Attack and Anomaly Detection in IoT Sensors and Sites

Harsh Agarwal - 181IT117
Dept. of Information Technology
NITK Surathkal
Mangaluru, India
agrawal.harsh14@gmail.com

Amith Bhat - 181IT105
Dept. of Information Technology
NITK Surathkal
Mangaluru, India
amithbhat01@gmail.com

Kumsetty Nikhil Venkat - 181IT224
Dept. of Information Technology
NITK Surathkal
Mangaluru, India
nikhilvenkat26@gmail.com

***Abstract*—Attack and anomaly detection in the Internet of Things (IoT) infrastructure is a rising concern in the domain of IoT. With the increased use of IoT infrastructure in every domain, threats and attacks in these infrastructures are also growing commensurately. Denial of Service, Data Type Probing, Malicious Control, Malicious Operation, Scan, Spying and Wrong Setup are some commonly observed attacks and anomalies which can cause an IoT system failure. In this project, performances of several machine learning models have been compared to predict attacks and anomalies on the IoT systems accurately.**

***Index Terms*—IoT, Machine Learning, DoS, Cybersecurity , Anomaly detection**

I. Introduction

With the increasing demand and growth in data-driven infrastructure, there has been a lot of research on the topics of Internet of Things(IoT) and Machine Learning(ML) . For example, there have been several papers exploring the application of ML in such diverse fields such as medicine, aerospace, financial, and educational sectors .

Besides machine learning, IoT services are also applied to these domains. The growing complexity in IoT infrastructures is raising unwanted vulnerability to their systems. In IoT devices, security breach and anomaly has become common phenomena nowadays.

IoT devices use a wireless medium to broadcast data which makes them an easier target for an attack. Normal communication attack in the local network is limited to local nodes or small local domain, but attack in IoT system expands over a larger area and has devastating effects on IoT sites.

Vulnerability in IoT nodes makes a backdoor for an attacker to gather confidential data from any important organization. For some stakeholders and entrepreneurs, data is the money for their business. For the government and some private agency, some data are classified and confidential. Hence, a secured IoT infrastructure is necessary for the protection from cybercrimes.

There are some trivial methods to solve the problems as mentioned above. For example, in signature based methods, attacks and anomaly are previously stored in a database. Moreover, this system is checked at particular time intervals against the database. However, this methodology generates overhead in processing, and it is vulnerable to unknown threats.

The advantage of data analysis based technique is that it works faster than other methodologies and it can overcome the problem raised from unknown threats. Hence, in this paper data analysis based techniques are used.

II. Objectives

The primary goal of the system is to develop a smart, secured and reliable IoT based infrastructure which can detect its vulnerability, have a secure firewall against all cyber attacks and recover itself automatically.

Here, an ML-based solution is proposed which can detect and protect the system when it is in the abnormal state. For this task, several ML classifiers have been exploited.

III. Methodology

In this section, we describe the structure of the proposed Machine Learning solution.

The overall framework is a combination of several independent processes. Fig. 1 depicts the overall framework of the system.

The first process of this framework is the dataset collection and dataset observation. In this process, the dataset was collected and observed meticulously to find out the types

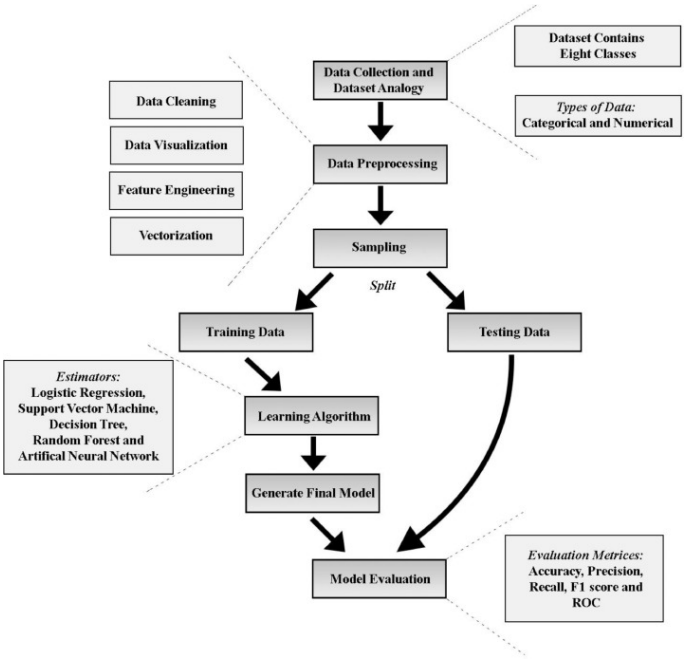

Fig. 1. Methodology of the proposed ML solution

of data. Besides, data preprocessing was implemented on the dataset. Data preprocessing consists of cleaning of data, visualization of data, feature engineering and vectorization steps.

These steps converted the data into feature vectors. These feature vectors were then split into 80–20 ratio into training and testing set. The training set was used in Learning Algorithm, and a final model was developed using an optimization technique. Different classifiers used in this work employed different optimization techniques. While the Logistic Regression used coordinate descent, the Support Vector machine used conventional gradient descent technique.

The final model was evaluated against the testing set using different evaluation metrics such as accuracy, precision, recall, F1 score, and Area under the Receiver Operating Characteristic (ROC) Curve.

A. *Obtaining the Data*

An Open Source dataset [**?**] of traffic traces in a virtual IoT environment is chosen for the problem. The data is generated by making use of a Virtual Environment created using Distributed Smart Space Orchestration System (DS2OS).

In the dataset, there are 357,952 samples and 13 features. The features along with their types are mentioned as follows:
1. Source ID: Nominal
2. Source Address: Nominal
3. Source Type: Nominal
4. Source Location Nominal
5. Destination Service Address: Nominal
6. Destination Service Type: Nominal
7. Destination Location: Nominal
8. Accessed Node Address: Nominal
9. Accessed Node Type: Nominal
10. Operation: Nominal
11. Value: Continuous
12. Timestamp: Discrete
13. Normality: Nominal

The dataset has 347,935 Normal data and 10,017 anomalous data and contains eight classes which were classified. Features "Accessed Node Type" and "Value" have 148 and 2050 missing data, respectively. The class label is Normality and the data is divided into 8 different classes according to Normality:
1. Denial of Service (Dos): The attacker sends too many ambiguous packets to flood out the target and make its services unavailable to other services.
2. Data Type Probing (D.P): In this case, a malicious node writes different data type than intended data type.
3. Malicious Control (M.C): The attacker can gain a valid session key and capture network traffic.
4. Malicious Operation (M.O): Malicious Operations are generally caused by malware.
5. Scan(SC): The data can get corrupted when it is acquired through hardware by scanning the system.
6. Spying (SP): The attacker exploits the vulnerabilities of the system, and they use a backdoor channel to break into the system and discovers important information.
7. Wrong Setup (W.S): the data may also get disrupted by the wrong system setup.
8. Normal(NL): if the data is entirely correct and accurate, then the data is called normal data.

B. *Pre-Processing the Data*

After Processing the data and extracting the necessary information from it, we analyze the following statistics of the frequency of different attacks in the data:

The timestamp column from the dataset has been removed as it has a minimal correlation to the dataset's predictor variable normality. Also a major task was to convert nominal categorical data into vectors. Categorical data can be converted into vectors in many ways. Label Encoding and One Hot

	Freq Count
Dos	5780
Data type Probing	342
Malicious Control	889
Malicious Operation	805
Scan	1547
Spying	532
Wrong Setup	122

TABLE I
TABLE 1: FREQUENCY DISTRIBUTION OF ATTACKS

Encoding is prevalent among them. In this project label encoding technique have been used to convert the data into a feature vector. Most of the features in this dataset contain nominal categorical value and many unique values. If one hot encoding were applied to these features, the number of features would have increased with a significant number, and the resulting dataset would have lots of dimensions. On the other case, by label encoding, the number of features were the same. Thus the dimension of the dataset was not increased. Besides these, one hot encoded features would have sparse features which are harder to fit in machine learning algorithm and takes a lot of processing time. Hence, label encoding is applied to the dataset.

C. *Applying the Algorithms*

*1) **Logistic Regression(LR)**:* Logistic Regression (LR) is a discriminative model which depends on the quality of the dataset. Given the features X = X 1 , X 2 , X 3 , . . . , X n (where, X 1 X n = Distinct features), weights W = W 1 , W 2 , W 3 , . . . , W n , bias b = b 1 , b 2 , . . . , b n and Classes C = c 1 , c 2 , . . . , c n (in our case, we have eight classes) the equation for estimation of posterior is given in following.

$$\text{Predicted Value: } p(y = C|X;W,b) = \frac{1}{1 + exp(-W^{transpose}X - b)}$$

Fig. 2. Predicted value for each class

*2) **Support Vector Machine(SVM)**:* SVM - Support Vector Machine is another discriminative model like Logistic regression. It is a supervised learning model for analyzing the data used for classification, regression, and outliers detection. Support Vector machine is most applicable in the case of Non-Linear data. Given Input X , Class or Label C and LaGrange multipliers ; weight vector can be calculated by following equation:

$$\Theta = \sum_{i=1}^{m} \alpha_i c_i x_i$$

Fig. 3. Weight Vector equation

The target of the Support Vector machine is to optimize the following equation:

$$\text{Maximize}_{\alpha_i} \sum_{i=1}^{m} \alpha_i - \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j c_i c_j < x_i x_j >$$

Fig. 4. Support vector machine equation

In the above equation ¡ x i , x j ¿is a vector which can be obtained by different kernels like polynomial kernel, Radial Basis Function kernel and Sigmoid Kernel.

*3) **Decision Trees(DT)**:* A Decision Tree starts with a single node and then it branches into possible outcomes. Each of these outcomes lead to additional nodes, which branch off into other instances. Given, features x, impurity measure I(data), the number of samples in parent node Pn, the number of samples in left child LCn and the number of samples in right child RCn; Decision Tree's target is to maximize following Information Gain given as follows:

$$\text{Information Gain}(P_d, x) = I(P_d) - \frac{LC_n}{P_n} I(LC_d) - \frac{RC_n}{P_n} I(RC_d)$$

Fig. 5. Information Gain in Decision Tree

Impurity Measure I(data) can be calculated in three techniques Gini Index IG, Entropy IH and Classification Error IE:

$$I_H(n) = -\sum_{i=1}^{c} p(c|n) log_2 p(c|n)$$

$$I_G(n) = 1 - \sum_{i=1}^{c} p(c|n)^2$$

$$I_E(n) = 1 - max\{p(c|n)\}$$

Fig. 6. Impurity Measure in Decision Tree

*4) **Random Forest(RF)**:* Random forest algorithm creates the forest with many decision trees. It is a supervised classification algorithm. It is an attractive classifier due to the high execution speed . Many decision trees ensemble together to form a random forest, and it predicts by averaging the predictions of each component tree. It usually has much better predictive accuracy than a single decision tree.

*5) **Artificial Neural Network(ANN)**:* Artificial Neural Network (ANN) is a deep learning method which trains the model based on raw data. Compared to other classifiers it has a large number of parameters for tuning which makes it a complex structure. It also takes a long time to optimize error than other techniques. Each single Neuron Node of Artificial Neural Network is trained with feature set X = X1, X2, X3, . . . , Xn (where, X1 Xn = Distinct features). The features are multiplied by some random weights, W = W1,W2,W3, . . . ,Wn and added with bias values, b = b1, b2, . . . , bn. The values are then given as input in non-linear activation function.

IV. Results and Analysis

In the Data Analysis subsection, it has been described that several machine learning techniques were applied to the dataset. Five-fold cross-validation was performed on the

dataset using each of these techniques. Fig. 7 and Fig. 8 shows how the accuracy results are converged after five-fold cross-validation.

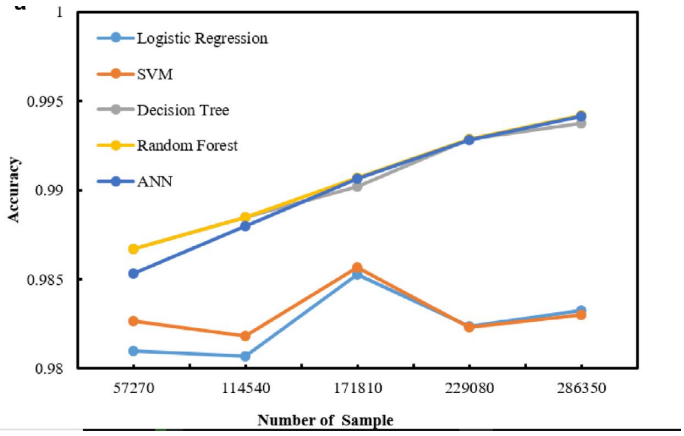

Fig. 7. Training accuracy for different techniques for 5 fold cross validation

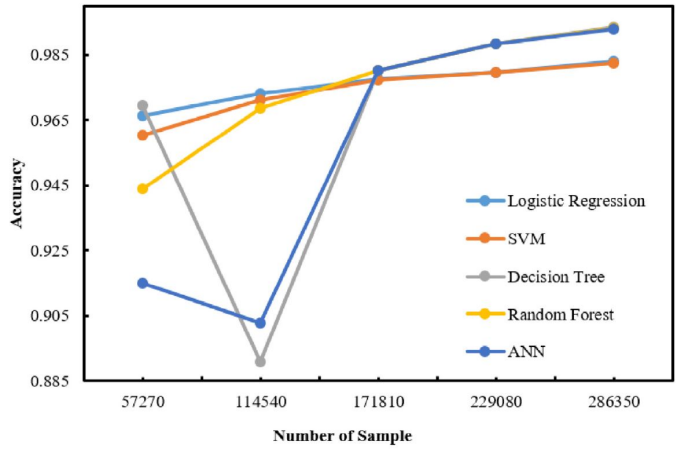

Fig. 8. Testing accuracy for different techniques for 5 fold cross validation

From the cross-validation, it can be inferred that Random Forest and Artificial Neural Network have performed best both in training and testing accuracy. Decision Tree performed with approximate similarity to Random Forest and Artificial Neural Network in the case of training. In the case of testing, the Decision Tree had most deviations than other techniques and performed poorly at first. However in the last three folds, it performed similarly to Random Forest and Artificial Neural Network. Support Vector machine and Logistic regression performed weakly than other techniques in training. In the case of testing and in the first two fold, Support Vector machine and Logistic regression both performed better than other techniques and logistic regression was best among them, but at the last three folds, they performed worse than others.

Evaluation Metrics		Classifiers				
		LR	SVM	DT	RF	ANN
Training	Accuracy	0.983	0.982	0.994	0.994	0.994
	STD(+/-)	0.0012	0.0015	0.00081	0.00081	0.0013
	Precision	0.98	0.98	0.99	0.99	0.99
	Recall	0.98	0.98	0.99	0.99	0.99
	F1 Score	0.98	0.98	0.99	0.99	0.99
Testing	Accuracy	0.983	0.982	0.994	0.994	0.994
	STD(+/-)	0.0055	0.0064	0.016	0.014	0.021
	Precision	0.98	0.98	0.99	0.99	0.99
	Recall	0.98	0.98	0.99	0.99	0.99
	F1 Score	0.98	0.98	0.99	0.99	0.99

Fig. 9. Evaluation metrics of our models

Fig. 9 represents different evaluation metrics for different techniques trained on the dataset. From Fig. 9 , it can be seen that Decision Tree and Random Forest have more accuracy, precision, recall, and F1 score values than other techniques. Artificial Neural Network also performed well in the case of evaluation. However, Decision Tree and Random Forest are a little more accurate than Artificial Neural Network. On the other case, Logistic regression and Support Vector machine also do well on our dataset but not as good as other classifiers. Now considering the confusion matrices of each technique, the most optimized technique can be found. From the confusion matrices it can be concluded that Random Forest is the best technique for this work.

RF

	DoS	D.P	M.C	M.O	SC	SP	W.S	NL
DoS	775	0	0	0	0	0	0	403
D.P	0	63	0	0	0	0	0	0
M.C	0	0	169	0	0	0	0	0
M.O	0	0	0	155	0	0	0	0
SC	0	0	2	0	305	0	0	0
SP	0	0	0	0	0	120	0	0
W.S	0	0	0	0	0	0	28	0
NL	18	0	0	0	0	2	0	69553

Fig. 10. Confusion matrix of Random Forest model

From Fig. 10, Random Forest classified every class correctly except DoS and Normality classes. Out of 1178 samples of DoS, it misclassified 403 samples as Normal. Moreover, for Normal class, it misclassified 18 samples as DoS out of 69,571 samples.

DT

	DoS	D.P	M.C	M.O	SC	SP	W.S	NL
DoS	775	0	0	0	0	0	0	403
D.P	0	63	0	0	0	0	0	0
M.C	0	0	169	0	0	0	0	0
M.O	0	0	0	155	0	0	0	0
SC	0	0	2	0	305	0	0	0
SP	0	0	0	0	0	120	0	0
W.S	0	0	0	0	0	0	28	0
NL	18	0	0	0	0	2	0	69551

Fig. 11. Confusion matrix of Decision Tree model

From Fig. 11, Confusion matrix for Decision Tree is similar to Random Forest except for the Normal class. In the case of Normal class, it misclassified 18 samples as DoS and two samples as spying out of 69,571 samples.

ANN

	DoS	D.P	M.C	M.O	SC	SP	W.S	NL
DoS	775	0	0	0	0	0	0	403
D.P	0	63	0	0	0	0	0	0
M.C	0	0	169	0	0	0	0	0
M.O	0	0	0	155	0	0	0	0
SC	0	0	2	0	305	0	0	0
SP	0	0	0	0	0	120	0	0
W.S	0	0	0	0	0	0	28	0
NL	18	0	1	0	0	2	0	69550

Fig. 12. Confusion matrix of Artificial Neural Network model

From Fig. 12, Artificial Neural Network performed similarly to Decision Tree. It only misclassified one more sample than Decision Tree. Artificial Neural Network correctly predicted every sample of 6 labels out of 8 labels. In the case of DoS, Artificial Neural Network misclassified 403 samples as Normal out of 1178 samples. While for the Normal Label, Artificial Neural Network misclassified 18 samples as DoS, two samples as Spying and 1 sample as Malicious Control out of 69,571 samples.

LR

	DoS	D.P	M.C	M.O	SC	SP	W.S	NL
DoS	775	0	0	0	0	0	0	403
D.P	0	0	0	0	0	0	0	63
M.C	0	0	10	0	0	0	0	159
M.O	0	0	0	78	0	0	0	77
SC	5	0	2	0	0	0	0	298
SP	0	0	0	0	0	0	0	120
W.S	0	0	0	0	0	0	0	28
NL	34	0	0	9	0	0	0	69528

Fig. 13. Confusion matrix of Logistic regression model

Logistic regression performed very poorly in this system.Fig. 13, shows the confusion matrix of Logistic regression. Logistic regression correctly classified 775 samples of DoS class but misclassified all the remaining 403 samples as Normal class. For Data Probing, it misclassified 63 samples as Normal out of 63 samples, for Malicious Control it misclassified 159 samples as Normal out of 169 samples, for Malicious operation it misclassified 77 samples as Normal out of 155 samples, for scan it misclassified 5 samples as DoS and 298 samples as Normal out of 305 samples, for spying it misclassified 120 samples as Normal out of 120 samples, for the Wrong Setup it misclassified 28 samples as Normal out of 28 samples and lastly for Normal, it misclassified 34 samples as DoS and 9 samples as Malicious Operation out of 69,571 samples.

SVM

	DoS	D.P	M.C	M.O	SC	SP	W.S	NL
DoS	775	0	0	0	0	0	0	403
D.P	0	0	0	0	0	0	0	63
M.C	0	0	10	0	0	0	0	159
M.O	0	0	0	33	0	0	0	122
SC	0	0	2	0	0	0	0	303
SP	0	0	0	0	0	0	0	120
W.S	0	0	0	0	0	0	0	28
NL	34	0	0	0	0	0	0	69537

Fig. 14. Confusion matrix of Support Vector machine model

In the case of Support Vector machine, it correctly classified 775 samples of DoS class but misclassified 403 samples as Normal class. Support Vector machine misclassified all samples of data probing, spying and the wrong setup as normal, for malicious control it misclassified 159 samples as Normal out of 169 samples, for malicious operation it misclassified 122 samples as Normal out of 155 samples, for scan it misclassified 2 samples as Malicious Operation and 303 samples as Normal out of 305 samples and for Normal, it misclassified 34 samples as DoS out of 69,571 samples.

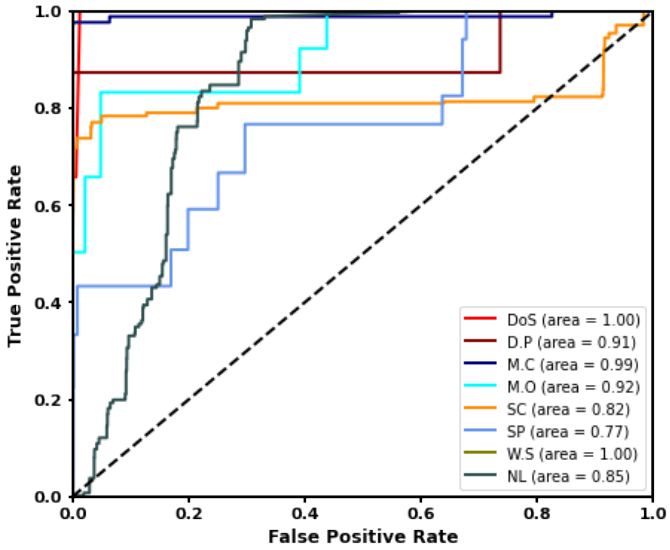

Fig. 15. ROC of Logistic regression model

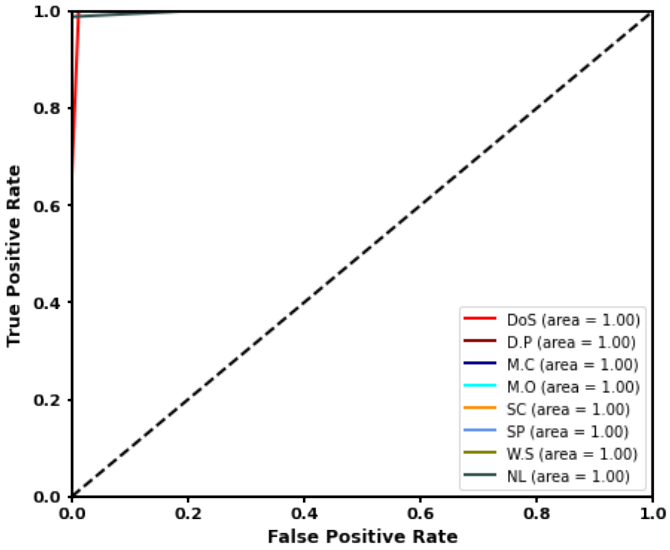

Fig. 17. ROC of Decision Tree model

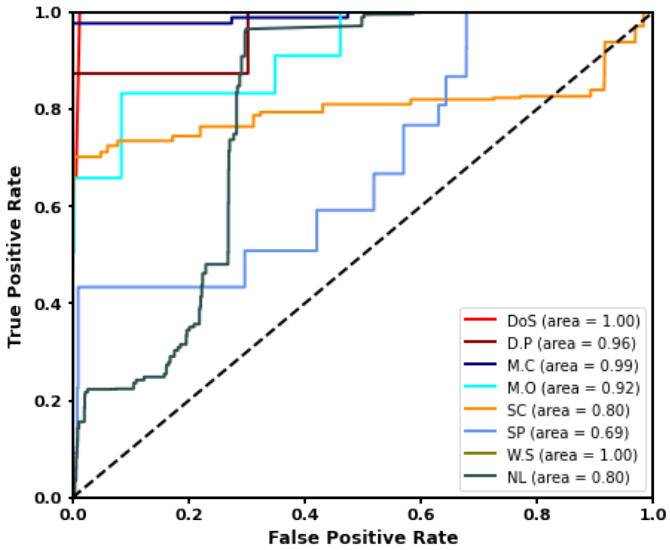

Fig. 16. ROC of Support Vector machine model

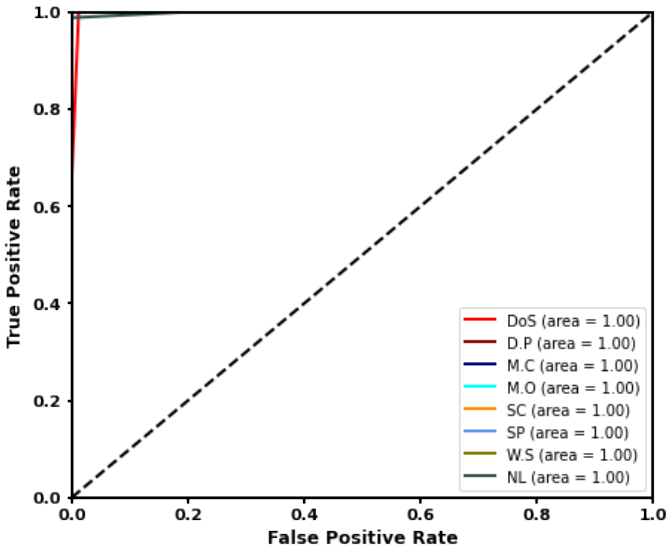

Fig. 18. ROC of Random Forest model

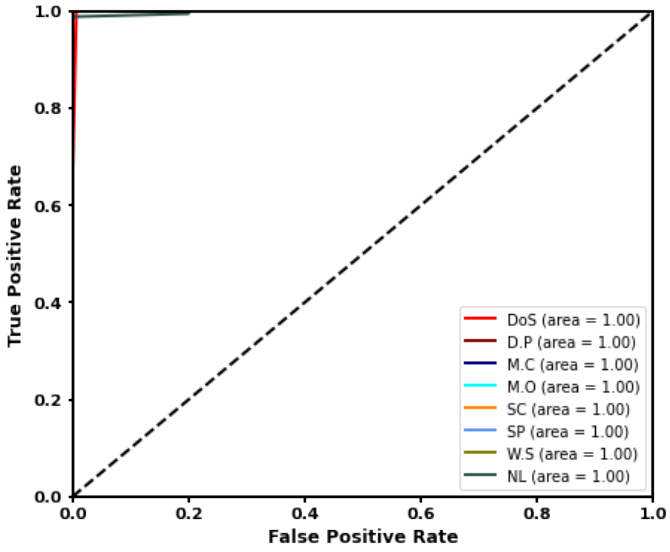

Fig. 19. ROC of Artificial Neural Network model

Lastly, the figures above depicts the Receiver Operating Characteristic (ROC) Curves of Logistic regression, Support Vector machine, Decision Tree, Random Forest, and Artificial Neural Network respectively. From area under the curves, it can be described that Decision Tree, Random Forest, and Artificial Neural Network have higher accuracy because all of the area under the curves for every class is approximately equivalent to value one. While in case of Logistic regression and Support Vector machine, only for DoS and Wrong Setup the area under the curve is equivalent to one.

V. Conclusions and Future Work

In this project, we observe that Random Forest classifier performs comparatively better than all the other algorithms with the accuracy of 99.3%.

It was found that the Random Forest algorithm is the best one to be applied on these kinds of datasets because Random Forest predicted Data Probing, Malicious Control, Malicious Operation, Scan, Spying, and Wrong Setup attacks accurately compared to other approaches. It also predicted the Denial of Service and Normal samples more accurately than any other ML model.

While this project has focused on classical ML models such as Support Vector machine, Artificial Neural Network, etc., we would like to work on designing and implementing a new algorithm specifically tailored for IoT systems.

Also, the dataset used was created using virtual environment data, hence we would like to perform this project again using empirical, real-world data.

Acknowledgments

We would like to thank Dr. Bhawana Rudra who helped us shape the idea of our project.

References

[1] M.Hasan, Md. Islam, Md. Israk Zarif, M.M.A. Hashem, Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches

[2] M.-O. Pahl, F.-X. Aubet, S. Liebald, Graph-based IoT microservice security, in: Proceedings of the NOMS 2018–2018 IEEE/IFIP Network Operations and Management Symposium, IEEE, 2018, pp. 1–3.

[3] M.-O. Pahl, F.-X. Aubet, All eyes on you: distributed multi-dimensional IoT microservice anomalydetection, in: Proceedings of the 2018 Fourteenth International Conference on Network and Service Management (CNSM)(CNSM 2018), 2018. Rome, Italy

[4] L. Wang, Support Vector Machines: Theory and Applications, 177, Springer Science Business Media, 2005.

[5] X. Liu, Y. Liu, A. Liu, L.T. Yang, Defending on–off attacks using light probing messages in smart sensors for industrial communication systems, IEEE Trans. Ind. Inf. 14 (9) (2018) 3801–3811.

[6] H.H. Pajouh, R. Javidan, R. Khayami, D. Ali, K.-K.R. Choo, A two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detection in iot backbone networks, IEEE Trans. Emerg. Top. Comput. (2016).

[7] I. Poyner, R. Sherratt, Privacy and security of consumer IoT devices for the pervasive monitoring of vulnerable people. (2018).

[8] T.K. Ho, Random decision forests, in: Proceedings of the third international conference on Document analysis and recognition, 1995, 1, IEEE, 1995, pp. 278–282.

[9] U.S. Shanthamallu, A. Spanias, C. Tepedelenlioglu, M. Stanley, A brief survey of machine learning methods and their sensor and IoT applications, in: Proceedings of the 2017 Eighth International Conference on Information, Intelligence, Systems and Applications (IISA), IEEE, 2017, pp. 1–8.

[10] J. Liu, Y. Xiao, C.P. Chen, Authentication and access control in the internet of things, in: Proceedings of the 2012 Thirty-second International Conference on Distributed Computing Systems Workshops (ICDCSW), IEEE, 2012, pp. 588–592.

[11] O. Brun, Y. Yin, E. Gelenbe, Y.M. Kadioglu, J. Augusto-Gonzalez, M. Ramos, Deep learning with dense random neural networks for detecting attacks against IoT-connected home environments, in: Proceedings of the 2018 ISCIS Security Workshop, Imperial College London. Recent Cybersecurity Research in Europe. Lecture Notes CCIS, in: 821, 2018.

[12] B. Usmonov, O. Evsutin, A. Iskhakov, A. Shelupanov, A. Iskhakova, R. Meshcheryakov, The cybersecurity in development of IoT embedded technologies, in: Proceedings of the 2017 International Conference on Information Science and Communications Technologies (ICISCT), IEEE, 2017, pp. 1–4.

[13] M.-O. Pahl, F.-X. Aubet, DS2OS traffic traces, 2018, (https://www.kaggle.com/francoisxa/ds2ostraffictraces). [Online; accessed 29-December-2018]

[14] J. Milosevic, N. Sklavos, K. Koutsikou, in: Malware in IoT software and hardware, 2016.

[15] R. Kozik, M. Choras´, M. Ficco, F. Palmieri, A scalable distributed machine learning approach for attack detection in edge computing environments, J. Parallel Distrib. Comput. 119 (2018) 18–26.