The Seybold Report

ISSN 1533-9211

DOI 10.5281/zenodo.6601187

ANALYSIS OF PERFORMANCE COMPARISON OF INTRUSION DETECTION SYSTEM BETWEEN SVM, NAÏVE BAYES MODEL, RANDOM FOREST, K-NEAREST NEIGHBOR ALGORITHM

[1]HARSH NAMDEV BHOR and [2]Dr. MUKESH KALLA

[1]Assistant Professor, K.J. Somaiya Institute of Engineering and Information Technology, Sion, Mumbai, India.
[2]Assistant Professor, Sir Padampat Singhania University, Bhatewar, Udaipur, Rajasthan, India

ABSTRACT -The computerized unrest has generously transformed ourselves in which Internet-of-Things (IoT) assumes a noticeable job. The quick improvement of IoT to most corners of life, nonetheless, prompts different arising online protection dangers. Consequently, recognizing and forestalling likely assaults in IoT networks have as of late pulled in vital premium from both scholarly world and industry. The development of the Internet of Things (IoT), distinctive IoT hubs, for example, 6LoWPAN gadgets can be associated as an organization to offer incorporated types of assistance. Since security and intrusion detection are becoming crucial among IoT devices, real-time detection of the attacks are critical to protect the IoT networks. However, there exists limited research for efficient network intrusion detection systems (NIDS) in the IoT networks. . However, there exists limited research for efficient network intrusion detection systems (NIDS) in the IoT networks. This paper therefore proposes a new NIDS protocol with an efficient replica detection algorithm to increase the utility and performance of existing NIDS, where a number of replica test nodes are intentionally inserted into the network to test the reliability and response of witness nodes. The proposed protocol, Enhanced NIDS, can address the vulnerability of NIDS and improve IoT network security to detect severe compromise attacks such as clone attacks. The simulation study shows that compared to the state-of-the-art SVELTE protocol, the proposed protocol can significantly increase the detection probability and reduce the energy consumption for detecting clone attacks in IoT networks. The aim of the research is As upcoming Phase, with design of proposed system focus on the going with issues, To analyze strong and powerless reasons for different area techniques IoT, to extend the assault disclosure reach to deliver more IoT advances to improve security of ready traffic and the heads and to develop advantageous solicitations, for instance, mindful association then autonomic organization structures.

KEYWORDS: Internet of Things, Network protocol, Security, Intrusion detection systems, Replica detection, Clone attacks.

1. INTRODUCTION

Nowadays, Internet of Things (IoT) have been used in a variety of critical domains and applications such as in energy , transportation and healthcare . However, achieving security in IoT is very challenging. For example, among the IoT communication technologies, IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN) is considered as one of the best protocols for realizing the IoT networks and applications . This is because 6LoWPAN communication is based on IPv6, which makes it possible to create intelligent services that could not be realized before, when the lowpower networks were closed and disconnected from the Internet . 6LoWPAN usually confront threats from attackers who intend to extract

The Seybold Report
ISSN 1533-9211
DOI 10.5281/zenodo.6601187

information from the IoT devices or disable certain functions of the IoT to achieve purpose of attackers . One of the well-known and severe attacks IoT networks is the clone attack, also called node replication attack . In this attack, an adversary captures a few nodes, replicates them and then deploys the replicas throughout the network. If these replicas are not detected, the network will be vulnerable to severe internal attacks . Considering a scenario of surveillance on the battlefield, the adversary now can overhear the traffic passing the replicas, which may contain the aforementioned locations of soldiers, inject false data into the network, which can be false commands, deactivate other nodes and even manipulate legitimate nodes. Therefore, it is critical to ensure the security and intrusion detection of IoT networks . Once an IoT device is compromised, an adversary can launch clone attacks by replicating the compromised node, distributing the clones throughout the network, and starting a variety of inside attacks. In order to tackle the clone attacks, network intrusion detection systems (NIDS) have been developed to detect malicious activities such as clone attacks, denial-of-service attacks, and replay attacks by monitoring the network . One commonly used NIDS method to detect clone attacks is the identity and distance verification. For example, in 6LoWPAN networks, NIDS is deployed at a system level or at the edge networks to analyze the Internet packets and events (inbound and outbound) to identify malicious activities, and take actions according to the security policies of the network.

2. LITERATURE REVIEW

Novakovic et al [1] shows that each feature selection algorithm produces a different performance in different implementation cases.

Zhou, Liang et al. [2] proposed a framework that utilizes a profound neural organization model for the order of cyberattacks. The framework depends on three stages: 1) Data procurement 2) Data pre-preparing. 3) Deep neural organization order. They utilized worldwide optical boundaries to accomplish elite exactness approx. 96.30% on SVM model with a learning rate 0.01, preparing ages 10, and info units 86. The outcomes show that SVM model performs in a way that is better than other customary AI calculations: irregular woodland, direct relapse, and k-closest area.

S. Naseer et al. [3] examined the inconsistency based interruption discovery framework. They constructed a model that utilizes different machine and profound learning calculations for oddity put together interruption recognition with respect to NSL-KDD dataset. They analyzed the customary AI arrangement calculations nearest neighbor, SVM, Random Forest and Decision Tree to profound convolution neural organization (DCNN) and LSTM models and asserted approx. 85-89% precision on NSL-KDD test dataset.

Jiang et al. [4] proposed a multi-channel interruption identification framework that utilizations long transient memory repetitive neural organizations (LSTM-RNNs). They coordinated the information preprocessing, include reflection, multi-direct preparing and identification in the interruption recognition calculation. The presentation of the proposed

The Seybold Report
ISSN 1533-9211
DOI 10.5281/zenodo.6601187

framework broke down on NSL-KDD dataset. The framework announced approx. 98.94% precision and 99.23% identification rate.

M. Tavallee [5] proposed the multi-layered mixture network interruption recognition framework. They analyzed the presentation of the NSL-KDD dataset on various AI order calculations including Naive-Bayes, Support Vector Machines, and Decision-Trees. The half breed indicator detailed approx. 91% exactness.

Shone et al. [6] proposed a model that joins the profound and shallow learning, prepared to do accurately dissecting a wide-scope of organization traffic named as non-symmetric profound autoencoder (NDAE) for unaided element learning. They actualized the classifier in illustrations handling unit (GPU)- empowered Tensor Flow and assessed on the benchmark KDD Cup '99 and NSL-KDD datasets.

Salama et al. [7] proposed a model that consolidates the limited boltzmann machine and backing vector machine classifiers for interruption discovery on NSL-KDD dataset. They chose 22 assault types in preparing set and 17 assault types in testing set. The model created the best when contrasted with conventional help vector machine.

Biswas et al. [8] examined different component choice strategies and grouping procedures to limit the repetition in information ascribes through the blend of CFS, PCA and IGR highlight choice techniques before apply the arrangement calculations like SVM, k-NN, NN, DT and NB. The exploratory outcomes on NSL-KDD dataset show that the presentation of k-NN classifier is superior to different classifiers and furthermore the Information Gain Ratio (IGR) highlight determination technique produces best than other element choice strategies.

Zhao et al. [9] proposed the interference disclosure method using the blend of Deep Belief Network (DBN) and Probabilistic Neural Network (PNN). In this methodology, they convert the rough data into low-dimensional data by using nonlinear learning model to assemble the low-dimensional data. The tests performed on KDD Cup99 dataset. The results show that the introduction of proposed model is generally in a manner that is superior to standard techniques PNN, PCAPNN and DBN

3. EXPERIMENTAL ANALYSIS

This can be processed into about 5 million connection records, each with about 100 bytes. It consists of approximately 4,900,000 single connection vectors each of which contains 41 features.

These include Basic features (e.g. protocol type, packet size), Domain knowledge features (e.g. number of failed logins) and timed observation features (e.g. % of connections with SYN errors).

Each vector is label as either normal or as an attack (of which there are 22 specific attack types).

The Seybold Report
ISSN 1533-9211
DOI 10.5281/zenodo.6601187

3.1 Proposed System

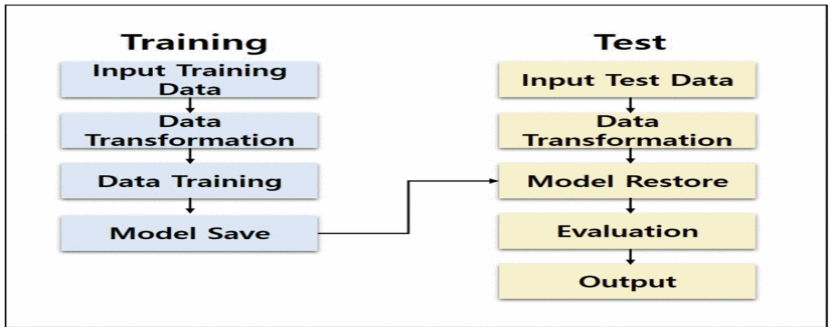

Fig. 1 Proposed system block diagram

3.2 ANALYSIS OF KDDCUP DATASET:

```
names = ["duration","protocol_type","service","flag","src_bytes",
    "dst_bytes","land","wrong_fragment","urgent","hot","num_failed_logins",
    "logged_in","num_compromised","root_shell","su_attempted","num_root",
    "num_file_creations","num_shells","num_access_files","num_outbound_cmds",
    "is_host_login","is_guest_login","count","srv_count","serror_rate",
    "srv_serror_rate","rerror_rate","srv_rerror_rate","same_srv_rate",
    "diff_srv_rate","srv_diff_host_rate","dst_host_count","dst_host_srv_count",
    "dst_host_same_srv_rate","dst_host_diff_srv_rate","dst_host_same_src_port_rate",
    "dst_host_srv_diff_host_rate","dst_host_serror_rate","dst_host_srv_serror_rate",
    "dst_host_rerror_rate","dst_host_srv_rerror_rate","label"]

df = pd.read_csv('Dataset/kddcup10.csv', names=names)
```

Fig. 2 KDD Dataset # Reading kddcup Dataset with 10% values.

```
RangeIndex: 494021 entries, 0 to 494020
Data columns (total 42 columns):
duration                     494021 non-null int64
protocol_type                494021 non-null object
service                      494021 non-null object
flag                         494021 non-null object
src_bytes                    494021 non-null int64
dst_bytes                    494021 non-null int64
land                         494021 non-null int64
wrong_fragment               494021 non-null int64
urgent                       494021 non-null int64
hot                          494021 non-null int64
num_failed_logins            494021 non-null int64
logged_in                    494021 non-null int64
num_compromised              494021 non-null int64
root_shell                   494021 non-null int64
su_attempted                 494021 non-null int64
num_root                     494021 non-null int64
num_file_creations           494021 non-null int64
num_shells                   494021 non-null int64
num_access_files             494021 non-null int64
num_outbound_cmds            494021 non-null int64
is_host_login                494021 non-null int64
is_guest_login               494021 non-null int64
count                        494021 non-null int64
srv_count                    494021 non-null int64
serror_rate                  494021 non-null float64
srv_serror_rate              494021 non-null float64
rerror_rate                  494021 non-null float64
```

Fig . 3 Column index with data type and object

The Seybold Report
ISSN 1533-9211
DOI 10.5281/zenodo.6601187

3.3 KDDcup Dataset – Distribution of attack

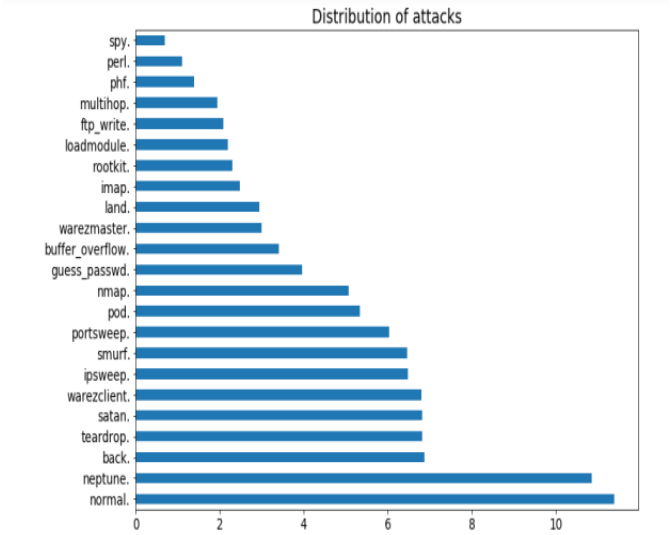

Fig. 4 Distribution of Attacks

3.4 KDDcup Dataset with SVM Model (80% Training Data & 20% Test Data)

	precision	recall	f1-score	support
0	1.00	1.00	1.00	17610
1	1.00	0.33	0.50	6
3	1.00	1.00	1.00	1
4	1.00	1.00	1.00	10326
5	1.00	0.98	0.99	122
6	1.00	0.91	0.95	11
7	1.00	1.00	1.00	38
8	1.00	1.00	1.00	193
9	1.00	0.97	0.98	91
10	0.97	0.99	0.98	134
11	1.00	1.00	1.00	1
12	0.00	0.00	0.00	3
13	0.99	0.99	0.99	214
14	1.00	1.00	1.00	2
15	0.99	0.97	0.98	170
16	1.00	1.00	1.00	1
17	1.00	0.74	0.85	39
19	1.00	1.00	1.00	2
20	0.90	0.94	0.92	153
22	0.00	0.00	0.00	1
accuracy			1.00	29118
macro avg	0.89	0.84	0.86	29118

Fig. 5 Overall accuracy of SVM model using test-set 20 is : 99.78 %

The Seybold Report
ISSN 1533-9211
DOI 10.5281/zenodo.6601187

3.5 KDDcup Dataset with SVM Model (70% Training Data & 30% Test Data)

	precision	recall	f1-score	support
0	1.00	1.00	1.00	26431
1	0.75	0.43	0.55	7
2	0.00	0.00	0.00	1
3	1.00	1.00	1.00	1
4	1.00	1.00	1.00	15475
5	1.00	0.98	0.99	192
6	1.00	0.95	0.97	20
7	1.00	1.00	1.00	53
8	1.00	1.00	1.00	282
9	0.99	0.98	0.98	133
10	0.96	0.97	0.96	208
11	1.00	1.00	1.00	4
12	0.00	0.00	0.00	3
13	0.98	0.99	0.99	300
14	1.00	1.00	1.00	2
15	1.00	0.98	0.99	253
16	1.00	1.00	1.00	1
17	1.00	0.73	0.84	55
19	1.00	1.00	1.00	5
20	0.91	0.94	0.92	247
21	0.00	0.00	0.00	1
22	0.00	0.00	0.00	2
accuracy			1.00	43676
macro avg	0.80	0.77	0.78	43676

Fig. 6 Overall accuracy of SVM model using test-set 30 is : 99.77%

3.6 KDDcup Dataset with SVM Model (60% Training Data & 40% Test Data:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	35261
1	0.67	0.44	0.53	9
2	0.00	0.00	0.00	2
3	1.00	1.00	1.00	1
4	1.00	1.00	1.00	20660
5	1.00	0.98	0.99	245
6	1.00	0.96	0.98	23
7	1.00	1.00	1.00	68
8	1.00	0.99	0.99	376
9	0.99	0.98	0.98	165
10	0.96	0.96	0.96	271
11	1.00	1.00	1.00	7
12	0.00	0.00	0.00	3
13	0.98	0.99	0.99	377
14	1.00	0.75	0.86	4
15	1.00	0.98	0.99	343
16	1.00	1.00	1.00	1
17	0.98	0.76	0.85	74
19	1.00	1.00	1.00	6
20	0.92	0.93	0.93	336
21	0.00	0.00	0.00	1
22	0.00	0.00	0.00	2
accuracy			1.00	58235
macro avg	0.80	0.76	0.78	58235

Fig. 7 Overall accuracy of SVM model using test-set 40 is : 99.76%

ISSN 1533-9211

DOI 10.5281/zenodo.6601187

3.7 ANALYSIS OF SVM MODEL

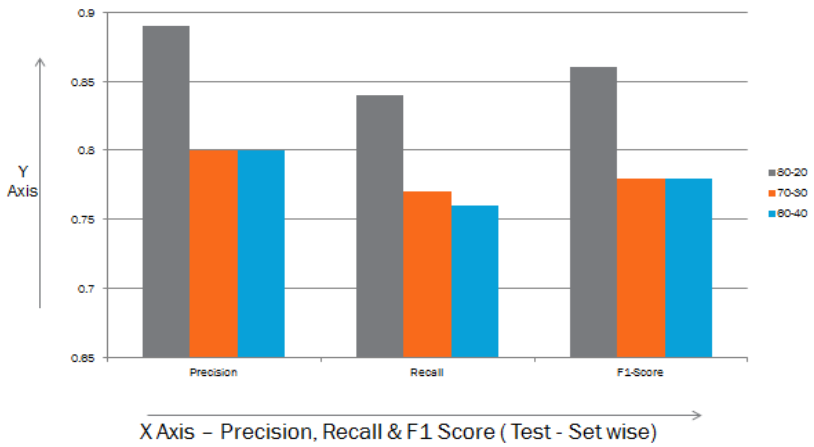

Fig. 8 Analysis of SVM Model

3.8 KDDcup Dataset with Naïve Bayes Model (80% Training Data & 20% Test Data)

	precision	recall	f1-score	support
0	1.00	0.45	0.62	17610
1	0.00	0.50	0.01	6
2	0.00	0.00	0.00	0
3	0.00	0.00	0.00	1
4	1.00	1.00	1.00	10326
5	0.80	1.00	0.89	122
6	1.00	0.91	0.95	11
7	0.75	1.00	0.85	38
8	0.99	1.00	0.99	193
9	0.75	0.88	0.81	91
10	0.11	0.99	0.19	134
11	1.00	1.00	1.00	1
12	0.00	0.00	0.00	3
13	0.10	1.00	0.18	214
14	1.00	1.00	1.00	2
15	0.26	0.92	0.40	170
16	1.00	1.00	1.00	1
17	0.03	0.31	0.05	39
18	0.00	0.00	0.00	0
19	0.01	1.00	0.02	2
20	0.02	0.47	0.04	153
22	0.00	0.00	0.00	1
accuracy			0.66	29118
macro avg	0.45	0.66	0.46	29118

Fig. 9 Overall accuracy of NB model using test-set 20 is : 66.35 %

The Seybold Report
ISSN 1533-9211
DOI 10.5281/zenodo.6601187

3.9 KDDcup Dataset with Naïve Bayes Model (70% Training Data & 30% Test Data)

	precision	recall	f1-score	support
0	1.00	0.47	0.64	26431
1	0.00	0.57	0.01	7
2	0.00	0.00	0.00	1
3	0.00	0.00	0.00	1
4	1.00	1.00	1.00	15475
5	0.79	1.00	0.88	192
6	1.00	0.95	0.97	20
7	0.76	1.00	0.86	53
8	1.00	1.00	1.00	282
9	0.75	0.88	0.81	133
10	0.04	0.28	0.06	208
11	1.00	1.00	1.00	4
12	0.00	0.00	0.00	3
13	0.09	1.00	0.16	300
14	1.00	1.00	1.00	2
15	0.23	0.94	0.37	253
16	1.00	1.00	1.00	1
17	0.06	0.96	0.11	55
18	0.00	0.00	0.00	0
19	0.02	1.00	0.04	5
20	0.02	0.42	0.03	247
21	0.00	0.00	0.00	1
22	0.00	0.00	0.00	2
accuracy			0.67	43676
macro avg	0.42	0.63	0.43	43676

Fig. 10 Overall accuracy of NB model using test-set 30 is : 67.06%

- This can be prepared into around 5 million association records, each with around 100 bytes. It comprises of around 4,900,000 single association vectors every one of which contains 41 highlights.
- These incorporate Basic highlights (for example convention type, parcel size), Domain information highlights (for example number of fizzled logins) and planned perception highlights (for example % of associations with SYN blunders).
- Each vector is mark as one or the other typical or as an assault (of which there are 22 explicit assault types).
- This can be prepared into around 5 million association records, each with around 100 bytes. It comprises of roughly 4,900,000 single association vectors every one of which contains 41 highlights.
- These incorporate Basic highlights (for example convention type, bundle size), Domain information highlights (for example number of fizzled logins) and coordinated perception highlights (for example % of associations with SYN mistakes).

KDD Cup Dataset comprises of around 4,94021 single association vectors every one of which contains 42 highlights. These incorporate Basic highlights (for example convention type, parcel size), Domain information highlights (for example number of fizzled logins) and planned perception highlights (for example % of associations with SYN blunders). Every vector is marked as one or the other ordinary or as an assault (of which there are 23 explicit assault types, as laid out in Table I). It is basic practice to utilize 10% of the full size dataset, as this furnishes an appropriate portrayal with diminished computational necessities. This

The Seybold Report
ISSN 1533-9211
DOI 10.5281/zenodo.6601187

10% subset is delivered and scattered close by the first dataset. In this paper, we utilize the 10% (thus alluded to as KDD Cup Dataset) subset, which contains 494,021 records.

Table 1. Composition of KDDcup Dataset

Sr. No.	Attack Type	Total No. of attack
1	normal.	87832
2	neptune.	51820
3	back.	968
4	teardrop.	918
5	satan.	906
6	warezclient.	893
7	ipsweep.	651
8	smurf.	641
9	portsweep.	416
10	pod.	206
11	nmap.	158
12	guess_passwd.	53
13	buffer_overflow.	30
14	warezmaster.	20
15	land.	19
16	imap.	12
17	rootkit.	10
18	loadmodule.	9
19	ftp_write.	8
20	multihop.	7
21	phf.	4
22	perl.	3
23	spy.	2

Evaluation Metrics

Assessment or Performance Metrics are utilized to assess the presentation of Intrusion identification frameworks (IDS) by utilizing grouping calculations. The anticipated results are between the reach 0 to 1. The disarray framework shows the factual outcomes based on real or anticipated records in a dataset. The most usually utilized assessment measurements are as per the following.

1) Accuracy: It is the assessed proportion of accurately perceived information records to the absolute number of information records in a given informational collection. The higher

ISSN 1533-9211

DOI 10.5281/zenodo.6601187

pace of precision shows that the model is performed better. Precision is characterized as follows.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

2) Precision: It is the assessed proportion of effectively distinguished assault information records to the absolute number of all recognized information records in a given dataset. The higher pace of exactness shows that the model is performed better. Exactness is characterized as follows.

$$\text{Precision} = \frac{TP}{TP + FP}$$

3) Recall: It is the assessed proportion of accurately arranged assault information records to the all out number of assault information records in a given dataset. The higher pace of review shows that the model is performed better. Review is characterized as follows.

$$\text{Recall} = \frac{TP}{TP + FN}$$

4) F1-Score-It is the symphonious mean of Precision and Recall. The higher pace of F1-Score shows that the model is performed better.
F1-Score is characterized as follows.

$$\text{F-score} = 2 \cdot \frac{Precision \cdot Recall}{Precision \ + \ Recall}$$

In above Equations, the term TP, TN, FP and FN are utilized for portraying the arrangement of Normal and Attack records in a dataset.

TP (True Positive) characterizes that the quantity of association records accurately ordered or recognized into Normal class of dataset likewise TN (True Negative) characterizes that the quantity of association records effectively arranged or distinguished into an assault class of dataset.

FP (False Positive) characterizes that the quantity of typical class association records are wrongly ordered or recognized into assault class correspondingly FN (False Negative) characterizes that the quantity of assault class association records are wrongly grouped or distinguished into ordinary class association records.

4. RESULT AND ANALYSIS

Support Vector Machine (SVM) is based on the principle of structural risk minimization, looking for an optimal interval to divide the instance into two categories

The Seybold Report
ISSN 1533-9211
DOI 10.5281/zenodo.6601187

TABLE 2. Comparison results for accuracy, precision, recall & Accuracy within the Support Vector Machine Method with 80%-20%, 70%-30% and 60%-40% Training & Testing Data on the KDDCup Dataset.

	Train – Test Data Set	Precision	Recall	F1-Score	Accuracy
Support vector machine (SVM)	80-20	0.89	0.84	0.86	99.78%
	70-30	0.8	0.77	0.78	99.77%
	60-40	0.8	0.76	0.78	99.77%

Note - Analysis of SVM Model with three categories – a) 80% Training Data & 20% Test Data, b) 70% Training Data & 30% Test Data, c) 60% Training Data & 40% Test Data)

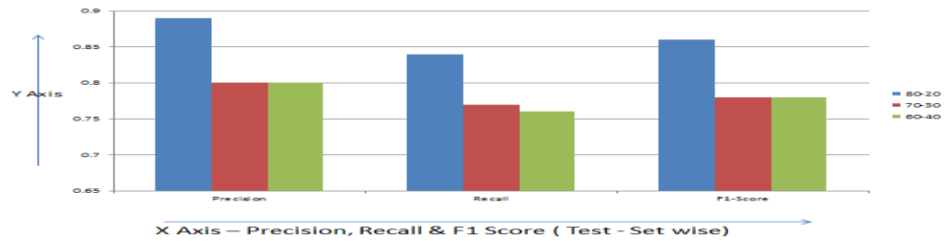

Fig 11. Graphical Representation of Performance of Precision, Recall and F1 Score on Vs Support Vector Machine Model on KDDCup Dataset.

Naive Bayes method is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

TABLE 3. Comparison results for accuracy, precision, recall & Accuracy within the Naive Bayes Method with 80%-20%, 70%-30% and 60%-40% Training & Testing Data on the KDDCup Dataset.

	Train – Test Data Set	Precision	Recall	F1-Score	Accuracy
NB	80-20	0.45	0.66	0.46	66.35%
	70-30	0.42	0.63	0.43	67.06%
	60-40	0.41	0.61	0.42	65.62%

Note - Analysis of Naive Bayes Model with three categories – a) 80% Training Data & 20% Test Data, b) 70% Training Data & 30% Test Data, c) 60% Training Data & 40% Test Data)

ISSN 1533-9211

DOI 10.5281/zenodo.6601187

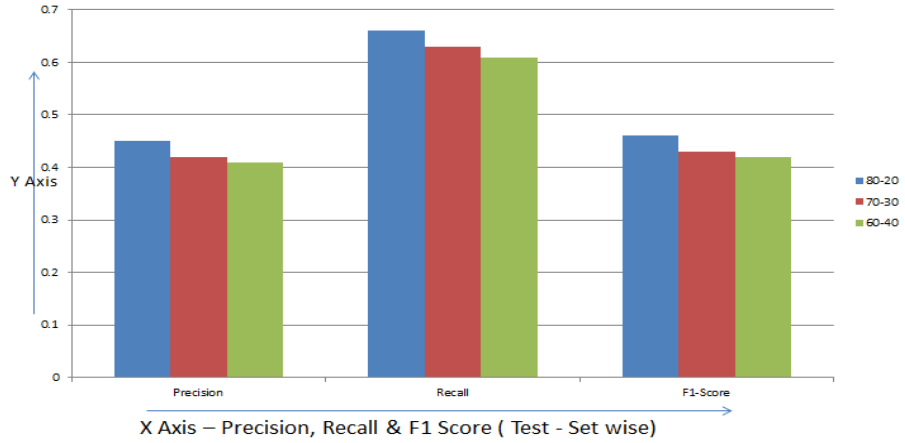

Fig 12. Graphical Representation of Performance of Precision, Recall and F1 Score on Vs Naive Bayes Model on KDDCup Dataset.

Random forest (RF)s are supervised learning algorithms. In an RF, several DTs are constructed and combined to acquire a precise and robust prediction model for improved overall results [165, 166]. Therefore, an RF consists of numerous trees that are constructed randomly and trained to vote for a class. The most voted class is selected as the final classification output [165]. Even though the RF classifier is constructed mainly using DTs, these classification algorithms substantially differ. Firstly, DTs normally formulate a set of rules when the training set is fed into the network, and this set of rules is subsequently used to classify a new input.

TABLE 4. Comparison results for accuracy, precision, recall & Accuracy within the Random forest Method with 80%-20%, 70%-30% and 60%-40% Training & Testing Data on the KDDCup Dataset.

	Train – Test Data Set	**Precision**	**Recall**	**F1-Score**	**Accuracy**
Random Forest (RF)	80-20	0.88	0.86	0.87	99.91%
	70-30	0.8	0.78	0.78	99.91%
	60-40	0.77	0.74	0.75	99.92%

Note - Analysis of Random Forest Model with three categories a) 80% Training Data & 20% Test Data, b) 70% Training Data & 30% Test Data, c) 60% Training Data & 40% Test Data.

ISSN 1533-9211

DOI 10.5281/zenodo.6601187

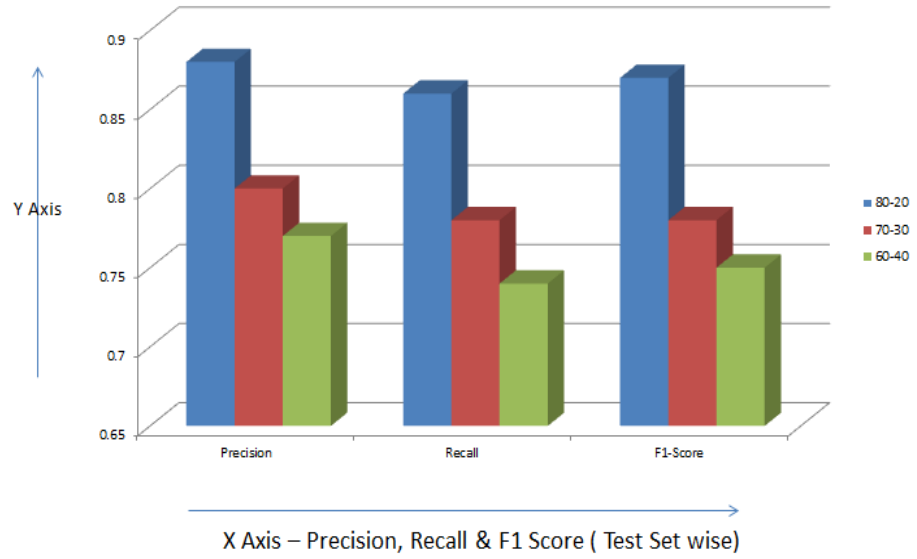

Fig 13. Graphical Representation of Performance of Precision, Recall and F1 Score on Vs Random Forest Model on KDDCup Dataset.

k-Nearest Neighbour (k-NN) is a simple and effective method for target classification based on the most recent training samples into feature space. When the prior knowledge about the data distribution is little or no prior knowledge, the KNN classifier transforms the samples into metric space and classifies new points based on the majority of votes obtained from the K nearest points in the training data. Usually, the Euclidean distance is often used as a distance metric to measure the similarity between two vectors.

TABLE 5. Comparison results for accuracy, precision, recall & Accuracy within the k-Nearest Neighbour Method with 80%-20%, 70%-30% and 60%-40% Training & Testing Data on the KDDCup Dataset.

	Train – Test Data Set	**Precision**	**Recall**	**F1-Score**	**Accuracy**
k-Nearest Neighbor (KNN)	80-20	0.89	0.85	0.86	99.77%
	70-30	0.79	0.78	0.78	99.77%
	60-40	0.76	0.77	0.76	99.78%

Note - Analysis of KNN Model with three categories, a) 80% Training Data & 20% Test Data, b) 70% Training Data & 30% Test Data, c) 60% Training Data & 40% Test Data

ISSN 1533-9211

DOI 10.5281/zenodo.6601187

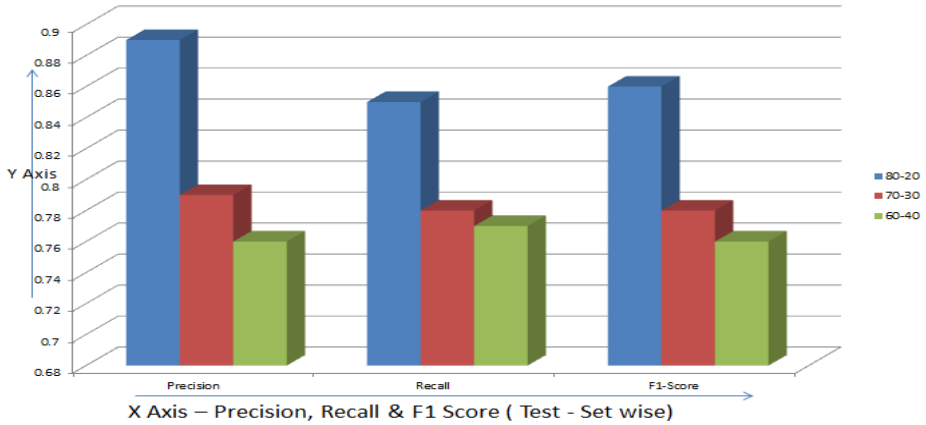

Fig 14. Graphical Representation of Performance of Precision, Recall and F1 Score on Vs KNN Model on KDDCup Dataset.

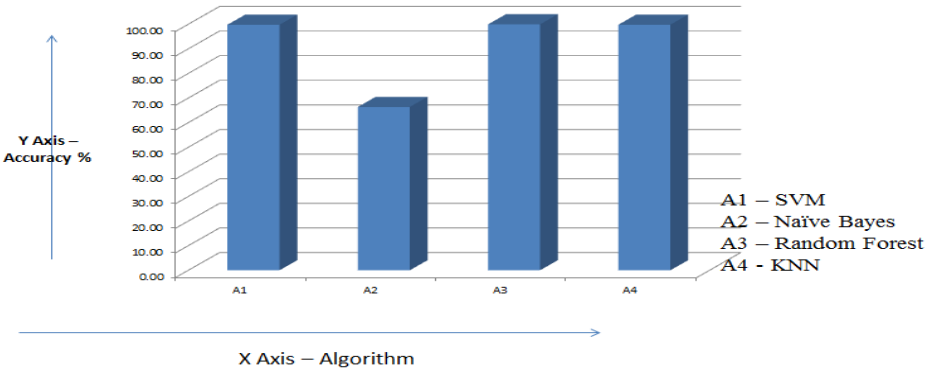

Fig.15. Graphical Representation of Accuracy Vs Support Vector Machine, Naïve Bayes, Random Forest & KNN Algorithms

In this measurement and comparison between two given platforms, namely, Vs Support Vector Machine, Naïve Bayes, Random Forest & KNN Algorithms with Four metrics namely accuracy, Precision, recall and F1 – Score. SVM scored 99.77%, Naïve Bayes Scored 66.34%, Random forest scored 99.91% and KNN scored 99.77% Average Accuracy respectively. SVM scored 83%, Naïve Bayes Scored 43%, Random forest scored 82% and KNN scored 81% respectively for Precision. And also, in terms of recall, SVM reached to the percentage of 79%, and 63%, 79%, and 80% are respectively for Naïve Bayes, Random Forest and KNN. SVM scored 81%, Naïve Bayes Scored 44%, Random forest scored 80% and KNN scored 80 for f1 score respectively. On the basis of evaluation metrics results, we have investigated a comparative analysis on benchmark datasets KDDCup Dataset, Random Forest perform highest Accuracy (99.91%), Support Vector Machine perform highest precision (83%), KNN performs highest Recall value (80%) and Support vector Machine performs highest f1 score value (81%).

ISSN 1533-9211

DOI 10.5281/zenodo.6601187

In this phase, we have investigated a comparative analysis on benchmark datasets KDDCup Dataset by using deep learning classification algorithms. Four classification algorithms have been used to measure the performance of accuracy, precision, recall and f1-score of datasets for intrusion detection system. On the basis of evaluation metrics results, we have concluded that our algorithms based on k-NN, SVM and RF classifiers perform approx. 100% in terms of performance evaluation metrics on KDDCup dataset, whereas Naïve Bayes classifiers perform approx. 66% Recall on KDDCup dataset. Hence, the comparative study results have promoted the hybrid feature selection methods for better performance of cutting-edge classifiers.

The findings from analysis have shown that despite the high detection accuracies being achieved, there is still room for improvement. Such weaknesses include the reliance on human operators, long training times, inconsistent or average accuracy levels and the heavy modification of datasets (e.g. balancing or profiling). Previous works have mainly considered accuracy in terms of performance measures, but scalability and precision are also important indicators for applying deep belief learning in the real-world Internet of Things.

5. CONCLUSION

In this paper, The findings from analysis have shown that despite the high detection accuracies being achieved, there is still room for improvement. Such weaknesses include the reliance on human operators, long training times, inconsistent or average accuracy levels and the heavy modification of datasets (e.g. balancing or profiling). Previous works have mainly considered accuracy in terms of performance measures, but scalability and precision are also important indicators for applying deep belief learning in the real-world Internet of Things.

6. Future Work

As upcoming Phase, with design of proposed system concentrate on the accompanying issues to examine solid and weak purposes of various location strategies IoT, to expand the attack discovery range, to address more IoT advances, to improve security of alert traffic and the executives; and to grow supplementary requests, for example, aware connection then autonomic administration frameworks.

References

1. Agyemang B., Xu Y., Sulemana N., and Hu H., “Resource-oriented architecture toward efficient device management for the Internet of Things,” Journal of Ambient Intelligence and Humanized Computing, 1-13, (2018).

2. Ali M. A., Fadlizolkipi M., Firdaus A., Khidzir N. Z., A hybrid Particle swarm optimization -Extreme Learning Machine approach for Intrusion Detection System, 978-1-5386-9175-5/18-IEEE, 1-4, (2018).

3. Al-Garadi M A, Mohamed A, Al-Ali A, Du X, Ali I, Guizani M, A Survey of Machine and Deep Learning Methods for Internet of Things (IoT) Security, IEEE Communications Surveys & Tutorials, 1-46, (2020).

The Seybold Report
ISSN 1533-9211
DOI 10.5281/zenodo.6601187

4. Anguraj, Dinesh Kumar, and Smys S.. "Trust-based intrusion detection and clustering approach for wireless body area networks." Wireless Personal Communications. 104 (1), 1-20, (2019).

5. Atefi K, Hashim H, Khodadadi T, A Hybrid Anomaly Classification with Deep Learning (DL) and Binary Algorithms (BA) as Optimizer in the Intrusion Detection System (IDS), 16th IEEE International Colloquium on Signal Processing & its Applications (CSPA 2020), 978-1-7281-5310-0/20-IEEE, 29-34, (2020).

6. Babovic Z. B., Protic Jelica, and Milutinovic Veljko, "Web performance evaluation for internet of things applications." IEEE Access. 4, 6974-6992, (2016).

7. Bellavista P., Cardone G., Corradi A., and Foschini L., "Convergence of MANET and WSN in IoT urban scenarios," IEEE Sens. J., 13(10), 3558–3567, Oct. (2013).

8. Chaqfeh M. A. and Mohamed N., "Challenges in middleware solutions for the Internet of Things," in Proc. Int. Conf. CTS, 21–26, (2012).

9. Chen, Yen-Kuang. "Challenges and opportunities of internet of things." Design Automation Conference (ASP-DAC), 2012 17th Asia and South Pacific. IEEE, (2012).

10. Chen S., Xu H., Liu D., Hu B., and Wang H., "A vision of IoT: Applications, challenges, and opportunities with china perspective," IEEE Internet Things J., 1(4), 349–359, (2014).

11. Choi J., In Y., Park C., Seok S., Seo H., and Kim H., "Secure IoT framework and 2D architecture for End-To-End security," The Journal of Supercomputing, 1-15, (2016).

12. Ciuffoletti. A, "OCCI-IOT: an API to deploy and operate an IoT infrastructure", IEEE Internet of Things Journal, 4 (5), 1341-1348, (2017).

13. Diogo P., Lopes N. V., and Reis L. P., "An ideal IoT solution for real-time web monitoring," Cluster Computing, 20(3), 2193-2209, (2017).

14. Dong Y., Wang R., He J., Real-Time Network Intrusion Detection System Based on Deep Learning, 978-l-7281-0945-9 IEEE, 1-4, (2019).

15. Emadi S, Al-Mohannadi A, Al-Senaid F., Using Deep Learning Techniques for Network Intrusion Detection, 978-1-7281-4821-2/20 - IEEE, 171-176, (2020).

16. Espí-Beltrán J. V., Gilart-Iglesias V., and Ruiz-Fernandez D. , "Enabling distributed manufacturing resources through SOA: The REST approach", Robotics and Computer-Integrated Manufacturing, 46, 156-165, (2017).

17. Fremantle P., and Aziz B., "Cloud-based federated identity for the Internet of Things." Annals of Telecommunications, 1-13, (2018).

18. Gardaševic G., Veletic M., Maletic N., Vasiljevic D., Radusinovic I., Tomovic S., and Radonjic M., "The IoT architectural framework, design issues and application domains," Wireless Personal Communications, 92 (1) , 127-148, (2017).

19. Gumusbas D, Yildirim T, Genovese A, Scotti F, A Comprehensive Survey of Databases and Deep Learning Methods for Cybersecurity and Intrusion Detection Systems, IEEE Systems Journal, 0.1109/JSYST.2020.2992966, (2020).

20. Halimaa A., Sundarakantham K., MACHINE LEARNING BASED INTRUSION DETECTION SYSTEM, Third International Conference on Trends in Electronics and Informatics (ICOEI 2019), 978-1-5386-9439-8, IEEE, 916-920, (2019).

21. Hakim L., Fatma R., Novriandi, Influence Analysis of Feature Selection to Network Intrusion Detection System Performance Using NSL-KDD Dataset, 978-1-7281-3436-9 - IEEE, 217-220, (2019).

The Seybold Report
ISSN 1533-9211
DOI 10.5281/zenodo.6601187

22. Han S. N., and Crespi N., "Semantic service provisioning for smart objects: Integrating IoT applications into the web", Future Generation Computer Systems, (2017).

23. Hou L., Zhao S., Xiong X., Zheng K., Chatzimisios P., Hossain M. S., and Xiang W., "Internet of things cloud: architecture and implementation," IEEE Communications Magazine, 54(12), 32-39, (2016).

24. Hui T. K., Sherratt R. S., and Sánchez D. D., "Major requirements for building Smart Homes in Smart Cities based on Internet of Things technologies", Future Generation Computer Systems, (2016).

25. Hwang Hyun Cheon, Park JiSu, and Shon Jin Gon, "Design and implementation of a reliable message transmission system based on MQTT protocol in IoT," Wireless Personal Communications, 91(4), 1765-1777, (2016).

26. Iqbal A., Ullah F., Anwar H., Kwak K. S., Imran M., Jamal W., and Rahman A., "Interoperable Internet-of-Things platform for smart home system using Web-of-Objects and cloud," Sustainable Cities and Society, 38, 636-646, (2018).

27. Ishaque M., hudec L., Feature extraction using Deep Learning for Intrusion Detection System, 978-1-7281-0108-8, IEEE, 1-5, (2019).

28. Khaled A. E., and Helal A., "Interoperable communication framework for bridging RESTful and topic-based communication in IoT," Future Generation Computer Systems, (2018).

29. KIM A, PARK M, LEE D H, AI-IDS: Application of Deep Learning to Real-Time Web Intrusion Detection, SPECIAL SECTION ON SCALABLE DEEP LEARNING FOR BIG DATA - IEEE Access, 10.1109/ACCESS.2020.2986882, 08. 70245-70261, (2020).

30. Khan R., Khan S. U., Zaheer R., and Khan S., "Future Internet: The Internet of Things architecture, possible applications and key challenges," in Proc. 10th Int. Conf. FIT, 257–260, (2012).

31. Krco S., Pokric B., and Carrez F., "Designing IoT architecture(s): A European perspective," in Proc. IEEE WF-IoT, pp. 79–84, (2014).

32. Kumbhare A. G., Simmhan Y., Frincu M., and Prasanna V. K., "Reactive resource provisioning heuristics for dynamic dataflows on cloud infrastructure", IEEE Transactions on Cloud Computing, 3(2), 105-118, (2015).

33. H. N. Bhor and M. Kalla, "An Intrusion Detection in Internet of Things: A Systematic Study," 2020 International Conference on Smart Electronics and Communication (ICOSEC), 2020, pp. 939-944, doi: 10.1109/ICOSEC49089.2020.9215365.

34. Bhor H.N., Kalla, M. (2021). A Survey on DBN for Intrusion Detection in IoT. In: Zhang, YD., Senjyu, T., SO–IN, C., Joshi, A. (eds) Smart Trends in Computing and Communications: Proceedings of SmartCom 2020. Smart Innovation, Systems and Technologies, vol 182. Springer, Singapore. https://doi.org/10.1007/978-981-15-5224-3_33.

35. Bhor H N, Kalla M. TRUST-based features for detecting the intruders in the Internet of Things network using deep learning. Computational Intelligence. 2021;1–25. https://doi.org/10.1111/coin.12473

36. H. N. Bhor and Dr. Mukesh Kalla, "A DEEP LEARNING BASED DATA SECURITY SYSTEM IN IOT NETWORK", Indian Patent 16/2022, 25037, April 22, 2022.