Computing, Information Systems, Development Informatics & Allied Research Journal
Vol. 12 No. 1, 2021 - www.cisdijournal.net

Performance Evaluation of Supervised Ensemble Cyber Situation Perception Models for Computer Network

***Olofintuyi, S.S. & Omotehinwa, T.O.**
Department of Mathematical Sciences
Achievers University
Owo, Ondo State Nigeria
*** Email**: olofintuyi.sundaysamuel@gmail.com
Phone: +2348061681274

ABSTRACT

The trend at which cyber threats are gaining access to companies, industries and other sectors of the economy is becoming alarming, and this is posting a serious challenge to network administrators, governments and other business owners. A formidable intrusion detection system is needed to outplay the activities of the cyberattacks. An ensemble system is believed to perform better than a single classifier. With this fact, five different Machine Learning (ML) ensemble algorithms are suggested at the perception phase of Situation Awareness (SA) model for threat detection and the algorithms include; Artificial Neural Network Based Decision Tree (ANN based DT), Bayesian Based Artificial Neural Network (BN based ANN), J48 Based Naïve Bayes Model (J48 based NB), Decision Tree based Bayesian Network (BN) and Random Forest based on Support Vector Machine (RF based SVM). The efficiency and effectiveness of all the aforementioned algorithms were evaluated based on precision, recall and accuracy. ANN based DT gave 98.87% accuracy, BN based ANN gave 99.72% accuracy, J48 based NB gave 98.90% accuracy, DT based BN gave 89.92% accuracy and FR based SVM gave 98.40% accuracy. The implication of these results is that BN based ANN is more suitable in the perception phase of SA for threats detection.

Keywords- Cyber-threats, Ensemble Algorithms, Computer Network, Intrusion Detection System, Machine Learning

CISDI Journal Reference Format
Olofintuyi, S.S. & Omotehinwa, T.O. (2021): Performance Evaluation of Supervised Ensemble Cyber Situation Perception Models for Computer Network. Computing, Information Systems, Development Informatics & Allied Research Journal. Vol 11 No 2, Pp 1-.14.
DOI Affix - https://doi.org/10.22624/AIMS/CISDI/2021/V12N1P1. Available online at www.computing-infosystemsjournal.info

1. INTRODUCTION

The tremendous growth of cyber threats over the years have been disturbing for network and application users. The activities of cyber threats became more rampant following the arrival of the internet of things (IoTs) (Sarker *et al.,* 2020). The effects of cyberthreats have been devastating on industries and companies and various sectors of the economy. Record has it that there were about 50 million malwares in 2010. Surprisingly, the number was doubled to 100 million in just 2 years. By year 2019, the number has greatly increased to 900 million known cyber threats (Sarker *et al.,* 2020). Some of the recent activities of cyber threat on company include activity on American Medical Collection Agency (AMCA), the company records were intruded by hackers for almost a year unknowing to the company in which 12 million records were compromised and a total of 12 million hosts affected (Hao *et al.,* 2020). Another recent example of cyber threat on companies is their activity on insurance companies. 900 million records of First America were compromised in the year 2019 (Sarker *et al.,* 2020).

Despite the effort of the researcher institute, government and many other companies in curbing these threats by investing heavily with manpower and money, the devastating effects of these threats are on the high side and they are becoming sophisticated in their operation. Presently, all the proposed attack detection mechanisms have deficiency in their detection rate (Olofintuyi *et al.,* 2019). An Intrusion Detection System (IDS) has been proposed to guide against activities of intruder, IDS helps to categorize the various threats to their respective classes using either machine learning or statistical methods (Hao *et al.,* 2020). User perspective on IDS differs, the known types of IDS include Host- based Intrusion Detection System (HIDS) and Network-based Intrusion Detection System (NIDS). In NIDS, activities of the threat are being checkmated across the computer network while in HIDS, the discrepancy and irregularity in operating systems are being scanned for (Olofintuyi, 2021). Approaches to IDS are either signature based or anomaly-based approaches. Signature based IDS can only detect known cyber threats because its record is already in the database of the administrator while anomaly-based IDS can detect a novel threat that is not in the database of the administrator (Olofintuyi, 2021). Data driven IDS is an IDS that is driven by cyber security data. The cyber security data has a pattern in it which helps the IDS to make predictions.

Machine Learning (ML) plays an important role in the data-driven IDS, despite the key role of ML in data driven IDS, there still remain some challenges of the different classifiers which use cyber data for prediction. The major reason for these challenges is because different classifiers make their predictions based on different contexts (Sarker *et al.,* 2019). Also, the traditional algorithm for threat detection is not robust enough for threat detection on the computer network (Mohammadi *et al.,* 2019). Before now various ML classifiers have been used which include SVM (Shams and Rizaner, 2018), ANN (Olofintuyi *et al.,* 2019; Sarker *et al.,* 2019) Bayesian method (Sarker, 2019), Tree based approach (Sarker *et al.,* 2020; Sarker *et al.,* 2019; Sarker and Salim, 2018) models and so on. The performance of all these models were evaluated based on the following metrics; True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). It seems impossible to reach a satisfactory level with all the metrics at the same time because they are all dependent on each other, striking the balances between all these metrics that are dependent is a big challenge (Hao *et al.,* 2020).

Based on these reasons, this study proposed different possible ensemble ML algorithms which includes; Artificial Neural Network Based Decision Tree (ANN Based DT), Bayesian Based Artificial Neural Network (BN Based ANN), J48 Based Naïve Baye Model (J48 Base NB), Decision Tree Based Bayesian Network (DT Based BN), and Random Forest Based on Support Vector Machine (RF Based SVM) in the perception phased of a Situation Awareness (SA) model. The SA model as explained by Endsley in his work classified the SA model into three phases; the perception phase, comprehension phase and projection phase (Olofintuyi *et al.,* 2019; Endsley, 1995). Malicious events such as probe, Denial of Service (DOS), Root to Local (R2L) and User to Root (U2R) are detected by the perception phase of SA model and then relay to the comprehension phase and finally to the projection phase which gives the network administrator all the necessary feedback (Olofintuyi *et al.,* 2019; Olofintuyi, 2021; Endsley, 1995). The different ensemble models are then compared with each other to see which one performs better. The next section of this work discusses the materials and methods used. Immediately, after the materials and methods used have been discussed, the experimental results derived from the research was discussed. The last section concluded the research work.

2. REVIEW OF RELATED WORK

Machine learning based approach has been used by various researchers to propose both signature and anomaly methods for threat detection and classification. Aslahi-Shahri *et al.,* (2016) proposed a hybrid method of genetic algorithm and support vector machine for threat detection. KDD 99 dataset was used in their proposed algorithm. The experimental result shows that 97.3% accuracy was obtained.

Chen et al., (2011) in their research work suggested a clustering algorithm and SVM. Also, the author deployed KDD 99 dataset and 95.7% accuracy was achieved. Olofintuyi, (2021) proposed an ensemble system of ANN based on a decision tree for threat detection, NSI-KDD dataset was used during the model building and simulation. Experimental results revealed that the proposed algorithm gave an accuracy of 98.7%. Harbi, (2010) used decision trees for threat detection and an accuracy of 98% was achieved using KDD 99 dataset. Sahu and Mehtre, (2015) proposed J48 for threat detection by using Kyoto 2006 dataset. 97.2% accuracy was achieved by the model. Jukic and Subasi, (2017) suggested combination of tree algorithms for threats detection, 89.24% accuracy NSL-KDD was achieved by the model. Wang *et al.,* (2010) proposed clustering algorithm and ANN. NSL-KDD dataset was used during the simulation phase, 96.71% accuracy was achieved after the experiment.

Machine learning can be supervised and unsupervised. In supervised ML, regression method and classification method are the most popular (Sarker *et al.,* 2019). These two popular approaches are used to predict any security problem in the future. For example, classification technique such as Naïve Bayes (John, 1995), ZeroR (Witten and Frank, 2005), SVM (Keerthi *et al.,* 2001), DT (Sarker *et al.,* 2019), One R (Holte, 1993), K-nearest neighbor (Aha *et al.,* 1991), logistic regression (Cessie and Houwelingen, 1992) and adaptive learning (Freund *et al.,* 1996) can be used to group threat to their respective classes. Recently, Olofintuyi, (2021) presented an ANN based DT for threat detection. Also, Sarker *et al.,* (2020) presented IntrudTree for threat classification on a computer network. On the other hand, regression technique is very useful when it comes to network packet parameters prediction. Another usefulness of regression technique is that it is suitable to detect the main reason for cybercrime and other related fraud (Watters *et al.,* 2012). Examples of regression methods include support vector regression Keerthi *et al.,* (2001) and linear regression (Han *et al*., 2011). The sole difference between regressions and classification is in their output.

For regression, the output variable is continuous or numerical which is contrary to classification method which output is discrete or in category. Hybridized learning is believed to be an extension of supervised learning because it mixes the various algorithms to solving a particular task (Breiman, 2001). Unsupervised ML detects structure and pattern in an unlabeled dataset (Sarker *et al.,* 2019). Cyber threats change their behaviors and patterns on the networks so that IDS wouldn't be able to detect them. The hidden pattern of these cyber threats can be detected by using clustering technique which is a good example of an unsupervised algorithm. Clustering algorithms can also be used to eliminate unwanted instances in a given dataset, identify anomalies and used for policy violation. Some of the most clustering algorithms include K-medoids Rokach (2010) and K-mean MacQueen (1967).

Another type of ML is known as semi-supervised ML. It lies in between both the supervised and unsupervised ML. semi supervised ML needs less time to handle small amounts of labelled data Ashfaq *et al.,* (2017), Self-training Lyngdoh *et al.,* (2018), Expectation maximization-based algorithms Goldstein (2012), Co-training Rath *et al.,* (2017) are some of the proposed techniques of supervised machine learning. A more accurate predictive performance is achieved by using multiple classifiers of ML than an individual classifier. Various classifiers can be ensembled using stacking, boosting and Bagging. Boosting basically turns a weak classifier to a strong classifier with bagging, different subsets of the same dataset are trained with the same classifier. Stacking uses meta-classifiers by combining various classification Aburomman and Ibne (2017). Jabbar *et al.,* (2017) presented an ensemble system where Naïve Baye and Random forest were ensembled. The proposed ensemble system gave better accuracy than the individual classifiers.

3. METHODOLOGY

This research adopts NSL-KDD dataset for modeling and training in the perception phase of SA. Models building and training were done in the WEKA environment. The model introduces five possible ensemble ML algorithms in the perception phase and then compares the results with each other. The ensemble ML that gives the highest accuracy for threat detection is forwarded into the classification phase and then into the comprehension phase which is finally projected to the network administrator. Figure 1 shows the five different ensemble models in the perception phase of SA

3.2 Dataset

NSL-KDD is an extract from KDD-99 dataset. The dataset was extracted in the year 2009 in order to eliminate all the irrelevant and redundant records that replicated themselves in the KDD-99 dataset. The NSL dataset has 41 features and all the threats are classified into four groups. The online dataset is numeric in nature and it ranges from different numbers. All the numeric numbers were converted to 0's and 1's. The four groups are explained below;

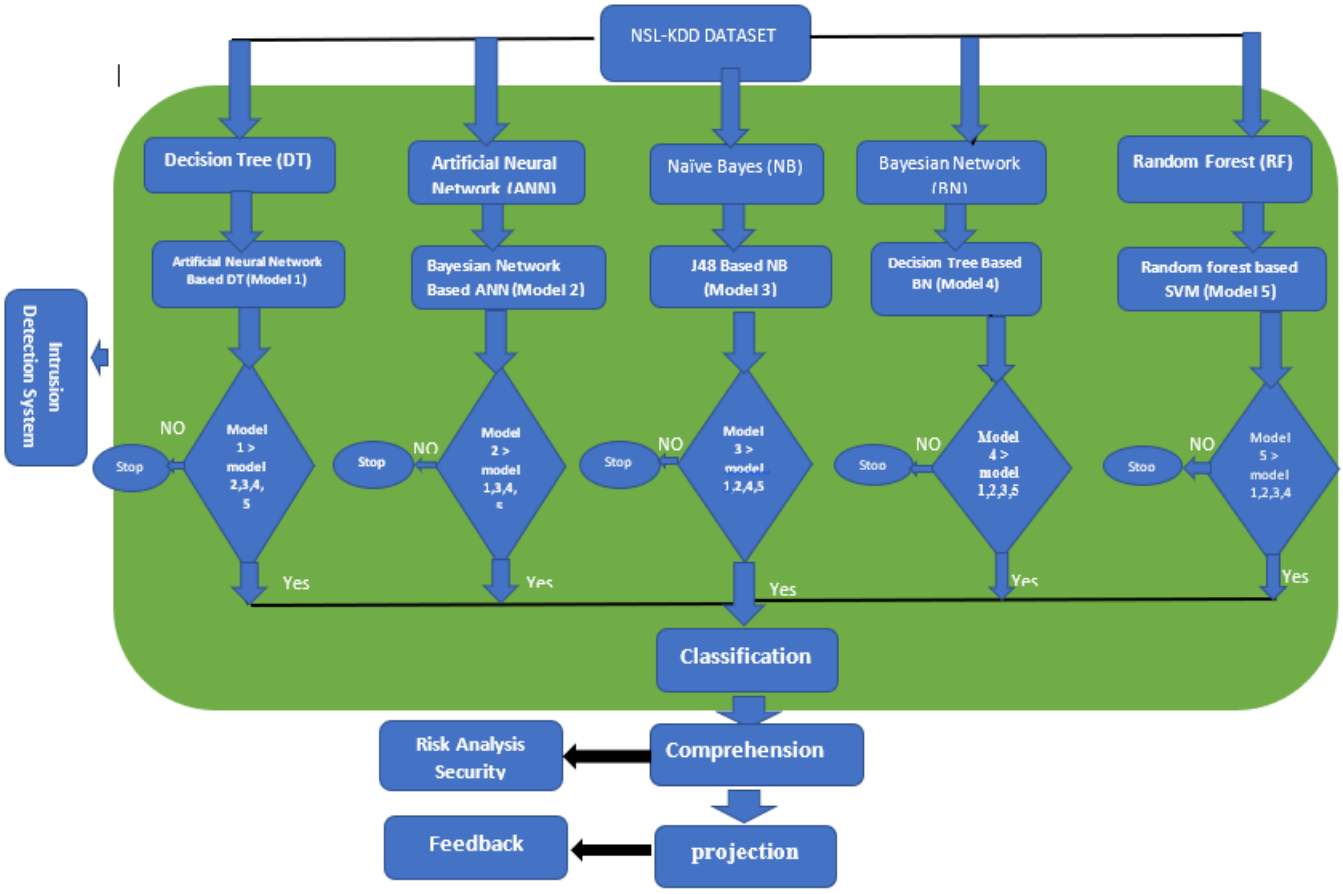

Figure 1: Proposed Situation Awareness Model

a) Denial of Service (DoS): The sole objective of the threats is to restrict or block activities performed on the network. This set of threats does its operation by keeping the memory so busy that legitimate requests will not be attended to. Examples of Dos includes; Ping of death, Back, SYN Flood, Process table, Apache2, Land and Mail bomb.
b) Probe: the sole aim of these threats is to acquire all the relevant information about the computer system. Once the information is obtained, the vulnerabilities are then explored. Examples of probes includes; Satan, Ipsweep, Saint, Mscan and Nmap.
c) User to Root (U2R): These groups of threats pretend to be a legitimate user and as a result, gained access to a computer system. Once the access is gained, the vulnerabilities of such a system are explored. Examples of U2R includes; Loadmodule, Perl, Fdformat and Xterm.
d) Root to Local (R2L): These set of threats deceive the end user of a computer network by sending packets of data to them which when accepted, makes the users system vulnerable. Examples of R2L includes; Dictionary, Guest, FTP write, Xlock and Imap. Table 1 depicts 41 features of the dataset used.

Table 1: Forty One (41) Aattributes of the Dataset Used

No	Feature name	Types	No	Feature Name	Types	No	Feature name	Types
1	Duration	continuous	15	Su_attempted	Continuous	29	Same_srv_rate	Continuous
2	Protocol type	Symbolic	16	Num_root	Continuous	30	Diff_srv_rate	Continuous
3	service	Symbolic	17	Num_file creation	Continuous	31	Srv_diff_host_rate	Continuous
4	Flag	Symbolic	18	Num_shell	Continuous	32	Dst_host_count	Continuous
5	Scr_bytes	continuous	19	Num_access file	Continuous	33	Dst_host_srv_count	Continuous
6	Dst_bytes	Continuous	20	Num_outbound_cmds	Continuous	34	Dst_host_same_srv_rate	Continuous
7	Land	Symbolic	21	Is_host_login	symbolic	35	Dst_host_diff_srv_rate	Continuous
8	Wrong fragment	Continuous	22	Is_guest_login	symbolic	36	Dst_host_same_src_port_rate	Continuous
9	Urgent	Continuous	23	count	Continuous	37	Dst_host_srv_diff_host_rate	Continuous
10	Hot	Continuous	24	Srv_count	Continuous	38	Dst_host_serror_rate	Continuous
11	Num_failed login	continuous	25	Serror_rate	Continuous	39	Dst_host_srv_rate	Continuous
12	Logged_in	Symbolic	26	Srv_serror_rate	Continuous	40	Dst_host_srv_serror_rate	symbolic
13	Num_compropmised	Continuous	27	Rerror_rate	Continuous	41	Dst_host_serror_rate	symbolic
14	Root_shell	Continuous	28	Srv_rerror_rate	Continuous			

3.3 Model 1 (Artificial Neural Network Based Decision Tree)

With DT, predictions are done with a historical dataset. DT is structured in a tree-like manner that has leaves, branches and internal nodes. The class label is represented with leaf, outcome represented by branches and all attributes are being represented by the internal node. DT as used in this section was used to classify the NSL-KDD with 41 features into “attack group” and “normal group”. The normal group is classified as “+1’s” while the attack group is classified as “0’s”. The output of DT serves as an input into ANN. When different models are ensembled, there is greater accuracy in the prediction. Basically, ANN based DT is used to classify the attack group into the various four different groups. There are three sections in the ANN which are input, hidden and the output layers. Outputs from the DT are inputted into the ANN. Since there are four groups in the NSL-KDD dataset as depicted in Table 2, the attack group is also classified into four categories. If the final output gives 0001, it is DOS, if it gives 0010, it is classified into R2L, if it reads 0100 it is U2R. Finally, if it reads 1000, it will be classified as probes.

Each threat feature is used as the input variable which is determined by

$$M_i = (M_1 M_2 M_3 \dots \dots M_n) \dots \dots \dots \dots \dots \quad (1)$$

Where i depict a number of variables. Synaptic weight on the input neuron at layer p is depicted in the equation below:

$$Z_P = W_{1P} M_1 + W_{2P} M_2 + \cdots \dots \dots \dots . W_{3P} M_3 + b \dots \quad (2)$$

The output of a threshold [+1, 0] is limited by the application of sigmoid function. Square error measure (E) is used to measure the difference between the actual output (x) and expected output (y)

$$E = (Y - X)^2 \dots \dots \dots \dots . \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots . . (3)$$

It is necessary to calculate the network's weight with respect to square error function. Exponential to 2 is canceled by $1/2$ when differentiating so as to redefine the square error function.

$$E = 1/2(Y - X)^2 \dots \dots \dots . \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots . (4)$$

Each neuron P, is defined by output α_P

$$\alpha_P = Q(net_p) = Q \sum_{k=1}^{n} W_{ip} M_i \dots \dots \dots \dots \dots \dots \dots \dots \dots . (5)$$

The activation function θ is differentiated and the derivative of the equation (2) is;

$$\frac{d\theta}{dz} = \theta(1 - \theta) \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \quad (6)$$

Chain rule was used twice for partial derivative of error (E) with respect to weight W_{iP}

$$\frac{dE}{dw_{ip}} = \frac{dE}{d\alpha_p} \quad \frac{d\alpha_P}{dnet_p} \quad \frac{dnet_p}{dw_{ip}} \quad \dots \dots \dots \dots \dots \dots \quad (7)$$

Terms on left hand side can be calculated from Equation 5

$$\frac{dnet_p}{dw_{iP}} = \frac{d}{dw_{ip}} (\sum_{k=1}^{n} w_{ip} M_i) = M_i \quad \dots \dots \dots \dots \dots \dots \dots \quad (8)$$

$$\frac{d\alpha_p}{dnet_p} = \frac{d}{dnet_p} \alpha(net_p) = \alpha(net_p)(1 - \alpha(net_p) \quad (9)$$

If X is the outer layer such that $X = \alpha_P$, then the first term is calculated by differencing error function in Equation 4

$$\frac{dE}{d\alpha_p} = \frac{dE}{dx} = \frac{\frac{d}{dx} 1}{2(Y - X)} = X - Y \quad \dots \dots \dots \dots \dots \quad (10)$$

The recursive expression for the derivative is derived as follows when E is given as a function of all the input neuron

$$\frac{dE}{d\alpha_p} = \sum_{i \in l} \left(\frac{dE}{dnet_i} \frac{dnet_i}{d\alpha_p}\right) = \sum_{i \in l} \left(\frac{dnet_i}{dnet_p} \frac{d\alpha_i}{dnet_i} W_{ip}\right) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (11)$$

Table 2: Different attacks group

S/N	Attack Group	Different attacks	Output
1	Denial of service attack	Ping of death, Back, SYN Flood, Process table, Apache2, Land and Mail bomb.	0001
2	Root to local	Satan, Ipsweep, Saint, Mscan and Nmap	0010
3	User to root	Loadmodule, Perl, Fdformat and Xterm	0100
4	Probes	Dictionary, Guest, FTP write, Xlock and Imap	1000

3.4 Model 2 (Bayesian Based ANN model)

This is the second ensemble model in the perception phase of SA. It is the combination of ANN and Bayesian models. There are three sections of ANN and they include input layer, hidden layer which uses sigmoid activation function and the output layers. NSL-KDD dataset serves as input into the ANN which classified the dataset into the "normal group" and the "attack group". The normal group is being classified as "+1's" while the attack group is classified as "0's". The output of ANN serves as input into the Bayesian model, with this Bayesian based ANN model being formulated. Bayesian based ANN models use probability theory to depicts uncertain knowledge.

This theory is derived from Bayes Theorem as follows;

$$P\left(\frac{X}{S}\right) = \frac{P\left(\frac{X}{S}\right) * P(X)}{P(S)} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (12)$$

Where X is the input variables of the dataset and S represents the attack groups which can be re-written as

$$P\left(\frac{X}{S}\right) = P\left(\frac{X_1}{S}\right) * P\left(\frac{X_2}{S}\right) * \ldots\ldots\ldots * P\left(\frac{X_n}{S}\right) \ldots\ldots\ldots\ldots . (13)$$

Since we are solving for S, P(x) which depicts the feature of the dataset is a constant, it can be removed from the equation and proportionality introduced. This gives the equation below;

$$P\left(\frac{S}{X}\right) \alpha\ P\left(\frac{X}{S}\right) * P(S) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots . (14)$$

OR

$$P\left(\frac{S}{X}\right) \alpha\ P(S) * \prod_{i=1}^{n} P\left(\frac{X_i}{S}\right) \quad (15)$$

The goal of Bayesian is to choose class S that has the maximum probability equ (14). Argmax is introduced into equ (15) because it gives the maximum value from a group targeted.

Table 2 depicts the attack group as classified by Bayesian based ANN models.

$$S = argmax_S[P(S) * \prod_{i=1}^{n} P\left(\frac{X_I}{S}\right) \ldots\ldots\ldots\ldots\ldots\ldots (16)$$

3.5 Model 3 (J48 Based Naïve Baye Model)

NB uses Bayes' principle that its assumption is based on robust independence among attributes, it is one of the ML algorithms that has been used by various researchers. NB is firstly used in the first phase of model 3 to classify the NSL-KDD dataset into an attack group or the normal group by using a probabilistic technique. J48 based NB model is used in the second sub-phase of model3 in the perception phase of SA. J48 is known to be an extension of ID3 and it is ML algorithm that has some features such as DT pruning, derivative of rules, accounting for missing values and continuous attribute value range. J48 also has internal nodes and branches as that of DT. J48 classifies the attack group from NB into the respective attack category as shown in Table 2

3.6 Model 4 (Decision Tree (DT)-based Bayesian Network (BN))

Bayesian model is used in model 4 as the first classifier in the perception phase of SA. Bayesian deploys probability theory for its prediction. As used in this sub-phase, it classifies the NSL-KDD dataset into the attack group and the normal group. The output from Bayesian as shown in the proposed mode (Figure 1) serves as input into DT. The formulated DT based BN basically has three components which include the decision node, branch and a leaf. Decision node is used for test attribute identification, while the branch is used for decision making based on the attribute identified. Finally, the leaf is used to classify the attack group into their respective attack category. Table 2 depicts the various categories of threats.

3.7 Model 5 (Random Forest (RF) based on Support Vector Machine (SVM))

Model 5 is an ensemble of SVM and RF. SVM is first used to classify the NSL-KDD dataset. SVM works by using the hyperplane to group the points in the space into two categories "the attack group" and "normal group". Optimum result is achieved only when equ (17) is at the maximum level which is also subjected to equ (18) and equ (19) respectively.

$$\min \frac{1}{2} \| W \|^2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (17)$$

$\propto_x \beta_x \ldots\ldots\ldots (\alpha_y \beta_y), \beta \in (1,0)\ where\ \alpha_x \beta_x \ldots\ldots\ldots (\alpha_y \beta_y)$ are trained data. y represents the number of samples, β is in category (0 and 1), W depicts the weight of the input, the bias represented by b, finally, the input features is represented by α.

The formular for the categories is represented below

$$(W.\alpha) + b \geq \beta_i\ if\ \beta_i = 1 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (18)$$

$$(W.\alpha) + b \leq \beta_i\ if\ \beta_i = 0 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (19)$$

The output of SVM serves as an input into RF. The ensembled RF based SVM uses a decision tree to classify the attacks into the various category groups using majority votes of the predicted value. The various category groups are depicted in Table 2.

3.8 Performance Evaluation

All the five different ensembled models were evaluated after simulation. The following metrics were used to evaluate the models;

True Positive (TP): defines events that are positive and are predicted as positive

False Positive (FP): Defines the event that are negative but are falsely predicted as positive events

True Negative (TN): Describe events that are negative and are predicted as negative events

False Negative (FN): predict events that are positive as negative to the network administrator

Recall: Defines the quantity and completeness of the models

$$Recall = \frac{TP}{TP + FN} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (20)$$

Precision: Defines the quality and exactness of the models

$$Precision = \frac{TP}{TP + FP} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (21)$$

Accuracy: Defines the effectiveness of the models

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \ldots\ldots\ldots\ldots\ldots \quad (22)$$

The confusion matrix for the performance evaluation parameters is shown in Figure 2

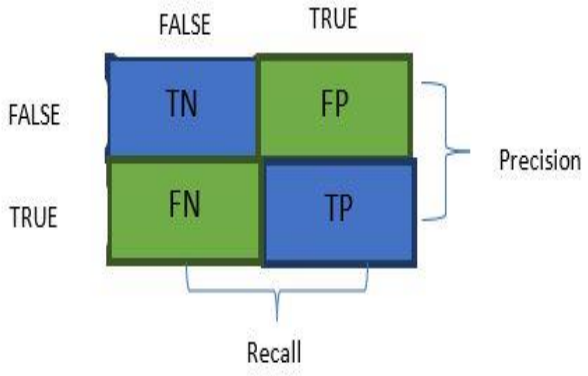

Figure 2: Confusion Matrix for Evaluation

4. EXPERIMENTAL SETUP

All model building and evaluation was done in Waikato Environment for Knowledge Analysis (WEKA). WEKA is a simulating software that has been widely used by various researchers. All the ensemble models were formulated in a WEKA environment. The NSL-KDD dataset used was downloaded online which has forty-one attributes. The dataset was preprocessed in an excel environment and saved in CSV format. For the dataset to be used in WEKA, it was converted into arff format. After the dataset has been accepted in the Weka environment, five different categories of ensemble algorithms were formulated and evaluated with the metrices earlier discussed.

For model training and testing, a 10-fold cross validation was used. The dataset was partitioned into 10 different samples, nine out of the samples were used in model training and the remaining one was used for model testing. This was done individually for all the ensembled algorithms. TP, FP, FN and FP for all the ensemble algorithms were displayed in the WEKA environment after model building, testing and evaluation. Performance of each ensemble model was then compared after evaluating the models.

4.1 Results and discussion

This section discusses the result of each ensemble model used in the perception phase of SA model for threat detection. It also displayed the effectiveness and efficiency of ensemble models for threats detection. The ensemble techniques used include the following; Artificial Neural Network based Decision Tree, Bayesian Network based Artificial Neural Network, J48 based Naïve Bayes, Decision Tree based Bayesian Network and Random Forest based Support Vector Machine. The results of Model 1 (ANN based DT Model) are firstly discussed.

ANN based DT model gave the following results as displayed in the confusion matrix in Figure 3. TN: 26505, TP: 32105, FP: 70, FN:92, Precision: 99.54%, Recall: 98.40% and Accuracy; 98.87%. While BN based ANN Model (Model 2) gave the following results; TN: 26753, TP: 32362, FN: 92, FP: 70. Precision: 99.78%, Recall: 99.71 and Accuracy 99.72% as displayed in Figure 4. Furthermore, Figure 5 depicts the confusion matrix for J48 based NB model (Model 3) and the following were obtained TN: 26513, TP: 32112, FN:519, FP:133, Precision: 98.40%, Recall 99.58% and Accuracy:98.90%.

Figure 6 shows the confusion matrix for DT based BN model (Model 4) and the results obtained were TN: 23853, TP: 29452, FN:3159, FP:2813, Precision: 91.28%, Recall 90.31 and Accuracy 89.92%. Finally, RF based SVM model (Model 5) gave the following results; TN: 26363, TP: 31968, FP: 280, FN: 666, Precision: 98.94%, Recall: 97.53% and Accuracy: 98.40 as depicted in Figure 7

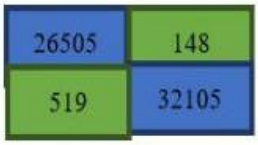

Figure 3: Confusion matrix for model 1

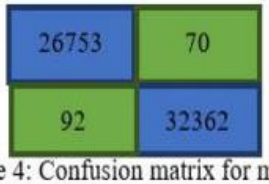

Figure 4: Confusion matrix for model 2

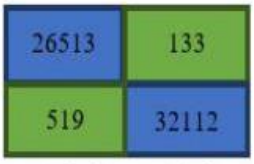

Figure 5: Confusion matrix for model 3

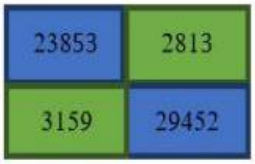

Figure 6: Confusion matrix for model 4

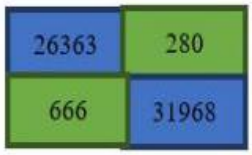

Figure 7: Confusion matrix for model 5

4.2 Performance Comparison of all Models

Table 3 depicts the performance of all the ensembled algorithms used in this research. It was observed that BN based ANN model (Model 2) gave the best accuracy as compared to other ensemble algorithms. Also, it was discovered that DT based BN model (Model 4) gave the least accuracy as compared to other ensemble algorithms used in this research work. Figure 8 depicts the bar chart for the performance metrics

Table 3: Performance Evaluation of all the Five Different Models

Classifiers	No of instances	TN	TP	FN	FP	Precision	Recall	Accuracy
ANN based DT (Model 1)	59,277	26505	32105	519	148	0.9954	0.9840	0.9887
BN based ANN (Model 2)	59,277	26753	32362	92	70	0.9978	0.9971	0.9972
J48 based NB (Model 3)	59,277	26513	32112	519	133	0.9840	0.9958	0.9890
DT based BN (Model 4)	59,277	23853	29452	3159	2813	0.9128	0.9031	0.8992
RF based SVM (Model 5)	59,277	26363	31968	666	280	0.9894	0.9753	0.9840

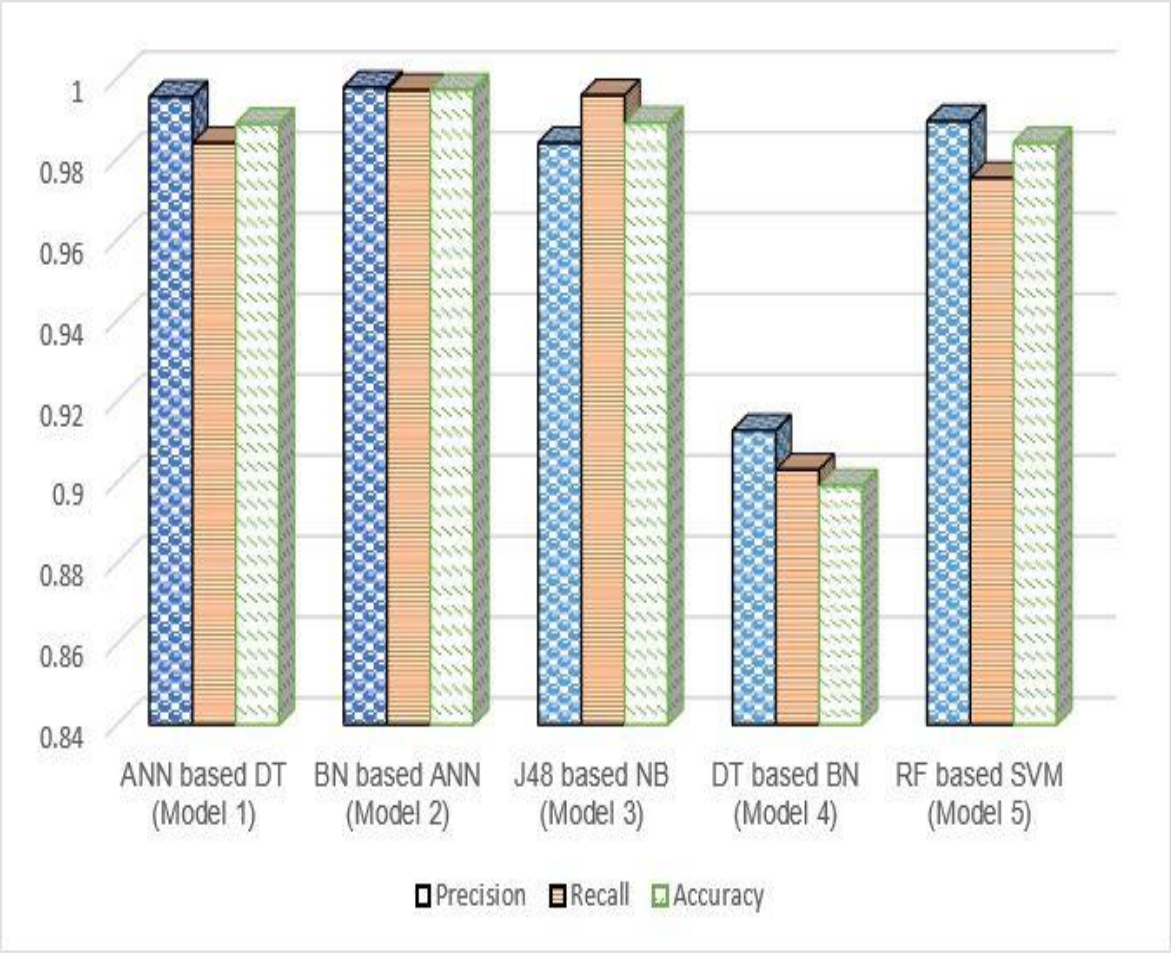

Figure 8: Performance comparison of all the models with respect to precision, recall and accuracy

5. CONCLUSIONS

Accessibility of cyber threats to a system can be reduced by an ensembled algorithm as shown in this study. A more formidable IDS is needed because of the rate at which cyber-threats are gaining access and billions of dollars are being stolen and other damages being carried out. Solving this great problem will be a relief to industries, companies and IT personnel. NSL-KDD online dataset used has various features and because of this some of the algorithms cannot classify them correctly. This research used five different possible ensemble algorithms for threats detection and clearly, the best algorithm for the detection was identified. BN based ANN model (Model 2) gave 99.72% accuracy over all other ensemble algorithms. This suggests that the BN based ANN model will be more suitable for threat detection in the perception phase of SA model. With this result, a network administrator will find it useful when an automated system is needed to be developed. Future work will include getting more cyber-security online datasets for an ensemble model building and training which will be used to design an IDS automated system.

REFERENCES

[1] Sarker, H., Kayes, A., Badsha, S., Alqahtani, H., Waters, P. and Alex, N. (2020). Cybersecurity data science: an overview from machine learning perspective. Journal of Big data, 2020.

[2] Sarker, H. Abushark. Y., Alsolami, F. and Khan, A. (2020). Intrudtree: a machine learning-based cyber security intrusion detection model. Symmetry, 12:754-761.

[3] Hao, Z., Feng, Y., Koide, H. and Sakurai, K. (2020). A sequential detection method for intrusion detection system based on artificial neural networks. International Journal of Networking and Computing, 10:213-226

[4] Olofintuyi, S.S., Omotehinwa, T. O., Odukoya, O.H. and Olajubu, E. A. (2019). Performance comparison of threat classification models for cyber-situation awareness. Proceedings of the OAU Faculty of Technology Conference, 305-309.

[5] Olofintuyi, S.S. (2021). Cyber Situation Awareness Perception Model for Computer Network. International journal of advanced computer science and application. 12(1): 392-397.

[6] Sarker, H., Kayes, A. and Watters, P. (2019). Effectiveness analysis of machine learning classification models for predicting personalized context-aware smartphone usage. Journal of Big Data

[7] Mohammadi, S., Mirvaziri H., Ghazizadeh-Ahsaee, M. and Karimipour, H. (2019). Cyber intrusion detection by combined feature selection algorithm. Journal of Information Security and Application, 44: 80-88

[8] Shams, E. A., and Rizaner, A. A.(2018). Novel support vector machine- based intrusion detection system for mobile ad hoc networks.Wireless Networks.

[9] Sarker, H. (2019). A machine learning based robust prediction model for real-life mobile phone data. Internet of Things, 5:180-193

[10] Sarker, H., Colman, A., Han, J.Khan A., Abushark, Y. and Salah, K. (2019). Behavdt: A behavioral decision tree learning to build user-centric context-aware predictive model. Mobile Network. Application. 1:1-11

[11] Sarker, H. and Salim, F. (2018). Mining user behavioral rules from smartphone data through association analysis. Springer, 10: 450-461.

[12] Endsley M. (1995). Toward a theory of situation awareness in dynamic systems. In Human Factors Journal, 37: 32-64.

[13] Aslahi-Shahri, B.M. (2016). A hybrid method consisting of GA and SVM for intrusion detection system. Neural computing and applications, 27:1669-1676.

[14] Chen, Y.H., Horny S.J. and Su, S.J. (2011). A novel intrusion detection system based on hierarchical clustering and support vector machines. Expert systems with Applications, 38:306-313

[15] Harbi, N., Rahman, C. M. and Farid, D.M. (2010). Attacks classification in adaptive intrusion detection using decision tree.

[16] Sahu, S and Mehtre B.M. (2015). Network intrusion detection system using j48 decision tree. International Conference on Advances in Computing, Communications and Informatics (ICACCI). IEEE, 2023-2026,

[17] Jukic, S., Kerric J. and Subasi, A. (2017). An effective combining classifier approach using tree algorithms for network intrusion detection," Neural Computing and Applications, 28:1051-1058

[18] Wang, G. (2010). A new approach to intrusion detection using artificial neural networks and fuzzy clustering. Expert systems with applications, 9: 46-52.

[19] John G. H. and Langle, P.(1995). Estimating continuous distributions in Bayesian classifiers. Proceedings of the eleventh conference on uncertainty in artificial intelligence, Morgan Kaufmann Publishers Inc. 338.
[20] Witten, I.H. and Frank, E.(2005). Data mining: Practical machine learning tools and techniques.
[21] Keerthi, S., Shevade, S.K., Bhattacharyya, C. and Murthy, K. (2001). Improvements to platt's smo algorithm for SVM classifier design. Neural comput. 13: 637-649.
[22] Holte, R. (1993). Very simple classification rules perform well on most commonly used datasets. Mach Learn, 11: 63-90.
[23] Aha, D. W., Kibler, D. and Albert, M. (1991). Instance-based learning algorithms, Mach Learn, 6:37-66.
[24] Cessie, S. and Houwelingen, J. (1992). Ridge estimators in logistic regression. J Royal Stat Soc C. 41:191-201.
[25] Freund Y. and Schapire, R. (1996). Experiments with a new boosting algorithm. Icml, 96:148-156.
[26] Watters, P. A., McCombie, S., Layton, R. and Pieprzyk, J. (2012). Characterising and predicting cyber-attacks using the cyber attacker model profle (camp). J Money Launder Control.
[27] Keerthi, S. S., Shevade, S. K. Bhattacharyya, C. and Murthy, K. (2001). Improvements to platt's smo algorithm for svm classifier design. Neural Comput. 13:637-649.
[28] Han, J., Pei, J., and Kamber, M. (2011). Data mining concepts and techniques.
[29] Breiman, L. (2011). Random forests. Mach Learn. 45:5-32.
[30] Rokach, L. (2010). A survey of clustering algorithms. In: Data Mining and Knowledge Discovery Handbook. Springer; 269-298.
[31] MacQueen, J, (1967). Some methods for classification and analysis of multivariate observations. In: Fifth Berkeley symposium on mathematical statistics and probability.
[32] Ashfaq, R.,Wang, X., Huang, J., Abbas, H. and He, Y. (2017). Fuzziness based semi-supervised learning approach for intrusion detection system. Inf Sci, 378:484-497.
[33] Lyngdoh, J., Hussain, M. I., Majaw, S. and Kalita, H. K. (2018). An intrusion detection method using artificial immune system approach. International conference on advanced informatics for computing research, Springer, 379-397.
[34] Goldstein, M. (2012). An expectation-maximization based local outlier detection algorithm. Pattern recognition 21st international conference, IEEE, 2282-2285.
[35] Rath, P. S., Barpanda, N. K., Singh, R. and Panda, S.(2017). A prototype Multiview approach for reduction of false alarm rate in network intrusion detection system. International Journal of Computational Network Communication Security, 5:49-55.
[36] Aburomman, A. A. and Ibne, M.B. (2016). A novel SVM-kNN-PSO ensemble method for intrusion detection system. Appl Soft Comput, 38:360-372.
[37] Jabbar, M. A., Aluvalu, R. and Reddy, S.S. (2017). A Novel Ensemble Intrusion Detection System. Procedia Computer Science, 115:226-234.