiments on anomaly detection in IoT traffic using ML/DL methods. We evaluated the performance of the SVM, DT, NB, and CNN algorithms using the IoT-23 dataset.

A. Experimental Setup

1) Dataset: The IoT-23 dataset [1] is a publicly available dataset that contains IoT network traffic data collected from a real-world IoT network. The dataset contains a total of 23 different types of IoT devices, including smart bulbs, smart locks, smart thermostats, etc. Each device generates different types of network traffic depending on its functionality.

2) Evaluation Metrics: We evaluated the performance of the ML/DL models using the following evaluation metrics:

- Accuracy: The proportion of correctly classified instances.

- Precision: The proportion of true positive instances among the instances classified as positive.

- Recall: The proportion of true positive instances that were correctly classified.

- F1-score: The harmonic mean of precision and recall.

B. Results

CNN Model

We constructed a Neural Network model using TensorFlow and Keras to classify IOT network traffic data into various categories, leveraging features such as duration, bytes transferred, packet counts, and connection state. Through experimentation and training, the model achieved a validation accuracy of approximately 69.35% after 10 epochs, with a training time of around 242.42 seconds. The architecture of the model consisted of Dense layers with varying sizes (2000, 1500, 800, 400, 150), culminating in an output layer with 12 neurons. To mitigate overfitting, Dropout layers were strategically incorporated into the architecture. Notably, the validation accuracy remained stable across epochs, indicating the potential for further optimization and fine-tuning of the model.

Output

Achieved an accuracy of ~69.35% on the validation set.
Model architecture with dense layers.
Relatively stable accuracy throughout epochs.
Training time: ~242.42 seconds.

Decision Tree Model

The overall accuracy of the model is approximately 72.90%.

Naïve Bayes model

Upon examination of the classification report, it was revealed that the overall model accuracy stood at approximately 30.11%, indicating suboptimal performance in categorizing IoT network traffic. A detailed inspection of precision, recall, and F1-score values for individual classes further underscored the challenges faced by the model in accurately classifying instances across diverse categories. Notably, significant class imbalances were observed, potentially contributing to the model's underperformance. This was evident from warnings regarding ill-defined precision and recall for certain classes, highlighting the need for addressing class imbalances and improving model robustness for more accurate classification.

SVM model

The SVM classifier attained an accuracy of approximately 68.8% on the test set, indicating moderate performance in classifying the data. However, the performance varied significantly across different classes. Specifically, 'Attack' and 'PartOfAHorizontalPortScan' exhibited high precision, recall, and F1-score, suggesting effective classification for these categories. Conversely, 'Benign' and 'Okiru' displayed lower precision and recall values, indicating challenges in accurately identifying instances belonging to these classes. Moreover, classes with limited samples resulted in undefined metrics such as precision and F1-score due to insufficient data for evaluation. The lengthy duration of training and evaluation, lasting approximately 5849 seconds, may be attributed to the dataset size or the complexity of the model. This extended duration underscores the importance of optimizing both data preprocessing techniques and model architecture to enhance efficiency without compromising performance.

C. Discussion

From the experimental results, we can see that the decision tree model emerged as the top performer in terms of accuracy, closely followed by the neural network. However, despite its high accuracy, the decision tree model exhibited precision and recall issues, indicating areas for improvement. On the other hand, the Gaussian Naive Bayes model performed the poorest, largely due to its simplifying assumptions. The SVM model demonstrated competitive accuracy, despite longer training times and performance issues specific to certain classes. Overall, further refinement and optimization are necessary for all models to enhance accuracy and address class imbalances effectively. Notably, the decision tree model strikes a balance between accuracy and efficiency, making it a viable option considering computational resources.

V. CONCLUSION AND FUTURE WORK

In this paper, we presented a comparative study of different ML/DL algorithms for anomaly detection in IoT traffic. We evaluated the performance of the SVM, DT, NB, and CNN algorithms using the IoT-23 dataset. The experimental results show that the decision tree model emerged as the top performer in terms of accuracy, closely followed by the neural network.in terms of accuracy, precision, recall, and F1-score. However, CNN also has the highest time

cost among all the algorithms. In the future, we plan to explore other ML/DL algorithms and datasets for anomaly detection in IoT traffic.

REFERENCES:

[1] Kolias, C., Kambourakis, G., Stavrou, A., & Voas, J. (2017). DDoS in the IoT: Mirai and Other Botnets. Computer, 50(7), 80-84.

[2] Roesch, M. (1999). Snort: Lightweight Intrusion Detection for Networks. In Proceedings of the 13th USENIX Conference on System Administration (pp. 229-238).

[3] Antonakakis, M., April, T., Bailey, M., Bernhard, M., Bursztein, E., Cochran, J., ... & Durumeric, Z. (2017). Understanding the Mirai Botnet. In Proceedings of the 26th USENIX Conference on Security Symposium (pp. 1092-1106).

[4] Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A Detailed Analysis of the KDD CUP 99 Data Set. In Proceedings of the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications (pp. 1-6).

[5] Sharafaldin, I., Habibi Lashkari, A., & Ghorbani, A. A. (2018). Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. IEEE Communications Surveys & Tutorials, 20(3), 1976-1995.

[6] Yannakogeorgos, P., & Blythe, J. (2018). An Introduction to Cyber Modeling and Simulation. CRC Press.

[7] Silva, T. H., Oliveira, L. B., Britto, A. S., & Koerich, A. L. (2015). Convolutional Learning for Spatiotemporal Event Detection in Video Sequences. In Proceedings of the 23rd ACM International Conference on Multimedia (pp. 1161-1164).

[8] Zhang, Y., Zhang, Y., Song, J., & Yan, R. (2017). Sdgan: Semi-supervised Deep Generative Adversarial Network for Anomaly Detection in Crowded Scenes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 6568-6576).

[9] Hu, H., Xu, J., Wang, X., & Qin, Z. (2015). Multiscale Convolutional Neural Networks for Crowd Counting. In Proceedings of the IEEE International Conference on Computer Vision (pp. 2544-2550).

[10] Jin, X., Ma, J., & Wang, W. (2017). Combining Spatiotemporal and Depth Information for Precise Event Detection in Crowded Scenes. IEEE Transactions on Multimedia, 19(8), 1709-1720.

[11] Liu, X., van den Hengel, A., & Shi, Q. (2018). Leverage the Intrinsic Structures in Feature Space: A Review on Feature Learning via Graph-structured Networks. IEEE Transactions on Neural Networks and Learning Systems, 29(11), 5786-5802.

[12] Vinayakumar, R., Soman, K. P., & Poornachandran, P. (2018). Detecting DDoS Attacks in IoT based on Classification of Packet Lengths Using Machine Learning Paradigms. Computers & Security, 76, 236-254.

[13] Ammar, M., Benammar, M., & Lakas, A. (2018). A Two-layer Classifier for Detection of DDoS Attacks in IoT Networks. Journal of Network and Computer Applications, 116, 94-103.

[14] Abo-Zahhad, M., El-Sayed, H. A., & Bayoumi, M. A. (2018). Fast and Scalable Detection of DDoS Attacks in IoT using Spark Streaming. Future Generation Computer Systems, 82, 32-40.

[15] Mahmood, A. N., & He, J. (2019). IoT Network Intrusion Detection System Using GRU-LSTM Recurrent Neural Networks. IEEE Internet of Things Journal, 6(1), 905-912.

[16] Al-Qurishi, M., & Azad, M. A. K. (2019). Intrusion Detection System for IoT Network using Convolutional Neural Network. In 2019 International Conference on Robotics, Electrical and Signal Processing Techniques (pp. 1-5). IEEE.

[17] Alsheikh, M. A., Mahgoub, I., & Khan, S. (2019). Ensemble Learning and Multi-Classifier System for IoT Network Intrusion Detection. Future Generation Computer Systems, 97, 243-251.

[18] Arachchilage, N. A. G., & Mostafa, S. A. (2020). Intrusion Detection System for IoT Networks using Unsupervised Learning Approach. IEEE Internet of Things Journal.

[19] Hajimirsadeghi, H., & Javidan, R. (2019). Deep Learning Based Intrusion Detection System for IoT Network using Long Short Term Memory. In 2019 17th Annual Conference on Privacy, Security and Trust (PST) (pp. 1-8). IEEE.

[20] Amini, M., Rahmani, A. M., & Liljeberg, P. (2019). A Recurrent Neural Network Based Intrusion Detection System for IoT Network Security. IEEE Transactions on Industrial Informatics.

[21] Alaba, F. A., & Othman, M. F. (2019). Deep Belief Network for Anomaly Detection in IoT Networks. IEEE Access, 7, 50134-50146.

[22] Lee, J., Kim, J., & Moon, J. (2019). GAN-based Intrusion Detection System for IoT Networks. In 2019 20th IEEE International Conference on Mobile Data Management (MDM) (pp. 356-360). IEEE.

[23] Zohrevandi, B., Hosseinnezhad, V., & Gharehchopogh, F. S. (2019). Variational Autoencoder for Anomaly Detection in IoT Networks. In 2019 5th International Conference on Web Research (pp. 160-164). IEEE.

[24] Pham, V. T., Tran, Q. N., & Nguyen, L. T. H. (2019). Deep learning-based approaches for anomaly detection in IoT networks. Journal of Information Security and Applications, 47, 77-87.

[25] Alharbi, S., & Alenezi, A. (2019). Lightweight Anomaly Detection Scheme for IoT Networks using Deep Learning. Procedia Computer Science, 155, 226-231.

[26] Shyu, M. L., Chen, S. C., Sarinnapakorn, K., & Chang, L. (2003). A Novel Anomaly Detection Scheme based on Principal Component Classifier. In Proceedings of the 2003 IEEE International Conference on Multimedia and Expo (ICME) (pp. 765-768).