International Conference on Advances in Materials, Computing and Communication Technologies
In Association with International Journal of Scientific Research in Science and Technology
Volume 9 | Issue 1 | Print ISSN: 2395-6011 | Online ISSN: 2395-602X (www.ijsrst.com)

Novel Intrusion Detection Techniques for IOT Enabled Smart Cities

Ms.V. Sobana[1], Dr. P. Krishna Kumar[2]
[1]PG Student, Department of CSE, VV College of Engineering, Tirunelveli, Tamil Nadu, India
[2]Professor, Department of CSE, VV College of Engineering, Tirunelveli, Tamil Nadu, India

ABSTRACT

Internet of Things for sustainable resource management critical to safeguard the future network infrastructure from intruders. With the growth of connected things, the most-widely used centralized (cloud-based) IDS often suffers from high latency and network overhead, thereby resulting in unresponsiveness to attacks and slow detection of malicious users. In this paper, the ML models to detect the various attacks accurately. To develop parallel machine-learning models corresponding to a partitioned attack dataset. In the distributed case, the parallel models individually perform both the feature selection and multi-layer perceptron classification. The effectiveness of the proposed architecture by using machine learning algorithms SVM and NB to achieve the high accuracy and lowest building time performance.

Keywords- Internet of Things, Intrusion Detection System, Multi-Layer Perceptron, Suport Vector Machine, Navie Bayes, Machine Learning

I. INTRODUCTION

to Since the past decade, the interconnection among humans, machines and services has grown significantly, resulting into a new communication paradigm of Internet of Things (IoT). Introduction the current generation (4G/5G) and future generation (6G and beyond) networks, which will have a significant position in the next generation of sustainable smart cities. The IoT has exemplary uses cases in many areas of transportation, healthcare, retail, industry and education and will continuously expand across different directions. Part of this remarkable growth of connected things can be credited towards wireless technologies in recent years as a key enabler, seamlessly connecting humans to physical objects through phone, tablet and personal computer interfaces. Until the year of 2020, we expect that wireless transmission contributes to two third of the overall internet data with cellular/Wi-Fi connections sharing 66% of the overall internet protocol (IP) data. A large number of attackers or assailants have been trying to steal personal information of target users or intend to gain unauthorized access to the target's resources or applications by attacking the vulnerabilities of wireless networks. The openness of wireless networks means that it is highly feasible to intercept and capture the packets from particular data traffic. To mitigate this issue and ensure secured communications among all clients and servers, several wireless network security mechanisms have been employed in the wireless network of local area context, including the Internet Control Message Protocol (ICMP), User Datagram Protocol (UDP) and Transmission Control Protocol (TCP). IDS is a security protection method used for uncovering suspicious actions in the system and timely intercepting the attacking source in order to safeguard the network. Based on the types of available cyber data, IDS may be categorized into host and network-based detection. Host-based corresponds to detection inside the electronic devices such as smartphones or laptops, which specifically watches in-host resources such as logs, disk resources and file systems. An example of host-based detection is antivirus. Network- based detection performs detection by analyzing the network traffic between electronic devices and internet. In this work, we focus our attention to the network-based IDS for monitoring malicious traffic in an IoT network that is constructed by fog and edge devices. A properly designed network-based IDS should detect various attacks accurately. To achieved higher accuracy and the CPU time will be significantly reduced.

In order to deal with these shortcomings of centralized IDS, this work aims at contributing to the following.

- With the proposed IDS architectures, we enable leveraging of extensive computational tasks from a single powerful unit into a number of less powerful units running in parallel. This is expected to make the IDS model to be attractive for deployment in a typical IoT network comprising a large number of edge devices and several fog devices.

Our approach in designing IDS architectures relies upon combining the best selection of feature extraction and selection methods prior to classification. Our aim is to individually develop parallel machine-learning models corresponding to a partitioned attack dataset. Whilst data preprocessing and feature extraction (stacked auto encoder-based) are common processes, the subsequent operation for computational task distribution is diverging as distinguished below.

- For the distributed approach, the parallel models individually perform both the feature selection and MLP classification. This will then be followed up by combining the parallel outputs undertaken by a coordinating edge or fog for final decision making.

On the other hand, the distributed method achieves a higher accuracy with faster processing due to individual classification and processing. Based on analyzing the Aegean Wireless Intrusion Detection (AWID) cyber-attack dataset that provide logs of both 802.11 normal and attack traffic record, numerical results confirm this intuition and demonstrate comparable detection accuracy of both methods with the superior centralized IDS. Furthermore, the distributed approach has a desirable result of being able to build the computational model at 2.5 times faster than the semi-distributed case.

II. LITERATURE REVIEW

Secure IoT for sustainable smart cities, IDS on IoT, feature selection method and classification, deep learning detection, and other reviews of IDS. The details of these categories are explained in the following

A. IoT for Sustainable Smart Cities:

The advancement in IoT has paved the way for managing constrained resources in the smart cities for sustainable living. In the case of electricity supply, IoT-enabled energy management could adopt the framework of a smart grid incorporating green, renewable energy sources for powering up the society. In terms of city's infrastructures, a wide-range of IoT-driven sensors can help monitor energy consumption of street lighting and building as well as acquire and optimize traffic pattern in which the data can be exploited to provide more effective utilization of energy sources, including electricity, heat and fuel. For environmental protection, IoT can assist in managing the household and industrial wastes, enabling rapid recycling, decomposition and transformation of residue materials into more useful products. In another domain, IoT can be beneficial to improve the quality of life for a sustainable society. For instance, IoT with sensing and embedded microcontrollers can be deployed to monitor the urban air quality and water quality, which in turn advocates proper intervention measures by the authorities.

The technological convergence of IoT supporting infrastructure and diverse use-cases means that there will be unprecedented growth of seamless integration of electronics (next- generation processors and memory), networking (5G- and 6G-based solutions) and artificial intelligence (AI). In fact, the vast adoption of AI for both end-user applications and future communication design can be a key for unlocking the potential of 6G networks.

The development of IDS with embedded intelligence to depart away from "one-off" security protection and incorporate a sophisticated means of continuous learning from evolving network data.

B. Smart Cities

IoT has paved the way for managing constrained resources in the smart cities for sustainable living. IoT-enabled energy management could adopt the framework of a smart grid incorporating green, renewable energy sources for powering up the society. IoT within future 6G networks (as well as 5G networks in the immediate future) for improving sustainability in the cities is the ability to provide reliable and secure means of data exchange. The development of IDS with embedded intelligence to depart away from "one- off" security protection and incorporate a sophisticated means of continuous learning from evolving network data. The vast adoption of AI for both end-user applications and future communication design can be a key for unlocking the potential of 6G network. The proposed technique can provide scalability, interoperability and flexibility since it is a distributed IDS. As a result, the solution may achieve better accuracy, a lowered down false alarm rate and quicker response time, but it has to rely on a cloud server for summarization. Once the cloud server is down, it will present an issue.

C. IDS for IoT

The proposed technique was shown to outperform the other benchmark methods, it may be unsuitable for other IoT protocols. Other than that, with an advanced decentralized computing structure called fog-computing as an IoT framework, the authors in proposed an Intrusion Detection/Prevention System (IDPS) with fog- assisted software defined networking (SDN).They proposed an effective algorithm to allocate resources in order to deal with an IoT scalability issue. Furthermore, they also proposed architecture to detect anomalies and address cyber threats at the edge of the IoT network, and examined four classifiers to detect intrusions. The proposed technique of combining classifiers achieved a good performance in many different scenarios, but particular performances of recursive neural network (RNN) and MLP appear to be fluctuating, sensitive to the choice of the learning window. Whereas, other authors developed a fog-oriented IDS utilizing Online Sequential Extreme Learning Machine (OS- ELM). Fog-nodes will identify the malicious traffic from the IoT environment and subsequently send the detected intrusion to a cloud server for summarization. The proposed technique can provide scalability, interoperability and flexibility since it is a distributed IDS. As a result, the solution may achieve better accuracy and quicker response time, but it has to rely on a cloud server for summarization.

D. Feature Selection Method and Classification

A lightweight IDS by utilizing a support vector machine (SVM)-based classier with three kernel functions. They stated that in that work, they only consider the types of attacks that will influence the traffic intensity. Therefore, they proposed an IDS that solely considers an attribute given by the packet arrival rate. The limitation of this work is given by the inability to recognize intrusions without accompanying impact on the intensity of traffic. This approach is only tested for denial-of- service (DoS), R2L, U2r, and Probe attacks, and it is a centralized approach. The AWID dataset and tested different classification algorithms. The result showed that the impersonation attacks seem to have a less satisfactory result or detection rate. The proposed machine learning-based IDS using SVM and NB as an e their experiment with two scenarios being selected to evaluate their performance. They used the KDDCUP99 dataset to selection classifier ranked the feature, and only the highest-ranking feature would be selected to classify the class of the dataset. Two classification algorithms namely, Support Vector Machine and Naïve Bayes, were employed to classify the class in the dataset. Herein the SVM classifier obtained a detection rate of 86% correctly classified. As a result, the proposed SVM technique is faster and has higher accuracy than the NB algorithm.

E. Review on IDS

This work concentrated on discussing potential solutions from the aspects of architecture types, IoT standard and technologies, review of security threats and practices for IoT. From a different perspective, also provided a comprehensive review of NIDS in terms of IoT security threats, traditional IDS techniques, tools and datasets for implementation, state-of-the-art for IoT and different machine learning techniques. This work is useful for recognizing a variety of IoT threats and as a reference for proposing NIDS techniques for IoT networks. Proposed a collaborative intrusion detection framework (COLIDE) by collaborating end-hosts/nodes locally with edge routers globally as well as collaborating host and network-based intrusion detection. The focus of this work is on testing for efficiency but not the detection rate and time. The distributed approach will perform classification independently with its selected features. If the IoT network, they can consider using a distributed approach. Comparison of existing works and our proposed technique is discussed and summarized in Tables1. From this tables, it can be noticed that the most of the classifiers exploit SVM and NB algorithms for feature selection. In this work, we aim to consider SAE for feature extraction and SVM and NB as the feature selection classifier because of their effectiveness. From the existing works, most papers are using KDDCUP dataset for conducting experiments. It is an older dataset in detecting DOS, R2L, U2R, and probing attacks while the dataset that will be chosen for this work is the AWID dataset, which is the most recent one that captures a wireless network is detecting various attacks accurately. The proposed technique named DFES, which achieved high accuracy in detecting impersonation attacks at the expense of taking a long CPU time. However, the reduced CPU time compared to D-FES but the accuracy is reduced as well. However, the proposed technique is expected to achieve higher accuracy and the CPU time will be significantly reduced for counteracting impersonation attacks due to its distributed nature.

Table 1: Comparison of Existing Works and Proposed Technique (Part 1)

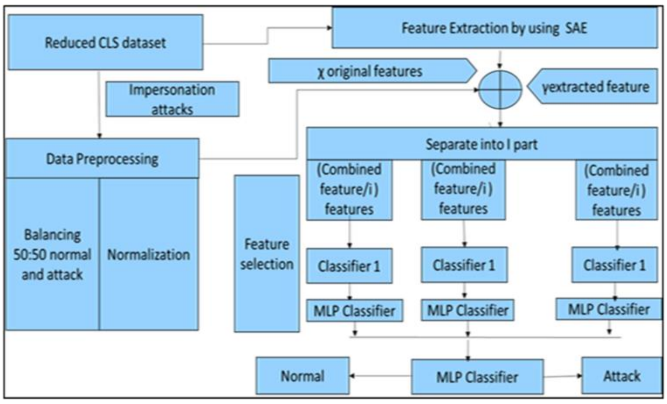
S. No	Proposed Technique	Classifier	Dataset	Advantages	Disadvantages
1	AWID Dataset	Adaboost, Hyperpipes, J48, NB, OneR, RF, RT, ZeroR	AWID	Introduce AWID dataset and good accuracy in injection attack	Lowest detection rate of impersonation attack

2	D-FES	SVM, ANN, C4.5, SAE	AWID	High accuracy (impersonation attack)	Long CPU time
3	DEMISe	(RBFC),C4.8,RF, MLP,SVM,LR, MI, Theoretic Feature Selection Filter	AWID	Reduced time to build time compared with D-FES	Accuracy is lower than D- FES

III. RESEARCH METHODOLOGY

In distributed approach χ represents the quantity of original features while y denotes the quantity of extracted features. Moreover, i denotes the number of partitions to be performed, and M is the number of features to be selected. In this work, the original dataset has 154 features. To introduce a distributed parallelism, we partition the dataset into three parts (i = 3) and we select relevant (top) M = 7 features. In this case, each partitioned sub-dataset consists of 204/3 = 68 features. For example, the first partition of the dataset (dataset A) consists of feature 1 to feature 68, the second partition of the dataset (dataset B) consists of feature 69 to feature 136, and the third feature selection, as depicted in Figure 2. The most accurate technique for each dataset is selected and sent to a decision-maker or a controller to decide whether this network data corresponds to an attack or not.The overall accuracy is given by the average accuracy across the three different branches of the model.

Figure 1: Procedures for the distributed approach



All data contained in the reduced CLS dataset should be transformed into numerical values before performing normalization for deep learning purposes. The dataset needed to be balanced into a 1:1 ratio of normal and attack classes. Normalization was then conducted, and feature extraction was utilized to construct and extract new useful features. The MLP classifier was selected as the classifier for the distributed approach.

A. Data Preprocessing

Data preprocessing is a machine learning technique that involves transforming raw data into an understandable format. Here the attack name and event is transformed into machine readable format. The AWID dataset contains two types of files,namely classes (CLS) and attacks (ATK). The CLS part has four categories,i.e., normal, dos,r2l,u2r and probe. ATK and CLShave two versions, namely the full and reduced versions. In this study, the reduced version was selected due to complexity consideration. The reduced CLS dataset contains more than 1.7 million instances and 154 features. Two attack categories in the CLS were not considered in this study because the focuses are only on the impersonation attack and normal network data.

$$X\ Scaled = \frac{X-Xmin}{Xmax-Xmin} \qquad (1)$$

Finally, the data were separated into training and testing sets with a ratio of 70:30.

B. Feature Extraction

Feature extraction extracts useful features from the original features for accuracy enhancement in classification and efficiency improvement of the classifier .A complete AE architecture consists of the encoding and decoding functionalities. The encoder is used to compress/extract the input neurons (X) into a representation of data (Y). The pre-training process trains a single AE utilizing one hidden layer. Furthermore, every AE can be individually trained, which will then be combined to construct a SAE. As illustrated in Figure 1, each neuron in the first encoder layer becomes the feature extractor, which is further used for training by the next AE. A 154:100:50 SAE architecture is selected due to its better performance. The second encoder layer's 50 neurons would become the extracted features and then be used together with the 154 original features.The encoder can be defined in the form of an encoder function as expressed in

Eq. 2, i.e.,

$$y = sf\,(Wx + bf\,) \qquad (2)$$

where x is the input, y is the output or extracted features of the encoder, sf is the nonlinear activation function to determine the necessity of features, W is the weight of each neuron and bf is the bias value for encoding. From the other end, below is the Eq. 3 for the decoder to reconstruct the original input,

$$z = sg(V{:}y + bg)\,(3)$$

where y is the output of the encoder, z is the reconstructed value, sg is the activation function of decoder, V is the weight of the neuron, and bg is the bias value for decoding.

C. Dataset Collection

A collection database for machine learning oriented is available with the UCI Machine Learning Repository which will be available from internet and it is open source. The data set are accommodated and preserved in the center for Machine Learning and Intelligent Systems in the University of California, Irvine. Each dataset contains separate webpage that presents the entire facts about its inclusion and any pertinent research that is examining it.

The datasets from the internet will take up the format of ASCII files, frequently the helpful CSV arrangement.

ID	Recording	Stat us	Gender	Jitter_rel	Jitter_abs	Jitter_RAP
CONT- 01	1	0	1	0.25546	0.000014581	0.001467
CONT- 01	2	0	1	0.36964	0.000021662	0.0019317
CONT- 01	3	0	1	0.23514	0.000013109	0.0013527
CONT- 02	1	0	0	0.2932	0.000017331	0.0011048
CONT- 02	2	0	0	0.23075	0.000014561	0.0010729
CONT- 02	3	0	0	0.16489	0.000010011	0.00081887
CONT- 03	1	0	1	0.22506	0.000014288	0.0013581

Table 2. Input Data

Id	Duration	protocol type	service	flag	src_bytes	dst_bytes
202	0	Icmp	25	4	0	0
768	0	Icmp	25	2	312	1856
3753	0	Icmp	25	2	245	2058
20087	0	Icmp	25	2	298	1267
16159	0	Icmp	20	2	740	0
18012	0	Icmp	25	2	304	16414

Table 3.Output Data

After fetching the data, the preprocessing is done to change the raw data to the machine readable form.

D. Feature Selection

Irrelevant features increase the computation time and reduce the prediction accuracy at the same time. Thus, feature selection is commonly utilized for enhancing the accuracy of prediction by removing noisy/less relevant teristics of various feature selection methods, the wrapper- based and filter-based methods were used for feature selection. The earlier method examines a subset of features that are generated any learning algorithm and has a good accuracy whilst computationally intensive. For the distributed method, SVM and NB were selected.

The filter-based method uses information / consistency / correlation measures to compute the relevance of a feature subset, has relatively fast computation at a reasonable accuracy and less computationally intensive than the wrapper one due to the absence of learning algorithm execution.

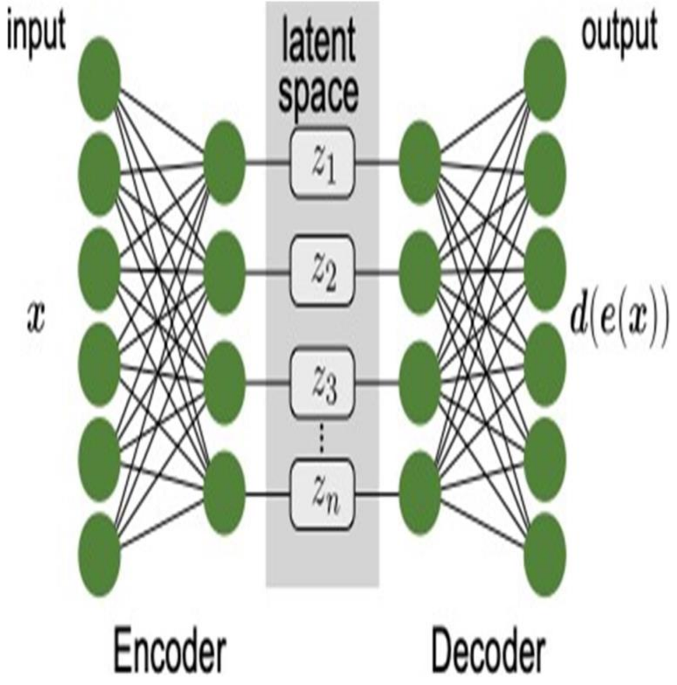
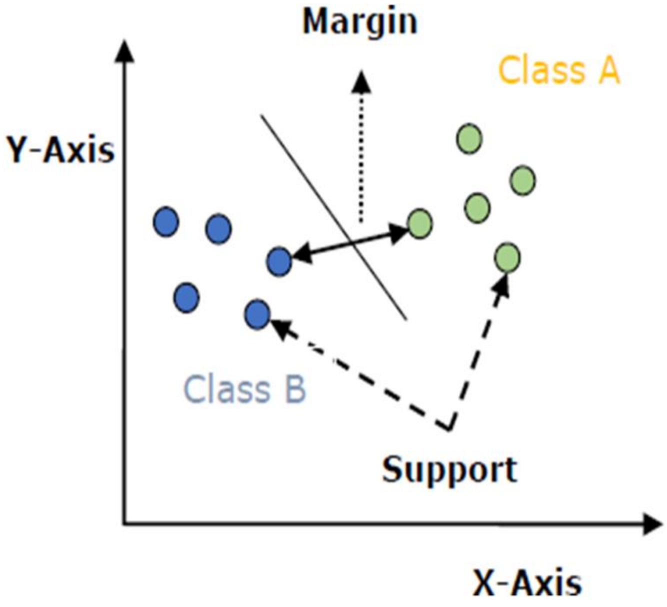


Figure 2: SAE architecture

A. SUPPORT VECTOR MACHINE (SVM)

Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for classification or regression challenges. However, it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate.



Algorithm

In this we present the QP formulation for v,s classification. This is a simple representation only.

SVM classification:

min α 1

α α y y K(x , x)

i

αi i1

2 i1

i j i j i j

j1

0 i⩽C, for all i; (4)

Variables i are called slack variables and they measure the error made at point (xi, yi). Training SVM becomes quite challenging when the number of training points is large. A number of methods for fast SVM training have been proposed.

B. NAIVE BAYES

The Naive Bayesian classifier is based on Bayes’ theorem with independence assumptions between predictors. A Naive Bayesian model is easy to build, with no complicated iterative parameter estimation which makes it particularly useful for very large datasets. Despite its simplicity, the Naive Bayesian classifier often does surprisingly well and is widely used because it often outperforms more sophisticated classification methods.

Algorithm

Bayes theorem provides a way of calculating the posterior probability, P (c|x), from P(c), P(x), and P (x|c). Naive Bayes classifier assume that the effect of the value of a predictor (x) on a given class (c) is independent of the values of other predictors. This assumption is called class condit ional independence.

$P(c|x) = P(x|c)P(c)/P(x)$ (5) P (c\X) = P (x1\c) x P (x2\c) x … x P (xn\c) x P (c)

Naive Bayes classifier calculates the probability of an event in the following steps:

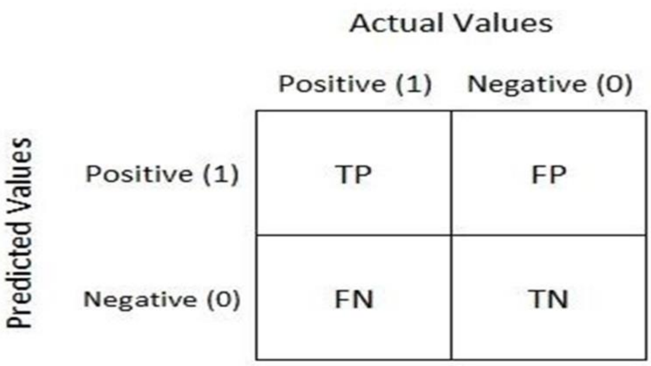
Step 1: Calculate the prior probability for given class labels. Step 2: Find Likelihood probability with each attribute for each class.

Step 3: Put these value in Bayes Formula and calculate posterior probability.

Step 4: See which class has a higher probability, given the input belongs to the higher probability class.

IV. PERFORMANCE MEASURES

The performance evaluation of proposed Neural Networks are simulated using PYTHON under windows environment. The implementation of this framework is performed on lung cancer dataset obtained from the UCI machine learning repository site. The attack dataset given as input to the neural network and the data is divided into training data and test data. The training set for the neural network consists of 70% of the total dataset and the testing set is 30% of the total data. The proposed method is effectively compared with Support Vector Machine algorithm, Naïve Bayesian algorithm in terms of performance metrics obtained from confusion matrix shown in table 1.



		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Table 1. Confusion Matrix Table

Analysis

Here several performance metrics are used to check the segmentation. Segmentation results and ground truth are compared to evaluate the performance.

a. Accuracy
b. Precision
c. Recall

a. Accuracy

This means as many times the different samples are tested with the same algorithm and the machine or system provides results how much accurate. The accuracy is the proportion of true results (both true positive and true negative) in the total data.

Accuracy (A) = (TP+TN) / (TP + TN + FP+FN)

b. Precision (or) Specificity

Precision effectively describes the purity of positive detections relative to the ground truth.

Precision= TP/TP+FP

c. Recall (or) Sensitivity

Recall effectively describes the completeness of positive predictions relative to the ground truth.

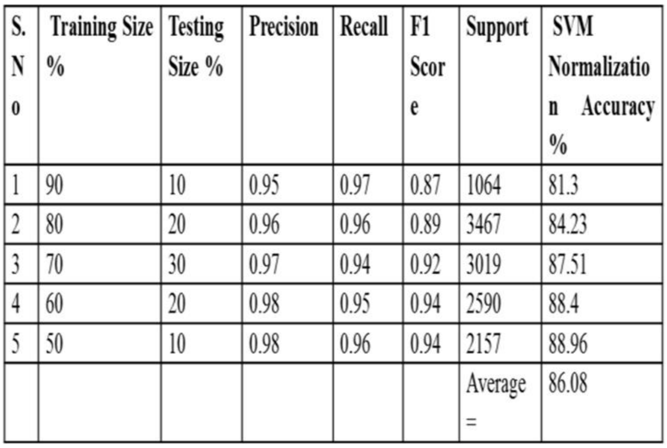
Recall(R) or Sensitivity= TP (TP+FN).

V. RESULT AND DISCUSSION

The Proposed system was investigated to address the impersonation attack and the background process will be explained in details in this section. The proposed idea was tested, and the result is showed in the last part of this section. In this experiment, the tools used for evaluation were AWID dataset while the hardware was 2.5 GHZ,AMD PRO A4-4350BR4, 64 bit OS with 1

TB HDD,DVD Drive,4GB of RAM. In order to validate the effectiveness of our IDS proposal, we carried out numerical experiments as described above for proposed techniques, namely the distributed cases.

Table 4. SVM Normalization Accuracy %



S. N o	Training Size %	Testing Size %	Precision	Recall	F1 Scor e	Support	SVM Normalizatio n Accuracy %
1	90	10	0.95	0.97	0.87	1064	81.3
2	80	20	0.96	0.96	0.89	3467	84.23
3	70	30	0.97	0.94	0.92	3019	87.51
4	60	20	0.98	0.95	0.94	2590	88.4
5	50	10	0.98	0.96	0.94	2157	88.96
						Average =	86.08

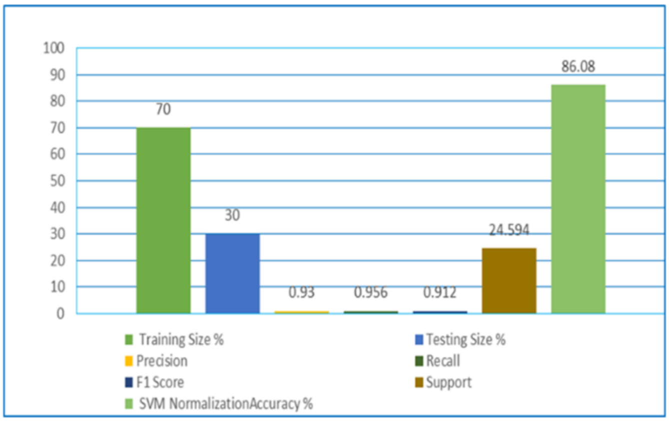
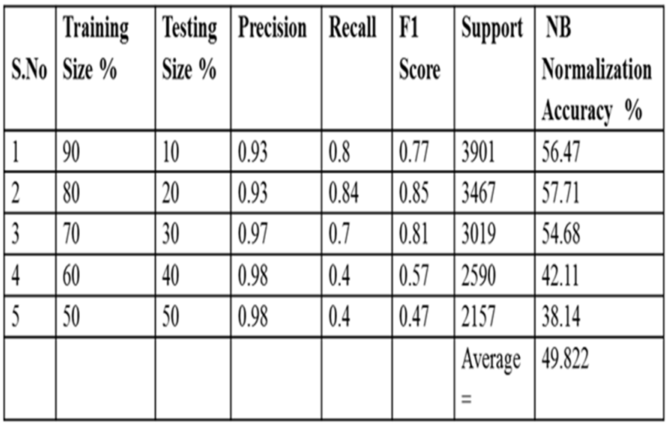


Figure 3. SVM Normalization Accuracy %

After normalization SVM algorithm could achieve the detection accuracy (86.08).



S.No	Training Size %	Testing Size %	Precision	Recall	F1 Score	Support	NB Normalization Accuracy %
1	90	10	0.93	0.8	0.77	3901	56.47
2	80	20	0.93	0.84	0.85	3467	57.71
3	70	30	0.97	0.7	0.81	3019	54.68
4	60	40	0.98	0.4	0.57	2590	42.11
5	50	50	0.98	0.4	0.47	2157	38.14
						Average =	49.822

Table 5. NB Normalization Accuracy %

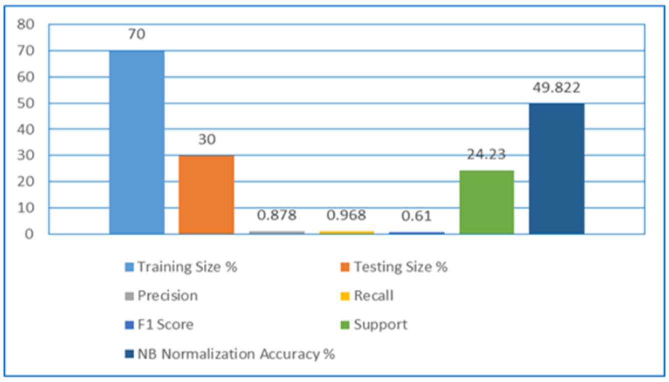


Figure 4. NB Normalization Accuracy %

After normalization NB produces the detection accuracy (49.822).

VI. CONCLUSION

Machine learning techniques should detect the various attacks accurately. The AWID dataset, contains two types of files, namely classes (CLS) and attacks (ATK). The CLS part has four categories, i.e., normal, dos, probe, r2l and u2r. The effectiveness of the proposed architecture by using machine learning algorithms SVM and NB. In the distributed approach, SVM algorithm could achieve the lowest CPU time to build the model (73.52 seconds) with the highest detection accuracy (86.08).NB produces lowest detection (49.822). From the experimental result, the SVM algorithm produces the highest accuracy of the proposed system is (86.08), Precision (0.978), Recall (0.988) and F1 Score (0.88).

In future the accuracy can be improved in distributed method. The distributed approach appears to be the fastest approach to detect malicious attacks.

VII.REFERENCES

[1]. S. Din, A. Paul, W.-H. Hong, H. Seo, Constrained application for mobility management using embedded devices in the internet of things based urban planning in smart cities, Sustainable Cities and Society 44 (2019) 144 {151}.

[2]. P.K.Khatua,V.K.Ramachandaramurthy, P.Kasinathan, J.Y.Yong, J.Pasupuleti, A. Rajagopalan, Application and assessment on internet of things toward the sustainability of energy systems: Challenges andissues, Sustainable Cities and Society 53(2020).

[3]. B. N. Silva, M. Khan, K. Han, Towards sustainable smart cities: A review of trends, architectures, components, and open challenges in smart cities, Sustainable Cities and Society 38 (2018) .

[4]. M. E. Aminanto, R. Choi, H. C. Tanuwidjaja, P. D. Yoo, K. Kim, Deep abstraction and weighted feature selection for Wi-Fi impersonation detection, IEEE Transactions on Information Forensics and Security 13 (3) (2018) .

[5]. A. A. Aryachandra, Y. F. Arif, S. N. Anggis, Intrusion detection System (ids) server placement analysis in cloud computing, in: 2016 4th International Conference on Information and Communication Technology (ICoICT), 2016.

[6]. L. Tian, Design and implementation of a distributed intelligent network intrusion detection system, in: 2010 International Conference on Electrical and Control Engineering, 2010.

[7]. C.Kolias, G. Kambourakis, A. Stavrou, S. Gritzalis, Intrusion detection in 802.11 networks: Empirical evaluation of threats and a public dataset, IEEE Communications Surveys & Tutorials 18 (1) (2016).

[8]. M. Abujubbeh, F. Al-Turjman, M. Fahrioglu, Software-defined wireless sensor networks in smart grids: An overview, Sustainable Cities and Society 51 (2019).

[9]. M. M. Rathore, A. Paul, W.-H. Hong, H. Seo, I. Awan, S.Saeed,Exploiting iot and big data analytics: Dening smart digital city using real time urban data, Sustainable Cities and Society 40(2018).

[10]. J. Loy-Benitez, Q. Li, K. Nam, C. Yoo, Sustainable subway indoor air quality monitoring and fault-tolerant ventilation control using a sparse auto encoder-driven sensor self-validation, Sustainable Cities and Society 52 (2020).

[11]. B. Nvs, P. Saranya, Chapter 18 - water pollutants monitoring based on internet of things, in: P. Devi, P. Singh, S. K. Kansal (Eds.), Inorganic Pollutants in Water, Elsevier, 2020.

[12]. R. Ande, B. Adebisi, M. Hammoudeh, J. Saleem, Internet of things: Evolution and technologies from a security perspective, Sustainable Cities and Society (2019) .

[13]. A. Chehri, H. T. Mouftah, Autonomous vehicles in the sustainable cities, the beginning of a green adventure, Sustainable Cities and Society 51 (2019) 101751.

[14]. M. A. Rahman, M. Y. Mukta, A. Yousuf, A. T. Asyhari, M. Z. A. Bhuiyan, C.Y. Yaakub, Iot based hybrid green energy drive lighting system, in: 2019 IEEE Intl Conf on Dependable, Autonomic and Secure Computing (DASC), 2019.

[15]. B. T. N. Moustafa, K. R. Choo, An ensemble intrusion detection technique based on proposed statistical flow features for protecting

network traffic of internet of things, in: IEEE Internet of Things Journal, Vol. 6, 2019.

[16]. K. S. S. Prabavathy, S. M. Shalinie, Design of cognitive fog computing for intrusion detection in internet of things, in: Journal of Communications and Networks, Vol. 20, 2018.

[17]. M. S. G. T. E. Anthi, L. Williams, P. Burnap, A supervised intrusion detection system for smart home iot devices, in: in IEEE Internet of Things Journal, Vol. 6, 2019.

[18]. V. S. S. U. Jan, S. Ahmed, I. Koo, Toward a lightweight intrusion Detection system for the internet of things, in: IEEE Access, Vol. 7, 2019.

[19]. P. Li, Y. Zhang, A novel intrusion detection method for internet of things, in: 2019 Chinese Control And Decision Conference (CCDC), 2019.