A Stacked Ensemble Intrusion Detection Approach for the Protection of Information System

Olayemi O. Olasehinde
Department of Cyber Security
Federal University of Technology, Nigeria

Abstract

Cyber attackers daily works round the clock to compromise the availability, confidentiality and integrity of information system, protection of information system has been a great challenge to network administrators. Intrusion detection system (IDS) analyse network traffics to detect and alert any attempt to compromise the computer systems and its resources, stacked ensemble build synergy among two or more IDS models to improved intrusion detection accuracy. This research focus on the application of stacked ensemble to the development of enhanced Intrusion Detection Systems (IDS) for protection of information system. Relevant features of the UNSW-15NB intrusion detection dataset were selected to train three base machine learning algorithms; comprising of K Nearest Neighbor, Naïve Bayes' and Decision Tree, to build the base-predictive models. Decision Tree model with features selected by Information Gain features selection technique, recorded the highest classification accuracy on evaluation with the test dataset. Three meta algorithms; Multi Response Linear Regression (MLR),Meta Decision Tree (MDT), and Multiple Model Trees (MMT) were trained with the predictions of base predictive models to build the stacked ensemble. Python programming language was used for the implementation of the ensemble models. The stacked ensemble recorded, improved classification accuracy of 3.0% over the highest accuracy recorded by the base models and 5.11% above the least accuracy recorded by the base models. False alarm improvement of 0.89% and 3.29% were recorded by the stacked ensemble over highest and least false alarm recorded by the base models respectively. The evaluation of this work shows a great improvement over reviewed works in literature

1. Introduction

Security of sensitive information has been a great challenge to several organisation, a system is considered secured if the three principles of computer security; Confidentiality, Integrity and Availability (CIA), are successfully satisfied, according to [1], Cyber attackers daily works round the clock to compromise information system and steal sensitive information. An attack is a threat against an information security that attempts to acquire, modify, obliterate, delete, confiscate, denial access to or disclose information without consent or authorization [2]. Cyber attacks against information system includes;

i. Partly or wholly compromise of the information system integrity
ii. Exhaustion of computer resources needed by the system to perform it functions
iii. Stealing of valuable data and information
iv. Complete system hijack and prevention of legitimate users access to the information system

[3] defines an intrusion as any process that bypass authentication or subvert access control procedures, it is the gateway through which external cyber attackers gain an authorized access to computer network. An intrusion detection system (IDS) is a system that scrutinizes every packets passing through a network to check for intrusive packets and issues alerts when such packets are discovered. According to [4], steady rise in the number of recorded cyber attacks due to the growing numbers interconnected networks to the internet, necessitates the need for an effective intrusion detection system, it is an indispensable device for network administrators because without it, it will be impossible to analyse the huge amount of packets traversing networks per second. It is a powerful defense mechanism against the hostile threats of cyber attackers [5].

A lot of information system seems secured when it is used in a standalone mode, but become vulnerable when connected to the internet, the presence of flaws or bugs in the design of information system are often capitalized upon by intruders to gain an unauthorized access into such system. Detection of cyber-attacks in computer network is essential for the protection of cyber infrastructures from cyber-attackers.[6].

The accuracy of the attack detection in the cyber systems depends heavily on the completeness of the collected sensor information or dataset. The effectiveness of IDS is measure by its ability to correctly identify attacks and normal traffics from the incoming network packet or a new instance of intrusion dataset, and it is based on;

i. Integrity of the intrusion dataset that represents the existing networks and it associated attack scenarios.
ii. Machine learning algorithm used to build the system

Machine Learning (ML) is an Artificial intelligent (AI) approach, that provides systems the ability to learn patterns of attacks and normal packets from the intrusion dataset and use the acquired intelligent to build Intrusion Detection System. Supervised Machine learning algorithms extract valuable knowledge from the mapping of supplied inputs and its desired output of the training dataset, then validates the obtained knowledge on the testing dataset. Regression and classification are supervised machine learning techniques. Unsupervised learning draws knowledge from the datasets consisting of input data without label responses, it partitions the dataset into clusters base on the similarities that exists among them, and validates by assigning a new test instance into the appropriate cluster, clustering analysis and association mining are examples of unsupervised learning method.

An ensemble learning (methods) are supervised machine learning process used to get an improved prediction accuracy, by strategically combining the predictions from multiple learning algorithms sequentially or in parallel. *Sequential* ensemble methods generates and combines base learners sequentially exploiting the dependence between the base learners., parallel ensemble methods generate and combines single learners (homogeneous ensemble) or multiple learners (heterogeneous ensemble) in parallel exploiting independence between the base learners.

Bagging, Boosting and Stacking are three ensemble combining techniques. Bagging also known as Bootstrap Aggregation is an homogeneous parallel ensemble techniques that evaluates base learner on sampling of a small subsets of original dataset, drawn with replacement and then determine their predictions by voting or simple averaging to generate final prediction. Boosting is an homogeneous and sequential ensemble technique that use weighted average to adjust misclassified instance and dictates what features the next model will focus on. Stacked ensemble is an heterogeneous, parallel ensemble that involves training a second-level "meta-learner" to find the optimal combination of the diverse sets of base models predictions.. This paper focuses on the protection of information system by applying stacked ensemble to improves the detection accuracies of IDS systems.

2. Literature Review

The methods used in the design of an intrusion detection ranges from single to hybrid and to ensemble, single method use one classification algorithm in it detection engine, hybrid method make use of two classification algorithms in it developments of the detection model while ensemble methods involves the use of two or more classification algorithm in the development of the detection model. Ensemble methods has proven to increase the classification accuracy and reduces wrong classification rate better than the other methods [7]. Over the years, how to combines several models in other to improve their performances accuracy using meta classifier such as stacking, boosting and bagging has been an area of research. The result of study in [8] shows that combining the results of several classifiers yielded a more accurate results than the best of the individual classifiers, the theoretical study in [9] showed that performance accuracy of weaker classifiers can be improved to the level and more than the of stronger classifiers using ensemble. Reference in [10] shows that ensemble can be used to improve the quality and robustness of clustering algorithms.

Ensemble methods have been found very appropriate to solve computer security problems because each activity performed on computer systems can be observed at multiple abstraction levels, and the relevant information may be collected from multiple information sources [11]. Choudhury and Bhowal builds several boosting ensemble for intrusion detection using of many ML algorithms, and concluded that Random forest and BayesNet are the two most suitable algorithms in terms of classification accuracy to build IDS model [12]. [13] proposed a Particle Swarm Optimization (PSO) for feature selection for an ensemble of three base classifiers; (Classification and Regression Tree - CART, Random Forest- RF and C4.5 Decision tree), the implementation of ensemble system showed a promising accuracy and lower false alarm rate than existing ensemble techniques. Reference in [14] Developed a Bagging, Boosting, and Stacking Ensemble Intrusion Detection models of four base models predictions; Naive Bayes, K nearest Neighbour, Decision Tree and Jrip Induction to improve their classification accuracy and reduce their false alarm rate using NSL-KDD Intrusion Detection Dataset, the stacked ensemble achieved an accuracy of 99% and 60% with known attacks and unknown (novel) attacks respectively. [15] Used stacked Ensemble to combines and improves the predictions of three base learners; K Nearest Neighbour, Naive Bayes and Decision Tree which served as input to Multiple Model Tree (MMT) meta learner algorithm individually and is collectively evaluated via ten folds cross validation to build the stacked ensemble model used for classifications of the network traffics into nine network attacks and normal. The results from this research showed that MMT stacked model of the three base learner predictions gives a better multi-class classification predictions accuracy than the best accuracy recorded by the three base models, it recorded the highest classification accuracy of 97.93% and lowest false alarm rate of 0.22% for the binary (attacks and normal label).

3. UNSW-NB15 Intrusion Detection

Attacks Categories

Datasets play an important role in the testing and validation of any intrusion detection method, most IDS research had used the KDD datasets, which were compiled in 1998 and 1999 [16], [17]. These datasets represented the first systematic approach to IDS data generation, and were an incredibly valuable and innovative resource at the time of it release. Over time, these datasets have lost most of their relevance through a natural aging process, and do not reflect the current real world trends, but are still being used by researchers for validation of Intrusion Detection System due to lack of viable alternative dataset. The UNSW-NB15 dataset has now provided a better alternatives to the use of KDD and NSL-KDD intrusion dataset, it gives an accurate evaluation and a better performance than NSLKDD and KDDCUP on different learning techniques. [18], The UNSW_NB 15 is the latest published dataset which was created in 2015 for research purposes in intrusion detection from the Cyber Range Laboratory of the Australian Centre for Cyber Security (ACCS), it is an hybrid of the realistic modern normal activities and the synthetic contemporary attack behaviours from network traffics.

The advantages of UNSW-NB15 dataset over NSLKDD includes; First, it contains real modern normal behaviors and contemporary synthesized attack activities. Second, the probability distribution of the training and testing sets are similar. Third, it involves a set of features from the payload and header of packets to reflect the network packets efficiently. Finally, the complexity of evaluating the UNSWNB15 on existing classification systems showed that this dataset has complex patterns, this means that the dataset can be used to evaluate the existing and novel classification methods in an effective and reliable manner[18]. The training and testing sets are made up of 82,332 and 175,341 records respectively, Table 1, shows the distribution and description of it nine attack categories and normal connections.

Table 1: Description of the attacks types in the UNSW-NB15 training and testing dataset

Type	No of Records	Description
Normal	93,0000	Natural transaction data
Analysis	2,674	This is a port based penetration intrusion attacks against web application
Backdoors	2,329	This is a penetration remote attacks to gain unauthorized access to a system
DoS	16,353	This is attack denials legitimate network users of their right to the system resources such as memory, storage space, needed to perform their computing operations
Exploits	44,525	This is a penetration attacks that use a sequence of instructions or code to takes advantage of a glitch, bug, or vulnerability in the operating system.
Fuzzers	24,244	This is a scan attacks that scan victim system using software testing technique to discover flaws and security loopholes in a program, operating system, or network by bombarding with several request to makes it crash.
Generic	58,871	This is a penetration attacks that A techniques works against all block-ciphers (with a given block and key size), without consideration about the structure of the block-cipher
Reconnaissance	13,987	This is a probe attack that that gathers information about a computer network to evade its security controls
Shellcode	1,511	This is a penetration attacks that make use of a Small program with instructions from a shell to compromised the victim's computer
Worms	174	This is a scan attacks that is self replicating malicious code attack that that spread itself to other computers, mostly over a computer network, without attaching itself to a program like a virus

4. Methodology

Figure 1 shows the architecture of the Stacked Ensemble Network Intrusion Detection Approach for Information System Security.. Discretisation of the UNSW-NB15 intrusion detection dataset was carried out to make it suitable for stacked ensemble intrusion detection model building and evaluation using class attribute interdependent maximization (C). Discritised variable D for attribute F of the dataset is given in equation 1

$$(C, D \mid F) = \frac{\sum_{i=1}^{n} \frac{max_i^2}{m_{ir}}}{n} \quad (1)$$

where n is the users predefined number of intervals, i iterates through all intervals, that is. i=1,2,...,n, maxi is the maximum value within the ith column, Mir is the total number of continuous values of attribute F.

Network packets consist of multiple features some of which has no relevance (correlation) to the determination of the target feature, these irrelevant features are the major reasons of increasing the false alarm rate (FAR) and decreasing the detection rate. Feature Selection (FS) techniques are used to determine the relevant features in a dataset, in other

to obtain a higher classification accuracy of the stacked ensemble system, three filter-based feature selection techniques; correlation, consistency and information gain were used to extract relevant feature subset of the dataset used to build the stacked ensemble models. The features subset with the highest merit will be selected and returned by correlation techniques as shows in Equation 2

$$M_s = \frac{k\,\bar{r}_{cf}}{\sqrt{k+k(k-1)\bar{r}_{ff}}} \quad (2)$$

where $\bar{r}_{cf}$ is the average attack categories to features, $\bar{r}_{ff}$ is the average features to features correlations and k is the number of features in the subset S

Given a training sample S the inconsistency count(IC) of an instance subset $A \in S$ is given in equation 3

$$IC_{X'}(A) = X'(A) - \max_k X'_k(A) \quad (3)$$

Where $X'(A)$ is the number of instances in S equal to subset A using only the features in $X'$and $X'_k(A)$is the number of instances in S of class k equal to A using only the features inX'.

Inconsistency features techniques select and returned features subset with the lowest of inconsistency rate shown in equation 4

$$IR(X') = \frac{\sum_{A \in S} IC_{X'}(A)}{|S|} \quad (4)$$

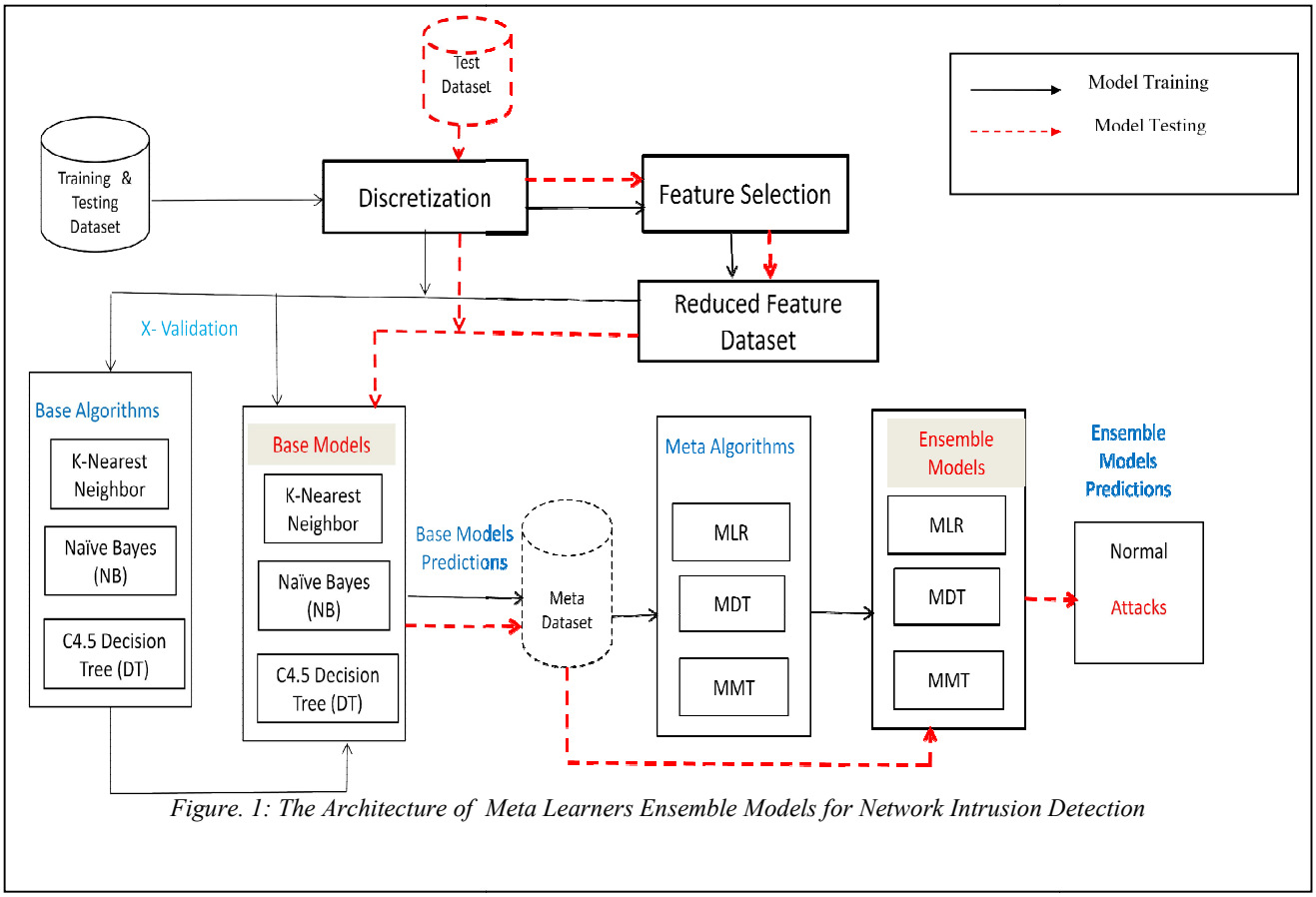

Figure. 1: The Architecture of Meta Learners Ensemble Models for Network Intrusion Detection

Information Gain attributes selection scored and ranks attributes based on their information gain in respect to the target feature (Y), Information Gain (IG) for attribute x is given in equation 5

$$IG(X) = H(Y) - H(Y \mid X) \quad (5)$$

Where H(Y) is Entropy of Y as shown in equation 6 and H(Y | X) is Entropy of Y given X as shown in equation 7

$$H(Y) = -\sum_{i=1}^{n} p(y_i)\, log_2\, p(y_i) \quad (6)$$

$$H(Y \mid X) = -\sum_{i=1}^{n} p(x_i) \sum_{j=1}^{k} H(y_j \mid x_i) log_2\, p(y_j \mid x_i) \quad (7)$$

Where n: is number of instance in the intrusion dataset. k: is the number of attack categories in the intrusion dataset. $P(y_i)$: is the probability of occurrence of attack categories value of instance i and $P(y_i|x_i)$: is the probability of attack categories value of instance i will occur given the occurrence of attribute value x of instance i

Stacked ensemble framework consist of two phases; In the first phase, the UNSW-NB15 dataset S, consisting of instances of the form $s_i = (x_i, y_i)$ where x_i and y_i represents feature vector and attack categories respectively wasbused to train machine learning algorithms L_1, . ,L_k to build base classifiers C_1, C_2,....,C_k, where $C_i = L_i(S)$. K-Nearest Neighbor, Naïve Bayes and C4.5 Decision tree machine learning algorithms were used to build the base classifiers. Let p_i and q_t represent the instance to be classified and the other instances in the dataset having the same number of features as P respectively, K- nearest neighbor Euclidean distance between p_i and q_t is defined in equation 8.

$$d(p_i, q_t) = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2} \quad (8)$$

From equation (8), a given instance will be classified as the attack categories having majority attacks among top k closest instance to the given instance. Given the UNSW-NB15 intrusion detection dataset that have X number *of* attributes called the predictors ($X = x_1, x_2,...,x_n$) and another attribute y called the class label, with ten members $y_1,....y_{10}$, the Naive Bayes probability that a class y_j will be assigned to a given unlabeled instance X is given in equation 9.

$$p(y_j \mid x_1, \dots, x_{43}) = \frac{p(y_j) p(x_i \mid y_j)}{p(x_i)} \quad (\forall_j = 0,1, \dots \dots, 9) \quad (9)$$

Maximum posterior probability for classifying a new instance attack categories is given in equation 10

$$y = \arg\max_{y} py_j \prod_{j=0}^{9} p(y_j) p(x_1, x_2, \dots x_{43} \mid y_j) \quad (10)$$

Decision Tree (DT), (C4.5) is a classification model consisting nodes that are attributes names of UNSW-NB15 and arcs which are attribute values connection to other nodes all the way to the leaves, which are the attack categories (class label). Decision Tree (DT) builds a classification tree, which will be used to predict the attack categories of a new instance in the test dataset; DT calculates the Gain Ratio of all the attributes of the training dataset, by dividing the information gain of an attribute with splitting value of that attribute. The formula for Gain Ratio is given in equation 11

$$Gain\ Ration\ A_i = \frac{Information\ \ Gain\ A_i}{Split\ Information\ A_i} \quad (11)$$

Split value of an attribute is chosen by taking the average of all the values in the domain of current attribute. It is given by (12)

$$Split\ info(A_i) = -\sum_{i=1}^{n} \frac{|t_j|}{|T|} \cdot \log_2 \frac{|t_j|}{|T|} \quad (12)$$

Where |T| is the number of values of the current attribute, t is the values of attributes A_i, n is the number of values in attribute A_i

The predictions of the base classifiers of the form $\hat{y}_1$, $\hat{y}_{2,..}$ $\hat{y}_{k.}$, where k is the number of base classifiers formed the meta level dataset $\hat{S}$ consisting of instances of the form $\hat{s}_i = (\hat{y}_i, y_i)$ where $\hat{y}_i$ and y_i represents feature vector of the base classifiers predictions and attack categories respectively.

In the second phase, the meta level dataset $\hat{S}$ will was used to train the meta level algorithm L_i^k using leave-one-out validation technique, by applying the meta algorithm L_i^k to the entire meta level dataset $\hat{S}$, leaving one fold $\hat{s}_i$ out for testing, $C_i^k = L_i^k(\hat{S} - \hat{s}_i)$. The meta classifiers generates predictions of the form $C_i^k(\hat{s}_i)$.

Multi - Response Linear Regression (MLR) is a meta level algorithm adaptation of multiple linear regression represented in equation 13

$$Y = \beta_0 + \beta_1 \hat{y}_{i1} + \beta_2 \hat{y}_{i2} + \beta_3 \hat{y}_{i3} + \dots \beta_k \hat{y}_{ik} \ (i = 1, n) \quad (13)$$

where β_o is the intercept of the regression equation , β_1 is the coefficient of $\hat{y}_{i1,}$ β_2 is the coefficient of $\hat{y}_{i2}$, β_k is the coefficient of $\hat{y}_{ik,}$ k is the number base level classifiers used for attack categories prediction (known), y_i is the correct attack categories for the meta level predicted instances (known), n is the total number of instance in the entire base model predictions.

Multiple Model Tree (MMT) are decision tree with linear regression functions at the leaves, MMT assumed that each base-level classifier predicts a probability distribution over the possible class values. Given meta level training set, a derived dataset for each of the target class were extracted. The derive dataset were used to build linear regression function to obtain the models trees, the model trees were used to induce new instance, the tree with highest value is predicted.

Meta decision trees (MDTs) are a novel method for combining multiple classifiers. MDT leaves specify which base-level classifier should be used, instead of predicting the class value directly, the attributes used by MDTs are derived from the class probability distributions predicted by the base-level classifiers for a given example. A meta decision tree selects an appropriate classifier for a given example in the domain. Consider the subset of examples

falling in one leaf of the MDT. It identifies a subset of the data where one of the base-level classifiers performs better than the others

The stacked ensemble network intrusion detection system was implemented using Python and R Programming Languages. The performance of the stacked ensemble model was evaluated with existing work using standard metrics.

5. Results and Discussion

Feature Selection

Three (3) feature selection methods; Information gain, consistency based method and correlation based were used to identify reduced features attributes that were used to determine the class label (attacks and attacks categories) of UNSW-NB15 dataset, the consistency based method being a subset selector, selected the best subset of the feature attributes that is best in determining the attacks categories, information gain and correlation based were attributes ranking selector, ranking is the process of ordering the features by the value of some scoring function, which measures the relevant of the feature to the determination of the class label.

Three feature selection experiments were carried out by each of the three (3) features selection methods, in the first experiment, the relevant feature attributes that can be used to identify the class label as either attack or normal were determined, in the second experiment, the relevant feature attributes for the multi-class identification of the attacks categories were determined, the third experiment identified relevant features attribute for each attacks categories classification. Table 2 shows the result of the features selection experiment that identified relevant features attributes for determining dataset instance (network packet) as either normal or attack by the three feature selection methods used, Table 3 shows the results of the relevant features that identified the attacks categories; that is multi-class classification by each of the three (3) features selection methods. Table 4 shows the results of relevant feature attributes identified to determine each of the attack types and normal by the three feature selection methods..

Table 2: Relevant features attributes identified for attacks/normal (Binary) classification by each of the three (3) features selection methods

Consistency Subset features selections (26)	Information gain ranked attributes selections (20)	Correlation based ranked attributes selections (15)
dur, service, sbytes, dbytes, rate, sload, dload, sinpkt, dinpkt, sjit, djit, tcprtt, synack, ackdat, smean, dmean, trans_depth, response_body_len, ct_srv_src, ct_dst_ltm, ct_dst_dport_ltm, ct_dst_sport_ltm, ct_dts_src_ltm, ct_src_ltm, ct_srv_dst, attack_cat	attack_cat, sbytes, smean, sload, dbytes, ct_state_ttl, dttl, dmean, sttl, rate, dload, dur, dinpkt, dpkts, synack, tcprtt, ackdat, sinpkt, sjit, spkts	attack_cat, dur, spkts,sttl, ct_dst_sport_ltm, ct_dst_dport_ltm, rate, ct_state_ttl, ct_srv_dst, ct_srv_src, ct_dts_src_ltm, ct_src_ltm, service, proto, ct_dst_ltm

Table 3: Relevant features attributes identified for Multi-class classification by each of the three (3) features selection methods

Consistency Subset features selections (39)	Information gain ranked attributes selections (32)	Correlation based ranked attributes selections (33)
dur, proto, service, state, spkts, dpkts, sbytes, dbytes, rate, sttl, sload, dload, sloss, dloss, sinpkt, dinpkt, sjit, djit, stcpb, dtcpb, dwin, tcprtt, synack, ackdat, smean, dmean, trans_depth, response_body_len, ct_srv_src, ct_state_ttl, ct_dst_ltm, ct_src_dport_ltm, ct_dst_sport_ltm, ct_dst_src_ltm, is_ftp_login, ct_flw_http_mthd, ct_src_ltm, ct_srv_dst, is_sm_ips_ports	dur, proto, service, state, spkts, dpkts, sbytes, dbytes, rate, sttl, dttl, sload, dload, sloss, dloss, sinpkt, dsinpkt, sjit, djit, tcprtt, synack, ackdat, smean, dmean, trans_depth, response_body_len, ct_srv_src, ct_state_ttl, ct_dst_ltm, ct_src_dport_ltm, ct_dst_sport_ltm, ct_dst_src_ltm, ct_flw_http_mthd, ct_src_ltm, ct_srv_dst	dur, proto, service, state, spkts, dpkts, sbytes, dbytes, rate, sttl, dttl, sload, dload, sloss, dloss, sinpkt, dsinpkt, sjit, djit, swin, stcpb, dtcpb, dwin, ct_dst_ltm, ct_src_dport_ltm, ct_dst_sport_ltm, ct_dst_src_ltm, is_ftp_login, ct_ftp_cmd, ct_flw_http_mthd, ct_src_ltm, ct_srv_dst, is_sm_ips_ports

Table 4: Relevant features attributes identified independently for each attacks types classification by the three (3) features selection methods

Network Connection Categories	Consistency Subset attributes selected	Information gain ranked attributes selected	Correlation based ranked attributes selected
Analysis (8)	1, 2, 9, 31, 35, 36, 40, 41	2, 31, 41, 35, 9, 16, 1, 34	1, 14, 13, 41, 15, 16, 17, 18
Backdoor (10)	1, 2, 3, 4, 27, 31, 33, 34, 40, 41	2, 41, 31, 35, 34, 36, 40, 33, 7, 27	1, 14, 13, 11, 15, 16, 17, 18, 19, 12
Dos (24)	1, 2, 3, 9, 10, 12, 13, 16, 17, 18, 19, 22, 24, 25, 26, 27, 29, 30, 31, 32, 33, 36, 40, 41	41, 1, 31, 16, 9, 12, 2, 36, 7, 27, 18, 35, 5, 34, 40, 32, 4, 33, 17, 13, 6, 8, 11, 28	1 , 14 , 13 , 11 , 15 , 16 , 17 , 18 , 19 , 12 , 10 , 41, 3 , 2 , 9 , 4 , 5, 6, 7, 8 , 20 , 21 , 22 , 35
Exploits (27)	1, 2, 3, 4, 9, 10, 12, 13, 16, 17, 18, 19, 21, 24, 25, 26, 27, 30, 31, 33, 34, 35, 36, 37, 39, 40, 41	12, 41, 9, 1, 16, 31, 2, 36, 35, 3, 7, 18, 34, 27, 40, 32, 17, 4, 25, 24, 26, 5, 13, 19, 8, 28, 6	1, 14, 13, 11, 15, 16, 17, 18, 19, 12, 10, 41, 3, 2, 9, 4, 5, 6, 7, 8, 20, 21, 22, 35, 34, 23, 36
Fuzzers (23)	8, 9, 12, 13, 14, 16, 17, 18, 19, 22, 24, 25, 26, 27, 28, 29, 31, 33, 34, 35, 36, 40, 41	7, 27, 2, 36, 12, 31, 41, 1, 9, 16, 40, 35, 8, 28, 33, 15, 34, 13, 17, 14, 5, 6, 24	1, 14, 13, 11, 15, 16, 17, 18, 19, 12, 10, 41, 3, 2, 9, 4, 5, 6, 7, 8, 20, 21, 22
generic (18)	1, 3, 10, 12, 16, 18, 19, 22, 27, 29, 30, 31, 33, 34, 35, 36, 40, 41	36, 35, 34, 33, 31, 41, 40, 1, 9, 16, 12, 3, 7, 27, 2, 18, 5, 4	1, 14, 13, 41, 15, 16, 17, 18, 19, 12, 11, 10, 5, 2, 4, 6, 9, 7
Reconnaissance (23)	1, 2, 3, 4, 9, 12, 13, 14, 16, 17, 18, 19, 24, 25, 26, 27, 29, 31, 34, 35, 36, 40, 41	2 , 9 , 16 , 1 , 12 , 7 , 27 , 36, 35, 34, 40, 24 , 13 , 31, 25 , 19 , 3 , 17 , 41, 29, 39, 18 , 4	1, 14, 13, 11, 15, 16, 17, 18, 19, 12, 10, 41, 3, 2, 9, 4, 5, 6, 7, 8, 20, 21, 22
Shellcode (1)	4	4	4
Worms (2)	2, 3	3, 2	1, 2
Normal (32)	1, 3, 5, 6, 7, 8, 9, 12, 13, 15, 16, 17, 18, 19, 22, 23, 24, 25, 26, 27, 28, 30, 31, 33, 34, 35, 36, 37, 39, 40, 41, 42	7, 8, 32, 28, 10, 11, 13, 27, 9, 26, 24, 25, 17, 12, 6, 1, 16, 5, 15, 18, 19, 14, 33, 36, 40, 3, 41, 4, 31, 30,34, 2	42, 14, 13, 11, 15, 16, 17, 18, 19, 12, 10, 41, 3, 2, 9, 4, 5, 6, 7, 8, 20, 21, 22, 35, 34, 23, 36, 37, 38, 39, 40, 33

Base Models Evaluation of the Test Dataset

Three base algorithms used in this work are; Naive Bayes, K Nearest Neighbor and C4.5 Decision Tree, each of these base algorithms were used to build base classification models with the three (3) reduced dataset generated from the three (3) features selection methods used; (Consistency base, correlation base and information gain feature selection), and the whole feature dataset. Each of the base model were evaluated on the UNSW NB15 testing dataset. Table 5 shows the no of instances of the testing dataset that were correctly and incorrectly classified by the three (3) based models of the three reduced features dataset, and it graphical representation is shown in Figure 2. Table 6 shows the Binary (Attacks/Normal) classification accuracy of the Base models, Naive Bayes recorded the highest classification accuracy of 70.20% with consistency reduced features and least classification accuracy of 56.04% with the whole features set. KNN recorded the highest classification accuracy of 82.35% with correlation reduced feature set, and the least classification accuracy of 75.71% with the whole features set, all the models built with the reduced feature sets performs better that the models built with the whole feature set, this shows that feature selection increases the accuracy of IDS models. Decision Tree with information gain reduced features dataset recorded the highest overall classification accuracy of 87.18% and the least classification accuracy of 75.71% with the with the whole features set, from Table 6, the least classification accuracy of 70.70% recorded by KNN and the 75.71% recorded by Decision tree are higher than the highest classification accuracy of 70.20% recorded by Naive Bayes. The highest classification accuracy of 82.62% by KNN model with information gain reduced feature sets is lower than the classification accuracy Decision Tree models with the three reduced feature set. Graphical representation of Table 6 is shown in Figure 3

Table 5: Testing Dataset Instances that were Correctly and Incorrectly Classified by the Base Models

Classification Based Models	Feature Selection Reduced Dataset	Correctly Classified Instances	Incorrectly Classified Instances	Percentage Classification Accuracy (%)
Naive Bayes	**Whole Dataset**	**98,262**	**77,079**	**56.04**
	Information Gain	122,020	53,321	69.59
	Consistency	**123,081**	**52,260**	**70.20**
	Correlation	117,026	58,315	66.74
K Nearest Neighbors	**Whole Dataset**	**123,959**	**51,382**	**70.70**
	Information Gain	**144,394**	**30,947**	**82.35**
	Consistency	143,868	31,473	82.05
	Correlation	**144,870**	**30,481**	**82.62**
Decision Tress	**Whole Dataset**	**132,243**	**42,598**	**75.71**
	Information Gain	**152,857**	**23,206**	**87.18**
	Consistency	152,135	23,206	86.77
	Correlation	149,560	25,781	85.30

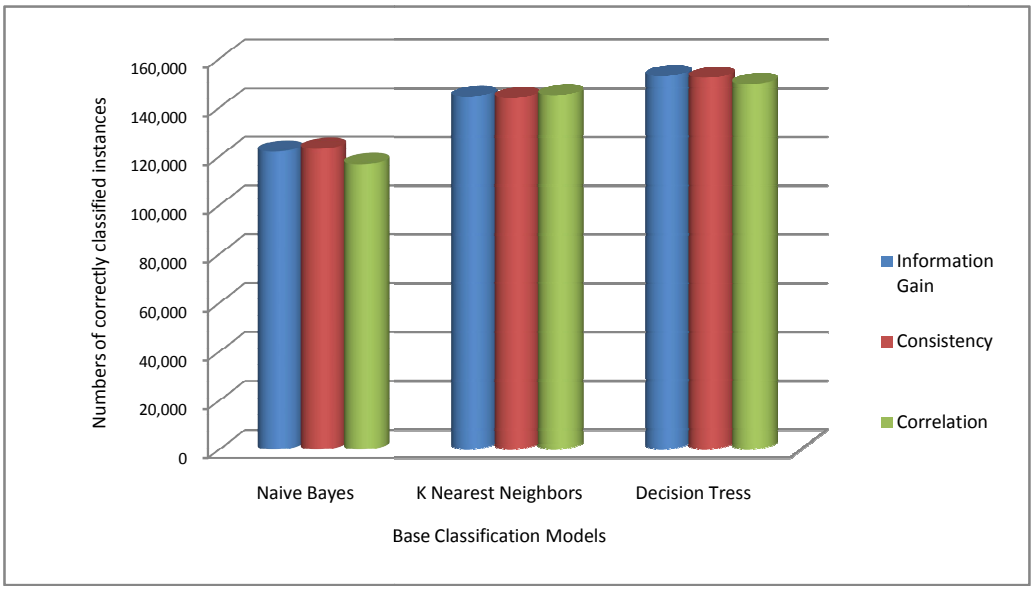

Figure 2: Bar chat of the Base Models Classification Accuracy on the Reduced Dataset

Table 6: Attacks/Normal classification accuracy by the three (3) Reduced Feature Testing Dataset Base Models

Classification Models	Dataset with All Features (%)	Consistency Reduced Feature Dataset (%)	Information Gain Reduced Feature Dataset (%)	Correlation Reduced Feature Dataset (%)
Naïve Bayes	56.04	70.20	69.59	66.74
K Nearest Neigbor	70.70	82.05	82.35	82.62
Decision Tree	75.71	86.77	87.18	85.30

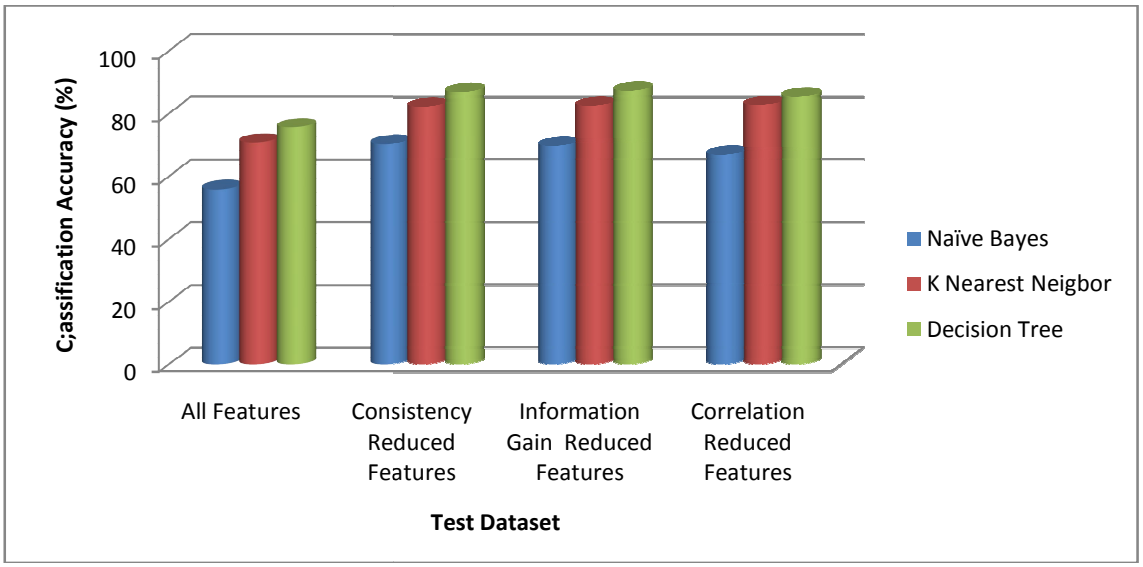

Figure 3: Bar Chart of Attacks/Normal classification accuracy of test Dataset

Table 7, shows the multi-class classification accuracy of the base models evaluation of the test dataset. Generic attack connection type recorded the highest classification accuracy for all the base models except for DT model on correlation reduced set. Normal connections recorded highest classification of 97.93% with DT model on correlation reduced set. Analysis, Fuzzers and worms attacks categories has the least classification accuracy. NB models on the consistency reduced set recorded the highest classification accuracy of 97.26% for Generic attack connection followed by correlation reduced set with accuracy of 95.26% for Generic attack connection. Information gain models recorded the classification accuracy of 92.63% on Generic attack connection.. KNN model with the information gain reduced set recorded the highest classification accuracy of 98.35% on generic attack connection, closely follow by the consistency reduced set with 98.30% accuracy on generic attack connection type, correlation reduced set recorded classification accuracy of 97.91% on generic attack connection type, DT with information gain and consistency reduced sets recorded classification accuracy of 98.60% and 98.49% on generic attack connection respectively, while correlation reduced set recorded highest classification accuracy of 98.20% on normal connection type.

Table 7: Multi-class classification accuracy of Base models with the three (3) Reduced Features Testing Dataset

Network Connection Categories	NB on Information Gain	NB on Consistency	NB on Correlation	KNN on Information Gain	KNN on Consistency	KNN on Correlation	Decision Tree on Information Gain Reduced Dataset	Decision Tree on Consistency Reduced Dataset	Decision Tree on Correlation Reduced Dataset
Analysis	60.15%	62.55%	67.35%	**0.00%**	**0.00%**	**0.00%**	**22.85%**	**22.70%**	**0.00%**
Backdoor	58.25%	60.48%	61.00%	46.96%	48.63%	46.30%	65.41%	64.43%	45.25%
Dos	55.16%	58.70%	66.46%	72.67%	72.81%	74.51%	82.11%	82.05%	79.47%
Exploits	50.86%	50.14%	50.92%	64.74%	62.94%	65.82%	71.90%	71.50%	69.48%
Fuzzers	50.76%	66.14%	**37.02%**	72.79%	71.74%	71.63%	79.42%	78.54%	78.71%
Generic	**92.63%**	**97.26%**	**95.26%**	**98.35%**	**98.30%**	**97.91%**	**98.60%**	**98.49%**	97.93%
Normal	76.48%	69.33%	67.27%	95.40%	96.08%	95.36%	97.38%	97.03%	**98.20%**
Reconnaissance	56.11%	57.21%	55.71%	63.08%	58.95%	63.64%	75.19%	73.62%	61.85%
Shellcode	88.44%	90.29%	89.41%	34.77%	58.16%	54.37%	69.20%	70.26%	69.20%
Worms	**40.00%**	**42.31%**	72.31%	26.15%	44.52%	37.69%	67.69%	64.52%	56.15%

Stacked Ensemble Predictions

The predictions of the three base models; Naïve Bayes, K nearest neighbor and decision tree were used to train each of the three stacking meta algorithms; Multi Response Linear Regression, Meta Decision Tree and Multiple Model Trees, to build the stacked ensemble models. Table 8 shows the overall classification performances of the three (3) metal models on each of the three (3) feature selected reduced datasets. Metal Decision Tree (MDT) on the consistency reduced dataset recorded the highest classification accuracy of 99.01%, while Multi Response Linear Regression (MLR) recorded the least classification accuracy of 96.58%,table 9 shows the multi class classification of stacked ensemble models with each of the reduced dataset, figure 4 shows the graphical presentation of table 8, figures 5, 6, and 7 shows graphical presentation of MLR stacking, MMT stacking and MDT stacking of the base models prediction respectively as presented in table 9

Table 8:Binary Classification Accuracy of the Metal Classifies on the Reduced Models Predictions

	Consistency Reduced Models	Information Gain Reduced Models	Correlation Reduced Models
Multi Response Linear Regression (MLR)	96.58%	97.80%	97.8%
Multiple Model Tree (MMT)	98.91%	98.16%	96.89%
Metal Decision Tree (MDT)	99.10%	98.45%	98.08%

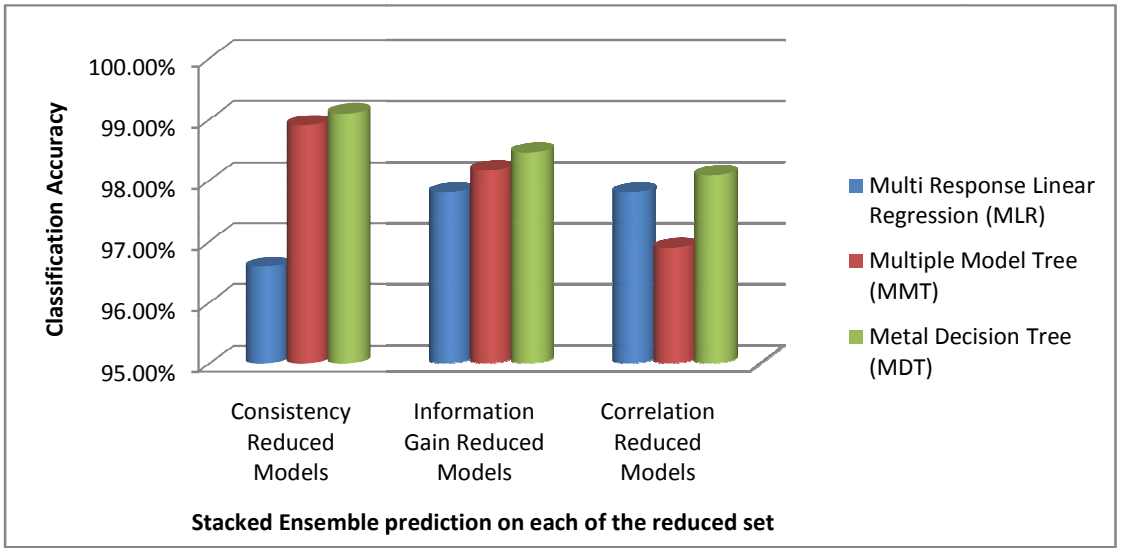

Figure 4 : Binary Classification Accuracy of the Metal models

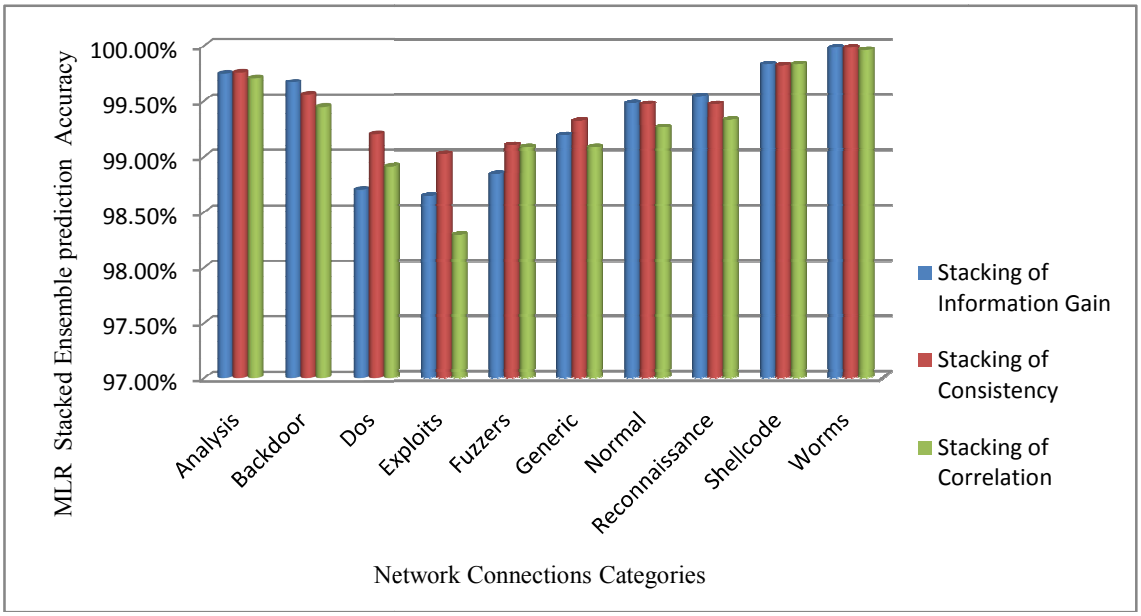

Figure 5: Prediction Accuracy of MLR Stacked Ensemble of the Reduced Base Models Predictions

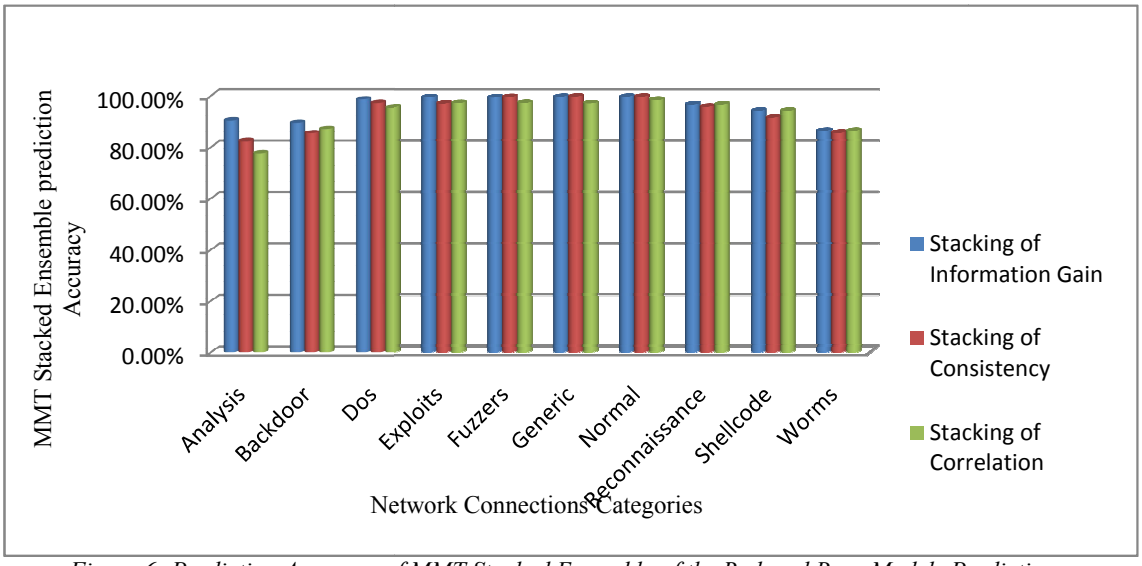

Figure 6: Prediction Accuracy of MMT Stacked Ensemble of the Reduced Base Models Predictions

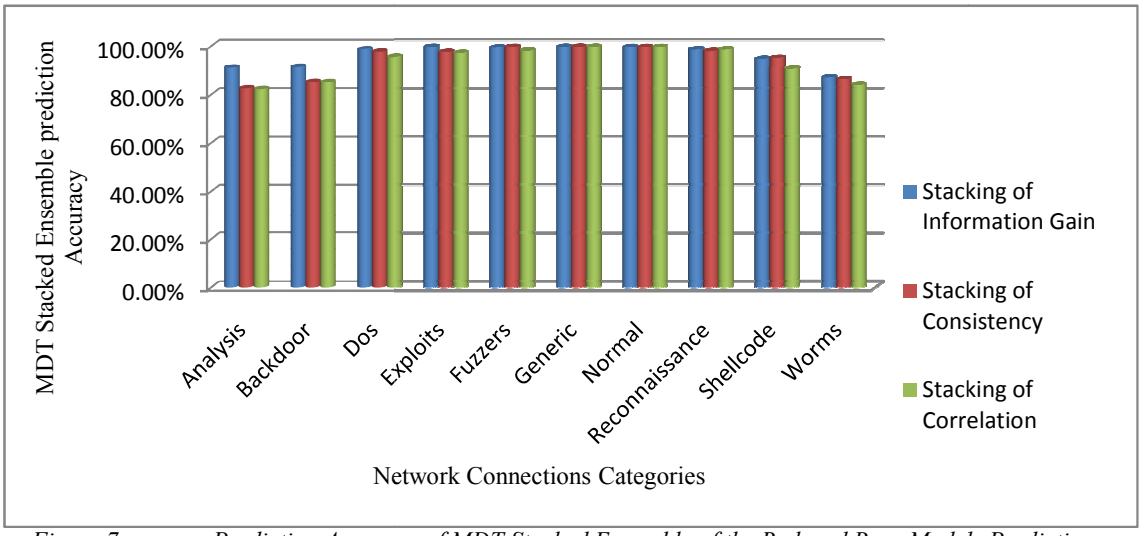

Figure 7: Prediction Accuracy of MDT Stacked Ensemble of the Reduced Base Models Predictions

Stacked Ensemble Predictions Improvement

Tables 10 and 11 shows the classification improvement and reduced false alarm rate with the stacking of the three based models, the highest classification rate of 87.18% recorded by the base model (C4.5 Decision Tree with Information gain reduced set) is lower by 9.4% than the least classification accuracy of 96.58% recorded by the stacking of the three base models (consistency reduced set) with MLR Meta -classifier, also, the lowest false alarm rate (FAR) of 1.62% by decision tree model on information gain reduced set is higher than the highest FAR of 0.27% recorded by the stacking of the three base consistency reduced set models with MLR meta-classifier. Figures 8 and 9 shows the classification accuracy improvement and false alarm rate reduction obtained by the stacking of the base models

Table 10: Classification Accuracy Improvement with the Stacking of Base Models

`	Basic Models			Staking with Base Models		
	NB	KNN	DT	MLR	MMT	MDT
Consistency	70.20	82.05	86.77	96.58	98.91	99.1
Information	69.59	82.55	87.18	97.80	98.16	98.45
Correlation	66.74	82.62	85.30	97.80	96.89	98.08

Table 11: False Alarm Rate Improvement with the Stacking of Base Models

`	Basic Models			Stacking with Base Models		
	NB	KNN	DT	MLR	MMT	MDT
Consistency	4.20	2.53	1.67	0.27	0.13	0.11
Information	4.37	2.46	1.62	0.25	0.22	0.19
Correlation	4.93	2.43	1.89	0.41	0.37	0.23

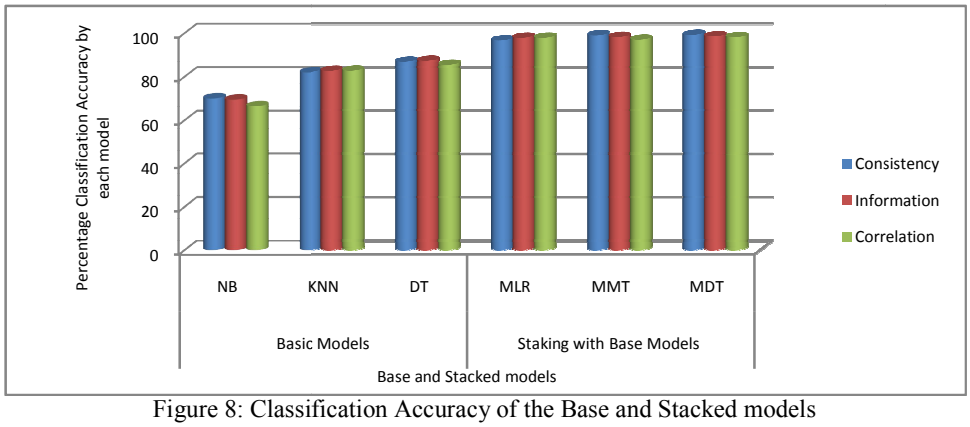

Figure 8: Classification Accuracy of the Base and Stacked models

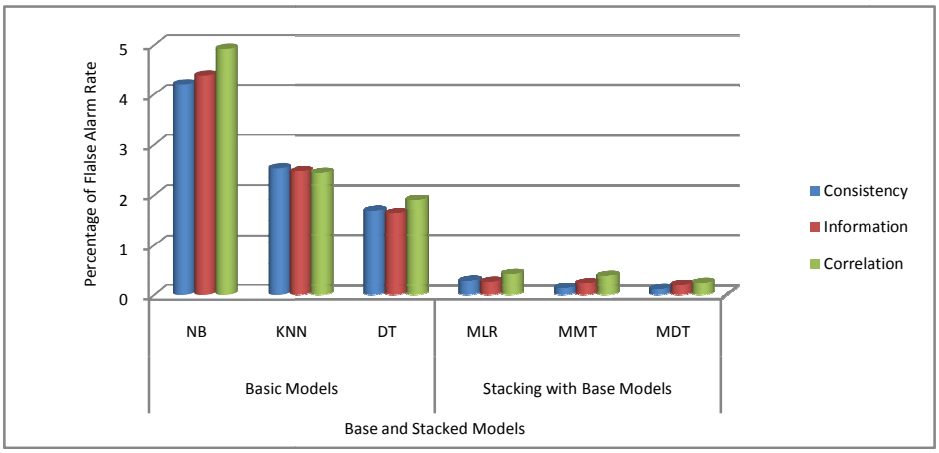

Figure 9: False Alarm Rate of the Base and Stacked models

Conclusion and Recommendation

Building a good model is one of the major challengers of an intrusion detection system, models with very high classification accuracy and low false alarm rate are desirable, but are difficult to obtain, training several algorithms and learning from their predictions to build a stacked ensemble system is capable of increasing prediction accuracy and reduces false alarm rate. This work focused on the development of stacked ensemble intrusion detection system using the UNSW-NB15 dataset, we investigated the possibility of using Stacked Ensemble meta algorithms to improve the classification accuracy and False alarm rate performance on network intrusion detection systems, using KNN, Naive Bayes and C4.5 Decision tree as base models. From this research, it was discovered that stacking with MDT outperformed stacking with MMT and MLR across the three reduced test dataset used.

The results from the study shows that Naive Bayes model with correlation reduced dataset has the highest classification accuracy of 67.35% and 72.31% for Analysis and Worms attack categories respectively, Decision Tree model with information gain reduced dataset recorded highest classification accuracy of 65.41%, 82.11%, 71.90%, 79.42%, 98.60% and 75.19% for Backdoor, Dos, Exploits, Fuzzers, Generic and Reconnaissance attacks categories respectively. Highest Normal connection classification accuracy of 98.2% was recorded by Decision Tree model with correlation reduced dataset, Shell code attacks category highest classification accuracy of 90.29% was recorded by Naive Bayes model with consistency reduced dataset. Metal Decision Tree (MDT) on the consistency reduced dataset has the highest classification accuracy of 99.01%, while Multi Response Linear Regression (MLR) has the least classification accuracy of 96.58%. From the results of analysis , it can be further concluded that Stacked ensemble learning technique increases classification accuracy and reduces false alarm rate of the base classification models, however, the execution time of all our ensemble models are to high, which makes it not adequate to be used for real time intrusion detection implementation. The performance evaluation shows our model outperformed the result from [14] in term of classification accuracy improvement and the false alarm rate reduction at both base level and meta level.

Despite the performance improvement recorded by the stacked ensemble models, their execution time is high, we therefore recommend the implementation of this system on Apache Spark big data processing tool environment so as to increase speed execution for both models building and evaluation in a real time mode and minimize any bottles neck that the models may introduce to the network.

Ethical Standard

Funding: This research work is a self funded research undertaken by the corresponding author in the Department of Computer Science, School of Computing, Federal University of Technology, Akure, Nigeria

Conflict of Interest: The corresponding author states that there is no conflict of interest

References

[1] Pontarelli, S., Bianchi, G., Teofili, S (2013).: "Traffic-Aware Design of a High-Speed FPGA Network Intrusion Detection System". IEEE Transactions on Computers 62(11),

[2] Paul van Kessel (2018) "Cyber Security Regained: Preparing to Face Cyber Attacks" 20th Global Information Security Survey 2017-18. white paper, https://www.ey.com/Publication/vwLUAssets/ey -cybersecurity-regained-preparing-to-face- cyber-attacks.pdf (Access Date: 15th January, 2020).

[3] Olasehinde O. O., Alese B.K. and Adetunmbi A.. O. (2018). "A Machine Learning Approach for Information System Security". IJCSIS. 16(12).

[4] Adebayo O. A. (2009) "A Bagging Approach to Network Intrusion Detection" Journal of the Nigerian Association of Mathematical Physics, V. 15.

[5] Lee, W., Stolfo, S.J. and Mok, K.W. (1999) "A Data Mining Framework for Building Intrusion Detection Models." Proceedings of the 1999 IEEE Symposium on Security and Privacy.

[6] Aydin M., Ali M., A.,Halim Zaim, and Gokhan Ceylan. (2009) "A hybrid Intrusion Detection System Design for computer network security", Computers and Electrical Engineering.

[7] Chebrolu, S., Abraham, A., and Thomas, J. P.(2005). feature deduction and ensemble design of intrusion detection system. computer and Security, 24(4).

[8] Hansen L K, Salamon P. .(1990): "Neural Network Ensembles" . IEEE Transaction on Pattern Analysis and Machine Intelligence, 12(10).

[9] Schapire R.E. .(1990): "The Strength of Weak learnability". Mach Learn ;5(2).

[10] Dimitriadou E., Weingessel A., Hornik K., (2003) "A Cluster Ensembles Framework, Design and Application of Hybrid Intelligent Systems", Handbook, IOS Press, Amsterdam, Netherlands. https://dl.acm.org/doi/10.5555/998038.998100 (Access Date: 5th October, 2019).

[11] Corona I., Giacinto G., Mazzariello C., Roli F., and Sansone C..(2009) "Information Fusion for Computer Security: State of the Art and Open Issues". Information Fusion, 10(4).

[12] Choudhury S. & Bhowal A., (2015) "Comparative Analysis of Machine Learning Algorithms along with Classifiers for Network Intrusion Detection", International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), Chennai, doi: 10.1109/ICSTM.2015.7225395.

[13] Tama B. A.and Rhee K. H., A (2015) "Combination of PSO-based Feature Selection and Tree-based Classifiers Ensemble for Intrusion Detection Systems" , in Advances in Computer Science and Ubiquitous Computing, *CSA/CUTE*..

[14] Syarif I., Zaluska E, Prugel-Bennett A., Wills G. (2012) "Application of Bagging, Boosting and Stacking to Intrusion Detection." In: Machine Learning and Data Mining in Pattern Recognition. Springer.

[15] Olasehinde O. O., Alese B. K. and Adetunmbi A., O. (2019) "Stacked Ensemble of Intrusion Detection Systems with Multiple Model Tree Meta Algorithm," 2020 International Conference in Mathematics, Computer Engineering and Computer Science (ICMCECS©2020 IEEE 10.1109/ICMCECS47690.2020.240893.

[16I ICKDDM (1998) International Conference on Knowledge Discovery and Data Mining-98, " Proceeding of the Fourth International Conference on Knowledge Discovery and Data Mining . https://dl.acm.org/doi/proceedings/10.5555/3000 292.

[17] ICKDDM (1999) "Proceeding of the Fifth ACM SIGKDD "International Conference on Knowledge Discovery and Data Mining-99, Association for Computer Machinery, New York, USA. ISBN 978-1-58113-1437 https://dl.acm.org/doi/proceedings/10.1145/3121 29

[18] Moustafa N. and Slay J. (2015) " A Hybrid Feature Selection for Network Intrusion Detection Systems: Central Points" , Proceedings of the 16th Australian Information Warfare Conference, Perth, Western Australia.

