Machine Learning based Intrusion Detection for Cyber-Security in IoT Networks

Amine Khatib[1,2,3], Mohamed Hamlich[1], Denis Hamad[2]

[1]ISSIEE, UH2C University, Casa, Maroc.
[2]LISIC, ULCO University, Calais, France.
[3]CERIM, HESTIM, Casablanca, Maroc.

Abstract. IoT network is a promising technology, IoT implementation is growing rapidly but cybersecurity is still a loophole, detection of attacks in IOT infrastructures is a growing concern in the field of IoT. With the increased use of Internet of Things in different areas, cyber-attacks are also increasing proportionately and can cause failures in the system. IDS becomes the leading security solution. Anomaly based network intrusion detection (IDS) detection plays a major role in protecting networks against various malicious activities. Improving the security of IoT networks has become one of the most critical issues. This is due to the large-scale development and deployment of IoT devices and the insufficiency of Intrusion Detection Systems (IDS) to be deployed for the use of special purpose networks. In this article, the performance of several machine learning models has been compared to accurately predict attacks on IoT systems, the case of imbalanced classes was subsequently treated using the SMOTE technique. The Nystrom based kernel SVM is the first time used to detect attacks in the IoT network and the results are promising. The evaluation metrics used in the performance comparison are accuracy, precision, recall, f1 score, and auc-roc curve.

1 Introduction

With and the growth of the IoT network system due to increasing demand, Internet of Thing is the latest promising emerging technology that connects everything in the world through the Internet. A large number of IoT devices and networks have been used in recent years for different areas of use [5]. Health [3], smart cities [4], supply chain [1] and agriculture [2]. With this extended use of IoT, new protocols are being deployed [6]. So IoT technology guarantees to improve and help our personal, professional and society life [1]. IoT models are getting more complicated every day [7, 8]. The IoT is a network of smart objects around the world through the internet without any human interference, which is great, but it is susceptible to cyber-attacks like any other network. research is geared towards machine learning based applications alongside IoT. Intrusion Detection System (IDS) is an effective technique for detecting cyber-attacks in any network. Most of the latest IDS are based on a machine learning algorithm for the detection of cyber-attacks in the network. currently their use is in all areas of human life [9].

The IoT network consists of connections between different types of smart objects ranging from supercomputers to small devices which can have very low computing power, so securing this type of network is difficult and therefore cybersecurity is a big loophole [10].

*

Statista has estimated the impressive number of connected IoT devices in 2020 and this number will double by 2025 [11]. Columbus reported that the world currently has a 20% adoption rate for IoT devices and expects to jump to 80% in less than a decade, which means that 20% of all services offered in the world involve technology related to IoT. With the massive increase in the use of IoT devices, many researchers have expressed concern about the security of IoT user information [12]. In 2016, a series of distributed denial of service attacks, targeting IoT networks and websites like Twitter, Netflix and PayPal really pointed out to the world how urgent it is to develop a strong security solution. Developing an IDS for IoT environments to mitigate these malicious attacks is crucial. An IDS designed for IoT networks must be able to analyse data and generate instant responses in real time and adapt to improved IoT infrastructure. Many researchers have shown promising results in detecting network intrusions, however, only a limited number target their research on IoT scenarios [12, 13].

* Corresponding authors: Email addresses: aminekhatib04@gmail.com, moha.hamlich@gmail.com, , denis.hamad@gmail.com,

We will present solutions based on machine learning that can detect and protect the system when it is in an abnormal state. we will also study the impact of data oversampling on the performance of the various machine learning models used, several machine learning classifiers were used. we will treat the binary and multi-class case and we will present a comparative study of the different techniques used after the use of the SMOTE technique to resampling and balance the data set. Further analysis and comparison with other work will be briefly described in the following sections. Section 2 provides a description of other researchs on anomaly detection in IoT network traffic. The description of the dataset, the different types of attacks as well as the different models and metrics used are detailed in section 3. Section 4 presents the results of our analysis and a comparative study. Finally, the conclusions and perspectives are presented in section 5.

2 Review

Recently, machine learning algorithms have shown promising performance in detecting abnormal and malicious activity in the loT network. Several similar works have been carried out in the fields of IoT.

Pahl et al. [7] mainly developed a detector and firewall for an anomaly of IoT microservices in an IoT site. Liu et al. [9] proposed a detector for On and Off-attack by a malicious network node in an industrial IoT site. By On and Off-attack, they meant that the IoT network could be attacked by a malicious node when it is in an active or On state. In addition, the IoT network behaves normally when its malicious node is in the idle or disabled state. In [15] the authors represented an intrusion detection system for the IoT. To this end, several ML classifiers have been used to successfully identify network analysis surveys and simple forms of Denial of Service (DoS) attacks. To generate the dataset, network traffic is taken for four consecutive days using Wireshark software. To apply ML classifiers, Weka software was used.

Ukil et al. [16] discussed the detection of anomalies in IoT-based healthcare analytics. A model for detecting heart abnormalities via a smartphone was also presented in this article. For the detection of anomalies in healthcare; IoT sensors, medical image analysis, biomedical signal analysis, big data mining, and predictive analytics have been used.

Pajouh et al. [17] presented an intrusion detection model based on a two-layer dimension reduction and two-level classification module. This model was also designed to identify malicious activities such as User to Root (U2R) and Remote to Local (R2L) attacks. For dimension reduction, component analysis and linear discriminant analysis were used. The NSL-KDD dataset was used to perform the entire experiment. To detect suspicious behavior with the two-level classification module, Naive Bayes and the Certainty Factor version of K-Nearest Neighbor were applied. Alrashdi et al. [19] conducted their research on the design of an anomaly detection IDS for Smart city. For the binary classification, they were able to obtain an accuracy rate of 99%. in the classification they applied just one classifier which is Random Forest (RF). In [14], Bakhtiar et al. have applied the J48 algorithm to create an intrusion detection system. Their experience was able to detect only denial of service attacks.

Several research works based on CNN, LSTM have been applied for the detection of anomalies in IOT networks [23, 24]. Monika et al. used a hybrid method which combines these two algorithms for the detection of cyber-attacks on an IoT infrastructure [25]. Diro et al. [21] presented a comparative study of a deep and shallow neural network. The main objective of this work was to detect four classes of attacks and anomalies. For four classes, the system achieved an accuracy of 98.27% for the deep neural network model and an accuracy of 96.75% for the shallow neural network model. A detailed description of a smart home system where security vulnerabilities have been detected by the Dense Random Neural Network (DRNN) deep learning method has been introduced in [22]. They mainly described the denial-of-service attack and the denial of sleep attack in a simple IoT site.

3 Methods and Data Description

3.1 Data

3.1.1 description:

UNSW-NB15 is a network traffic data set with different categories for normal activities and synthetic attack behaviours. The raw network packets of the UNSW-NB15 dataset was created by the IXIA Perfect Storm tool in the Cyber Range Lab of the Australian Centre for Cyber Security (ACCS) for generating a hybrid of real modern normal activities and synthetic contemporary attack behaviours.
-The total number of records: 2,540,044.
-The number of records in the training set:175,341
-The number of records in the testing: 82,332
-Number of features: 49.
-Response Features (attack_class, label).
-Attack class: This dataset has nine types of attacks, namely, Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode and Worms.
-Label: attack or normal behavior.

Table 1. Frequency distribution of attack-class

Attacks-class	Frequency count
normal	56000
Generic	40000
Exploits	33393
Fuzzers	18184
DoS	12264
Reconnaissance	10491
Analysis	2000
Backdoors	1746
Shellcode	1133
Worms	130

Table 2. Frequency distribution of label

Label	Frequency Count
0(normal)	56000
1(attack)	119341

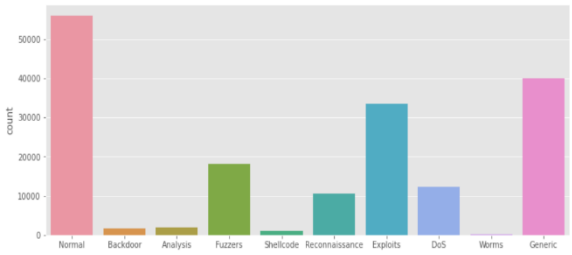

Figure 1: Visualization of the distribution of Attack class

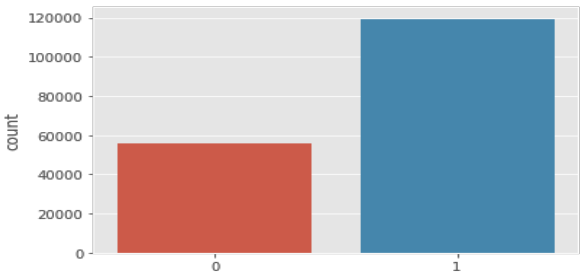

Figure 2: Visualization of the distribution of Label

3.1.2 Data pre-processing:

All machine learning research requires exploratory data analysis and data observation. The first task of this research was to make the dataset feedable by any classifier. For this reason, the first step was therefore to deal with the missing data. In the dataset

First, we need to check if there are any missing values in the dataset. A basic strategy for working with incomplete datasets is to remove entire rows and / or columns with missing values. In fact, there are strategies for imputing missing values [26]. For simplicity, we will reject the four rows with missing values. For feature selection, no machine learning approach has been taken here as this will not have a significant impact on data analysis.

In the characteristic engineering steps, it is necessary to first determine the type of entities in the dataset. The dataset contains categorical and numeric data. Categorical data can be classified into ordinal and nominal values respectively, while numeric data set into discrete and continuous values. The next vital task is to convert the categorical data into vectors. Categorical data can be converted to vectors in several ways. Label Encoder and One Hot Encoder are popular among them.

3.2 Evaluation criteria

Evaluating a model is an essential part of building an effective machine learning model. There are several evaluation metrics The following metrics were calculated to evaluate the performance of the developed system. Using these metrics, one can decide which technique is best suited for this job.
Confusion matrix
The confusion matrix is used to visualize the performance of a technique. It is a table that is often used to describe the performance of a classification model on a set of test data for which the true values are known. It allows for easy identification of confusion between classes. Most of the time, almost all performance measures are computed from it. A confusion matrix is a summary of prediction results on a classification problem. A definition of True Positive (TP), False Positive (FP), False Negative (FN) and True Negative (TN) for multiple classes can be given from confusion matrix. Let C_i be any class out of the eight classes. Following are the definitions of TP, FP, FN, and TN for C_i:
TP (C_i) = All the instances of C_i that are classified as C_i.
FP (C_i) = All the non-Ci instances that are classified as C_i.
FN (C_i) = All the C_i instances that are not classified as C_i.
TN(C_i) = All the non-C_i instances that are not classified as C_i.

Accuracy

The accuracy of a model is only a subset of the model's performance. Accuracy is one of the evaluation metrics of classification models. Eq. (1) represents a single-class accuracy measure.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Precision

Precision means the positive predictive value. It is a measure of the number of true positives the model

claims compared to the number of positives it claims. The precision value for a single class is given in the following equation:

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

Recall

The recall is known as the actual positive rate which means the number of positives in the model claims compared to the actual number of positives there are throughout the data. The recall value for a single class is given in the following equation:

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

F1 score

The F1 score can also measure a model's performance. It is a weighted average of the precision and recall of a model. The F1 Score value for a single class is given in Eq. (4).

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} = \frac{2 \times TP}{2 \times TP + FP + FN} \quad (4)$$

ROC-AUC Curve

AUC (Area Under the Curve) - ROC (Receiver Operating Characteristics) curve is a performance measurement for the classification problems at various threshold settings. ROC is a probability curve and AUC represents the degree or measure of separability. It tells how much the model is capable of distinguishing between classes. The ROC curve is plotted with TPR (True Positive Rate) against the FPR (False Positive Rate) where TPR is on the y-axis and FPR is on the x-axis.

$$FPR = \frac{FP}{TP+FP} \quad (5)$$

$$TPR = \frac{TP}{TP+FP} \quad (6)$$

3.3 Theoretical considerations

Linear Discriminant Analysis

The goal of the LDA technique is to project the original data matrix onto a lower dimensional space. To achieve this goal, three steps had to be carried out. The first step is to calculate the separability between the different classes (ie the distance between the means of the different classes), which is called the inter-class variance or matrix inter-classes S_B. The second step is to calculate the distance between the mean and the samples of each class, which is called the within-class variance or the within-class matrix Sw. The third step is to construct the lower dimensional space which maximizes the variance between classes and minimizes the intra-class variance.

Given the original data matrix $X = x_1, x_2, \ldots, x_N$, where x_i represents the ith sample, and N is the total number of samples. Each sample is represented by M Features ($x_i \in R^M$). In other words, each sample is represented as a point in M-dimensional space.

Assume the data matrix is partitioned into c classes as follows, $X = [\omega_1, \omega_2, \ldots, \omega_c]$, the separation distance between the different classes which is noted $(m_i - m)$ will be calculated as follows:

$$(mi - m)^2 = (\Pi^T \mu_i - \Pi^T \mu)^2 = \Pi^T (\mu_i - \mu)(\mu_i - \mu)^T \Pi \quad (7)$$

where m_i represents the projection of the mean of the ith class and is calculated as follows, $m_i = \Pi^T \mu_i$.
where m is the projection of the total mean of all the classes and it is calculated as follows, $m = \Pi^T \mu$.
Π represents the transformation matrix of LDA.
$\mu_i(1 \times M)$ represents the mean of the ith class and it is calculated as in equation (8), et $\mu(1 \times M)$ is the total mean of all the classes.

$$\mu_j = \frac{1}{n_j} \sum_{x_i \in \omega_j} x_i \quad (8)$$

The term $(\mu_i - \mu)(\mu_i - \mu)^T$ in the equation (7) represents the separation distance between the mean of the ith class $(\mu_i)$and the total mean (μ), or simply it represents the inter-class variance of the ith class (S_{Bi}). Replace S_{Bi} in the equation (7)as follows:

$$(m_i - m)^2 = \Pi^T S_{Bi} \Pi \quad (9)$$

The total inter-class variance is calculated as follows $(S_B = \sum n_i S_{Bi})$

The within-class variance for each class can be calculated as follows,

$$S_{Wj}{}_{j=1,\ldots,c} = \sum (x_{ij} - \mu_j)(x_{ij} - \mu_j)^T = d_j^T * d_j$$

where (x_{ij}) represents the ith sample of the jth class. and (d_j) is centring data of the jth class $d_j = \omega_j - \mu_j = \{x_i\}_{i=1}^{n_j} - \mu_j$. The total intra-class variance represents the sum of all the intra-class matrices of all the classes:

$$S_W = \sum S_{Wi} = \sum_{x_i \in \omega_1} (x_i - \mu_1)(x_i - \mu_1)^T + \sum_{x_i \in \omega_2} (x_i - \mu_2)(x_i - \mu_2)^T + \ldots + \sum_{x_i \in \omega_c} (x_i - \mu_c)(x_i - \mu_c)^T \quad (10)$$

After having calculated the inter-class variance$(S_B)$and the intra-class variance (S_W), the transformation matrix (Π) of the technique LDA can be calculated as in the equation (11), called Fisher criterion, we try to optimize it by maximizing $(S_B)$and minimizing (S_W) .This formula can be reformulated as in the equation (12)

$$\underset{W}{\text{argmax}} \frac{\Pi^T S_B \Pi}{\Pi^T S_W \Pi} \quad (11)$$

$$S_B \Pi = \lambda S_w \quad (12)$$

Support Vector Machine (SVM)

Support Vector Machine is another discriminative model like LR. It is a supervised learning model for analysing the data used for classification, regression, and outliers detection. SVM is most applicable in the

case of Non-Linear data. Given Input X , Class or Label C and LaGrange multipliers a, weight vector ω can be calculated by following equation:

$$\Theta = \sum \alpha_i c_i x_i \quad (13)$$

The target of the SVM is to optimize the following equation:

$$\max_{\alpha_i} \sum \alpha_i - \sum \sum \alpha_i \alpha_j c_i c_j < x_i x_j > \quad (14)$$

In Eq. (14), $< x_i, x_j >$ is a vector which can be obtained by different kernels like polynomial kernel, RBF kernel and Sigmoid Kernel.

We will also implement SVM with the Nystrom approximation which is a low-rang approximation technique to approximate the kernel matrix. This approach is the first time used to detect attacks in the IoT network system.

Decision Tree (DT)

A decision tree is a decision support tool representing a set of choices in the graphic form of a tree. The various possible decisions are located at the ends of the branches (the "leaves" of the tree), and are reached according to decisions taken at each stage.
A DT generally 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. So, from there, it became a tree-like shape; in other words, a flowchart-like structure. Considering a binary tree Fig. 3 where a parent node is split into two children node a left child and a right child. Parent node, left child and right child contains data P_d, LC_d, RC_d , respectively.
Given, features X, impurity measure I(data), the number of samples in parent node P_n , the number of samples in left child LC_n and the number of samples in right child RC_n ; DT's target is to maximize following Information Gain in Eq.17.

$$(P_d, X) = I(P_d) - \frac{LC_n}{P_n} I(LC_d) - \frac{RC_n}{P_n} I(RC_d) \quad (15)$$

Impurity Measure $I(data)$ can be calculated in three techniques Gini Index I_G , Entropy I_H and Classification Error I_E .

Random Forest (RF)

As the name suggests, the random forest algorithm creates the forest with many decision trees. It is a supervised classification algorithm. Many decision trees come together to form a random forest, and it predicts by averaging the predictions from each component tree. It generally has better predictive accuracy than a single decision tree. In general, the more trees there are in the forest, the more robust the forest appears.

Logistic Regression (LR)

LR is a discriminative model which depends on the quality of the dataset. Given the features $\{X = X_1, X_2, X_3, \ldots, X_n\}$ (where, $(X_1 - X_n)$ = Dist inct feat ures), weights $(W = W_1, W_2, W_3, \ldots, W_n)$, bias $(b = b_1, b_2, \ldots, b_n)$ and Classes$(C = c_1, c_2, \ldots, c_n)$ (in our case, we have 10 classes for multi-class classification and) the equation for estimation of posterior is given in following Eq. (16).

$$\left[p(y = C|X; W, b) = \frac{1}{1 + exp(-W^T X - b)}\right] \quad (16)$$

where the predicted Value is P.

4 Experiments and results

4.1 Machine Learning algorithms

We will implement the different machine learning algorithms seen above on our database for the two cases: binary (see table 1.b) and multi-class (see table 1.a).

Table 3 : Evaluation metrics for ML algorithms (multi-class)

Metrics/Methods	LDA	SVM	KSVM	Nystrom-SVM	RF	LR	DT	Adaboost
Accuracy	0.89	0.81	0.79	0.84	0.85	0.84	0.86	0.85
Precision	0.88	0.82	0.73	0.76	0.86	0.82	0.91	0.92
Recall	0.73	0.75	0.76	0.75	0.82	0.74	0.84	0.85
F1	0.76	0.76	0.81	0.86	0.84	0.73	0.86	0.88
AUC_ROC_Score	0.86	0.80	0.82	0.91	0.94	0.92	0.93	0.92

Table 4 : Evaluation metrics for ML algorithms (binary)

Metrics/Methods	LDA	SVM	KSVM	Nystrom-SVM	RF	LR	DT	Adaboost
Accuracy	0.87	0.89	0.84	0.913	0.94	0.88	0.95	0.88
Precision	0.86	0.91	0.82	0.84	0.91	0.84	0.91	0.91
Recall	0.993	0.86	0.87	0.85	0.98	0.86	0.95	0.86
F1	0.84	0.85	0.88	0.89	0.95	0.9	0.96	0.89
AUC_ROC_Score	0.87	0.86	0.86	0.80	0.87	0.95	0.88	0.93

4.2 Resampling data using SMOTE technique

Machine learning algorithms trained with unbalanced data cannot effectively recognize attacks if they are minority data. One way to solve this problem is to use resampling, which adjusts the ratio between the different classes, the lens and to make the data more balanced. we will investigate the influence of

oversampling on the performance of the different classifiers that we use.

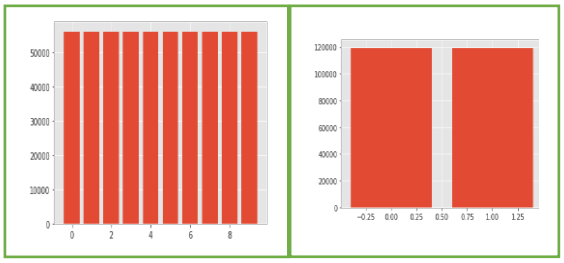

Figure 3 : frequency distribution after resampling data for Attack-class(left) and Label(right)

Results:

Metrics/Methods	LDA	SVM	KSVM	Nystrom-SVM	RF	LR	DT	Adaboost
Accuracy	0.91	0.93	0.78	0.84	0.86	0.87	0.90	0.88
F1	0.80	0.83	0.85	0.87	0.86	0.88	0.87	0.89

Figure 4 : Evaluation models after using SMOTE technique (multi-class)

Metrics/Methods	LDA	SVM	KSVM	Nystrom-SVM	RF	LR	DT	Adaboost
Accuracy	0.94	0.94	0.84	0.95	0.95	0.89	0.95	0.90
F1	0.88	0.87	0.87	0.88	0.96	0.91	0.96	0.87

Figure 5: Evaluation models after using SMOTE technique (binary-class**)**

5 Conclusion

On the basis of this study, it was found that in the multi-class case, LDA, RF, DT techniques gave better performance than others, using this type of data to detect cyber-attacks on IoT network traffic, because they predicted attacks with higher accuracy compared to other approaches. in the binary case the DT, RF techniques and the Nystrom-SVM-which is applied the first time for attacks detection in the IoT network traffic, presented better performance. When we trained our algorithms with balanced data, we noticed a better efficiency in the detection of attacks.
This depth study is necessary to develop a robust detection algorithm. It has been assumed that the network traffic is that of the loT network traffic, because very few IoT network traffic datasets are publicly available. In addition, in the case of streaming data, different issues may arise. More empirical study is needed on this problem focusing on streaming learning.

References:

[1] Abdel-Basset, M., Manogaran, G., Mohamed, M.: Internet of things (IoT) and its impact on supply chain: A framework for building smart, secure and ecient systems. Future Generation Computer Systems 86, 614{628 (2018).

[2] Ahmed, N., De, D., Hussain, I.: Internet of things (IoT) for smart precision agriculture and farming in rural areas. IEEE Internet of Things Journal 5(6), 4890{4899 (2018)

[3] Alansari, Z., Soomro, S., Belgaum, M.R., Shamshirband, S.: The rise of internet of things (IoT) in big healthcare data: Review and open research issues. In: Saeed, K., Chaki, N., Pati, B., Bakshi, S., Mohapatra, D.P. (eds.) Progress in Advanced Computing and Intelligent Engineering. pp. 675{685. Springer Singapore, Singapore (2018)

[4] Arasteh, H., Hosseinnezhad, V., Loia, V., Tommasetti, A., Troisi, O., Shae-khah, M., Siano, P.: Iot-based smart cities: a survey. In: 2016 IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC). pp. 1{6. IEEE (2016)

[5] Hodo, E., Bellekens, X., Hamilton, A., Dubouilh, P.L., Iorkyase, E., Tachtatzis, C., Atkinson, R.: Threat analysis of IoT networks using articial neural network intrusion detection system. In: 2016 International Symposium on Networks, Computers and Communications (ISNCC). pp. 1{6. IEEE (2016)

[6] Nogues, M., Brosset, D., Hindy, H., Bellekens, X., Kermarrec, Y.: Labelled network capture generation for anomaly detection. In: International Symposium on Foundations and Practice of Security. pp. 98{113. Springer (2019)

[7] 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

[8] 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

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

[10] A. Chadd, " DDoS attacks: past, present and future," Network Security, vol. 2018, pp. 13-15, 2018.

[11] Statista . (2019) . Internet of Things (loT) connected devices installed base worldwide from 2015 to 2025 (in billions) [3] Columbus, L. (2018) .

[12] Shakdhe, A., Agrawal, S., & Yang, B. (2019, May) . Security Vulnerabilities in Consumer loT Applications. In 2019 IEEE 5th Inti Conference on Big Data Security on Cloud (BigDataSecurity)(Pp. 16).

[13] Shunnan, M. M., Khrais, R. M., & Yateem, A. A. (2019, December). loT Denial-of-Service Attack Detection and Prevention Using Hybrid IDS. In 2019 International Arab Conference on Information Technology (ACIT) (pp. 252-254) . IEEE.

[14] Alrashdi, I., Alqa zzaz, A., Aloufi, E., Alharthi, R., Zohdy, M., & Ming , H. (2019 , January). AD-loT: anomaly detect ion of loT cyberattacks in smart city using machine learning. In 2019 IEEE 9th Annual Computing and Communication

[15] E. Anthi, L. Williams, P. Burnap, Pulse: an adaptive intrusion detection for the internet of things (2018).

[16] A. Ukil , S. Bandyoapdhyay , C. Puri , A. Pal , Iot healthcare analytics: The importance of anomaly detection, in: Proceedings of the 2016 IEEE 30th

[17] 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) .

[18] G. D'Angelo , F. Palmieri , M. Ficco , S. Rampone , An uncertainty-managing batch relevance-based approach to network anomaly detection, Appl. Soft Comput. 36 (2015) 408–418 .

[19] Alrashdi, I., Alqa zzaz, A., Aloufi, E., Alharthi, R., Zohdy, M., & Ming , H. (2019 , January). AD-loT: anomaly detect ion of loT cyberattacks in smart city using machine learning. In 2019 IEEE 9th Annual Computing and Communication Work shop and Conference (CCWC) (pp. 0305-0310). IEEE.

[20] Bakhtiar, F. A., Pramukantoro, E. S., & Nihri , H. (2019, March). A Lightweight IDS Based on J48 Algorithm for Detecting DoS Attacks on loT Middleware. In 2019 IEEE 1st Global Conference on Life Sciences and Technologies (LifeTech) (pp. 41-42). IEEE .

[21] A .A . Diro , N. Chilamkurti , Distributed attack detection scheme using deep learning approach for internet of things, Future Gen. Comput. Syst. 82 (2018) 761–768 .

[22] 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 .

[23] Yanmiao Li, Yingying Xu, Zhi Liu, Haixia Hou, Yushuo Zheng, Yang Xin, Yuefeng Zhao, Lizhen Cui, Robust detection for network intrusion of industrial IoT based on multi-CNN fusion, Measurement, Volume 154, 2020, 107450, ISSN 0263-2241,

[24] D. Wu, Z. Jiang, X. Xie, X. Wei, W. Yu and R. Li, " LSTM Learning With Bayesian and Gaussian Processing for Anomaly Detection in Industrial IoT," in IEEE Transactions on Industrial Informatics, vol. 16, no. 8, pp. 5244-5253, Aug. 2020, doi: 10.1109/TII.2019.2952917.

[25] M. Roopak, G. Yun Tian and J. Chambers, " Deep Learning Models for Cyber Security in IoT Networks," 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), 2019, pp. 0452-0457, doi: 10.1109/CCWC.2019.8666588.

[26] Stef van Buuren, Karin Groothuis-Oudshoorn (2011). "Mice: Multivariate Imputation by Chained Equations in R". Journal of Statistical Software 45: 1-67.