Computer Engineering and Intelligent Systems
ISSN 2222-1719 (Paper) ISSN 2222-2863 (Online)
Vol.8, No.2, 2017

An Efficient Intrusion Detection Approach Utilizing Various WEKA Classifiers

Ravi kishor Ahirwar
PG Scholar, CSE, VITS, Bhopal, India

Prof. Sumit Sharma
HOD CSE, VITS, Bhopal, India

Abstract
Detection of Intrusion is an essential expertise business segment as well as a dynamic area of study and expansion caused by its requirement. Modern day intrusion detection systems still have these limitations of time sensitivity. The main requirement is to develop a system which is able of handling large volume of network data to detect attacks more accurately and proactively. Research conducted by on the KDDCUP99 dataset resulted in a various set of attributes for each of the four major attack types. Without reducing the number of features, detecting attack patterns within the data is more difficult for rule generation, forecasting, or classification. The goal of this research is to present a new method that Compare results of appropriately categorized and inaccurately categorized as proportions and the features chosen. In this research paper we explained our approach "An Efficient Intrusion Detection Approach Utilizing Various WEKA Classifiers" which is proposed to enhance the competence of recognition of intrusion employing different WEKA classifiers on processed KDDCUP99 dataset. During the experiment we employed Adaboost, J48, JRip, NaiveBayes and Random Tree classifiers to categorize the different attacks from the processed KDDCUP99.
Keywords: Classifier, Data Mining, IDS, Network Security, Attacks, Cyber Security

1. INTRODUCTION

Intrusion detection systems help to identify malicious and dangerous attacks sent to networks and computers while allowing normal traffic to arrive at its intended destination. In order for intrusion detection systems to identify harmful traffic to computers and networks, packets of data are classified to determine if the contents contain malicious actions or not. Fields of data representing the traffic flow must be collected and analyzed to determine which traffic may pass and which traffic is blocked. The two primary methods used for intrusion detection are signature-based systems and anomaly based systems. A signature based system attempts to match specific patterns in the packets traversing the network for byte strings which are known to be malicious. Anomaly based systems analyze the statistics of the traffic to determine if the packet is malicious.

Data for intrusion detection systems may be collected from multiple sources such as system approach logs and activity logs. As these disparate sources merge into a single corpus of data with many records that may provide insight into the collected activity. Each record contains fields that provide information about the activity that the record represents. Some of the fields may contain similar, irrelevant, or missing data, which could potentially cloud the analysis and the overall quality of data. The amount of data collected may also be quite large and impractical to analyze.

For anomaly intrusion detection, fields within the data files are referred to as features. These features describe a particular aspect of information in the record. Since there may be duplicated and irrelevant features contained within the data, using only those features directed at the analysis reduces the computing resources and may improve the accuracy of the resulting analysis. The process of selecting the data, to include only required features, is termed feature selection. The goal of feature selection is to use only the fields that represent the packet activity while maintaining the integrity of the record and the integrity of entire data set.

There are various methods available to select these pertinent features based on statistics by using one or more algorithms such as used in artificial intelligence, clustering, classification, statistics, and specialized applications targeting specific problems. There are no generic solutions to detect each various type of intrusion or anomalous activity.

Intrusion detection came into picture after the significant paper from Anderson in around 1980s [1]. Since then, a number of frameworks and methods have been proposed, implemented and later utilized. Various techniques like association rules, NaiveBayes classifier, Bayesian networks, support vector machines, clustering methods like k-means and the fuzzy c-means, genetic algorithms, artificial neural networks, hidden Markov models (HMM), autonomous and probabilistic agents for intrusion detection, etc. have been exploited to design the framework for such systems [2, 3]. Fig 1 shows incredible growth of Cyber Attacks from 2009 to 2014 (a 400% enhance over the past 5 years. Technology innovations are born to prevent such attacks but same innovations are enemy other way to destroy an innovation which creates unique challenge in the files of cyber security. All the major domains including Finance, Retail Industry and also Government Agencies have become

victims for cyber-attacks.

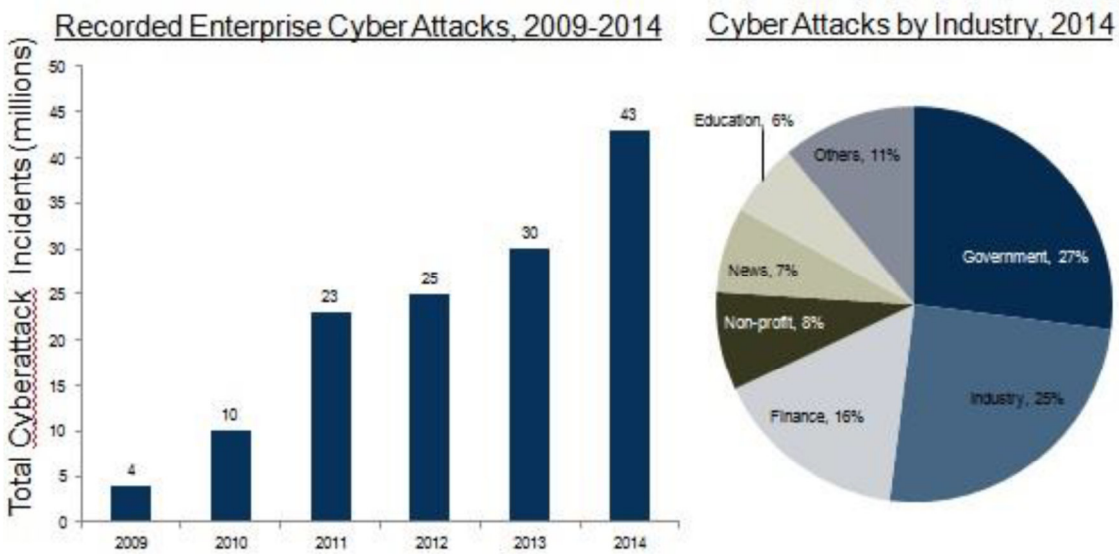

Figure 1: Cyber Attacks Statistics for 2014

All the research work has its own pros and cons. The idea is to combine the best of all and make system more competent and accurate. This ensemble approach utilizes the advantages of some of the standalone techniques and minimizes their respective shortcomings. Currently, any security tool which should detect potential threats over a network scans the incoming network packets and maps the most probable attack to each suspicious packet. With this one-to-one mapping between an event and the predicted attack, it is highly likely that the actual attack bound to happen is missed out, as follows. The main intention of attackers could be beyond just one type of attack. It could be gaining approach to the system, compromising the valuable classified information, bringing the system down or more. Such highly coordinated attacks cannot be detected with existing intrusion detection systems. Experience shows that thousands of alarms go off at the same time. It is extremely important to not only identify the potential attacks but also to quantify their probability of occurrence without having to manually sort through these alarms to identify which is a potential attack and even worse, be wrong identification.

The goal of this research is to present a new method that correctly identifies relevant features from an intrusion detection dataset that reduces the amount of data required for anomalous activity detection while maintaining the integrity of the data set. By reducing the redundant features, irrelevant features, and noise, better results may be gained in the analysis of the data for identifying anomalous activities.

The expected results of this research included the following goals:

1. Methods to identify relevant features and minimize the number of features selected from a source of network traffic data without altering the characteristics of the data representation.
2. Compare results of correctly classified and incorrectly classified as percentages, and the features selected.

2. IDS Overview

In routine life the requirement for rapidity approach of information through web has enhanced. Therefore the space for sustaining safety in any organization either opens or secret system has become fundamental. As a consequence of enhance in network connections and methods, illegitimate entrance and disruption of the data is activated. As a result, it is essential to generate an effective approach path. In common intruders have competence to discover shortcoming in systems or networks and could initiate vulnerabilities. Although the approach control points exist in network, they are ineffective in providing thorough safety to the systems.

To recognize intruders, emergent Intrusion Detection Systems (IDSs) is the most excellent resolution to defend systems and networks. Hence the effort of IDS is not only to identify intruders but as well to observe the attack of intruders. A precise system of securing information and resources from prohibited approach, injurious and denial of utilization is to be constructed. For all system, the defense perception is to be prepared based on the expected performance. Primarily safety is concerned with the following features in a computer organization.

- ***Confidentiality:*** data is to be accessed only by allowed users.
- ***Integrity:*** data must persist unchanged by damaging or malevolent efforts.
- ***Availability:*** computer is liable to function without decrease of approach and grant resources to authorized clients when they desire it.

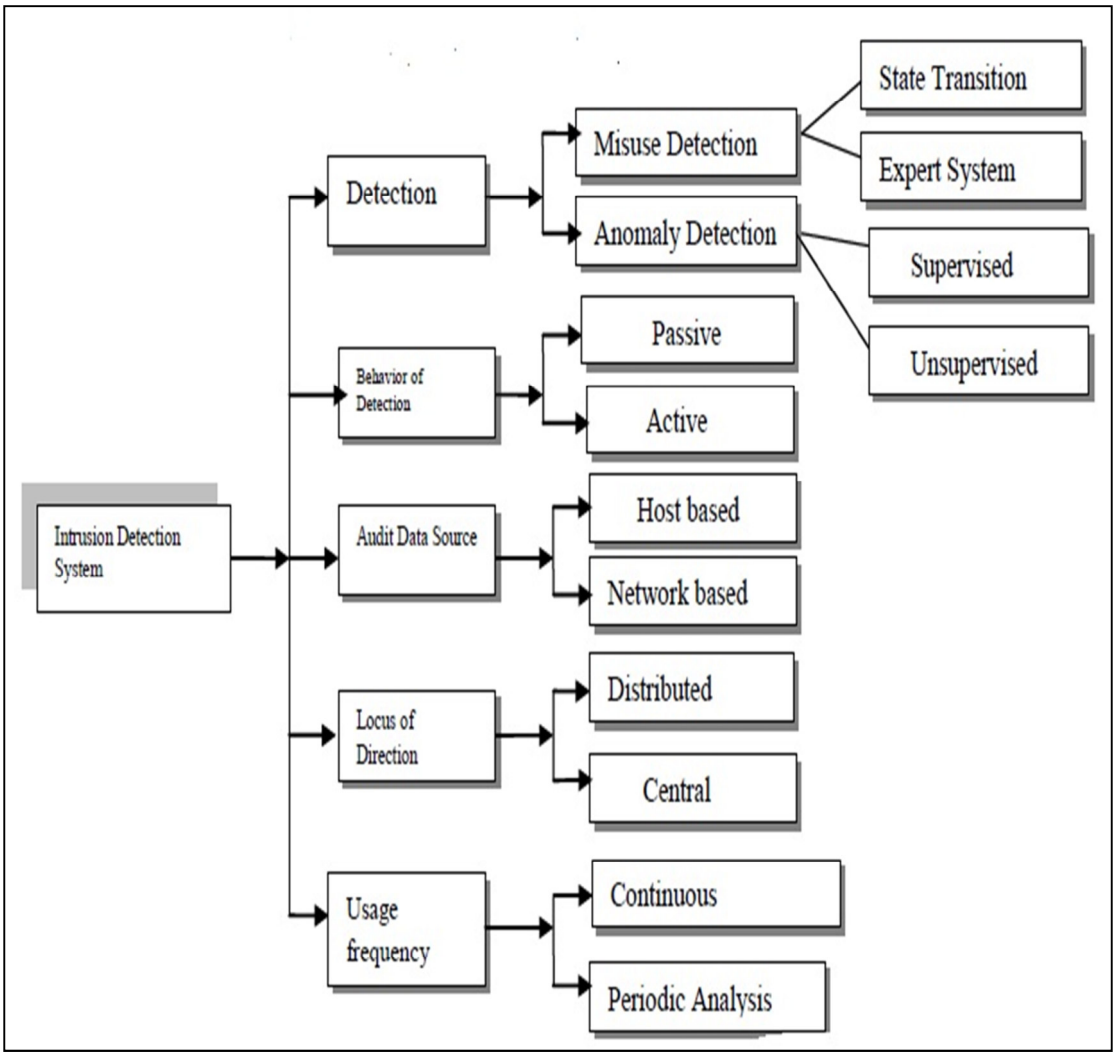

Figure 2: Classification of IDS based on its characteristics

Especially an intrusion is described as a set of occurrences which are strange and sudden to the client, which negotiate the security of a computer organization. It could be made from outside area or inside area of the organization. Formerly in 1980"s P. Anderson has described intrusion as the range of illicit strength to access data, cheat data, or making the computer organization insecure. Intrusion Detection System (IDS) was economically endorsed in the year 1990. Since then various designs were proposed to adapt intrusion detection systems [4].

It performs similar to an intruder alarm and discovers any variety of contravention and produces alarms similar to audible, visual and as well messages similar to e-mail. On the complete, IDS is principally demoralized for preventing imperfect actions that may assault or abuse the organization by discovering attacks through providing preferable maintain for security organization and also provide useful information concerning intrusion. But formation of IDS should own small false alarms while task of the detection of attacks. IDSs have become defensive methods everywhere in existing networks. There is no thorough and expert methodology proposed in verifying the potency of these organizations.

There are complex relationships existing among features as well as intrusion classes. It will produce more processing costs and also delays in detecting intrusions. In view of the restrictions on humans and computers together, feature selection is accordingly essential such that burden in handling data and time required in noticing intrusions will be lessened.

In detecting intrusions, IDS defends a computer network from illicit persons, possibly insiders. The attack recognition effort is considered as the model of classification expert in distinguishing "harmful" connections referred as intrusions or attacks, and "sympathetic" connections referred as normal. There are various categories of IDSs are prevailing that are based on structure and detection process. In addition to these, there are other characteristics one could utilized to classify IDS as shown in the fig. 2.

3. Related Work

Authors of research work [5] stated an Aho-Corasick algorithm based on parallel string matching for recognition of intrusion. The balance Space utilization among homogenous Finite State Machine (FSM) for every string matcher and a finest set of bit location clusters are established and the objective patterns are sorted by Binary Reflected Grey Code (BRGC) which diminishes the bit transmissions and are consumed for recognition of intrusions.

Work of [6] has examined the feature selection of network traffic and the impacts on the detection rates. The KDDCUP 99 dataset is utilized as experimental dataset. The detection rates are found by choosing the various combinations of these feature groups. The ineffectiveness of the approach is also shown in finding anomalies by looking at the host based features within the shorter time interval of 2 secs.

In research work [7] authors have acknowledged a novel process for HNIDS via taking two stage strategies with weight balancing model. In the online stage, the network packets are detained and divide according to the nature of protocol, then intrusion are discovered by every sensor. In the offline, training dataset is utilized to construct model, which could identify intrusion. It calculates the SMOTE over sampling process, AdaBoost and random forests algorithm.

Authors of [8] have researched with Conditional Random Fields and Layered Approach to tackle two concerns namely precision and Recall. The proposed system based on Layered Conditional Random fields outperforms other well recognized process for instance the decision trees and the NaiveBayes. The improvement in attack detection is very high, particularly, for the U2R attacks (34.8% improvement) and the R2L attacks (34.5% improvement).

Authors of [9] have focused on the exercise of weight of network protocol and modeled a weight founded anomaly detector which could effectively discover outliers of network servers. It expands these researches by pertain a novel noise decreased Fuzzy Support Vector Machine to enhance the recognition rate. The novel process known as PAYL-FSVM employs reform error based fuzzy membership function to decrease the noise of the data and to resolve the sharp boundary difficulty. The outcome of noisy data still receives part in reducing the precision.

Authors in [10] have developed a C4.5 Decision Tree algorithm and converted it into rules. The rules are utilized to detect the intrusions from the normal data. The network behavior is analyzed and classified as normal or misuse. The complete processing of the network data is found to be an overhead in this case.

Xiaodan Wang et al [11] have proposed Decision Tree based Support Vector Machine. The feature space of the Support Vector Machines is divided based on the decision tree structure. The structure of the tree is closely related to the performance. A new reparability measure is described based on the distribution of the training samples in the feature space. This measure is utilized in the formation of the Decision Tree. The performance is improved than the individual usage of Decision Tree or Support Vector Machines.

Fariba Haddadi et al [12] have represented the two layer feed forward NN for detection of intrusions. Early stopping strategy is utilized in training to overcome the matter of over-fitting. DARPA dataset is utilized for the experiments. The pre-processed data is converted in the range [-1, 1] and given to the NN for classification of Intrusions.

Demidova and Ternovoy [13] have demonstrated the use of Neural Networks for detecting network attacks. The Back-Prorogation Neural Network is utilized to find the attacks in the network traffic. The detection rate is enhanced whereas the false alarm rate is also very high.

AI Islam and Sabarina [14] have devoted research efforts to model the detection system utilizing Recurrent Neural Networks (RNN) which detects the flooding attacks such as DoS and DDoS attacks. Several index terms like Denial-of-service, Distributed-Denial-of-Service, IP spoofing, Flood attack, Zombie, RNN Ensemble are described and they are utilized in detection rate of attacks but the detection of new attacks is found to be very low.

Intelligent intrusion detection Hierarchical Neuro-Fuzzy Classifier is utilized Principal Component Analysis (PCA) to reduce the features and Fuzzy-C Means Clustering is utilized to create the Fuzzy rules. kddcup99 data is utilized for evaluation of the experiments. Genetic Algorithm is utilized in optimizing the results of the detection model.

In research work [15] authors have designed an intrusion recognition model utilizing Evolutionary Neural Networks. The enhancement is shown with respect to less time for recognition since the organization of the network and load of the network are revealed simultaneously. Experimental studies with the dataset 99 Defense Advanced Research Projects Agency (DARPA), recognition of intrusion Evaluation data authenticate that Evolutionary Neural Networks ENNs are successful for recognition of intrusion with small trade off with the training time.

Kok-Chin Khor et al [16] have the employed the use of single and multiple Bayesian classifier approach utilizing variations of Bayes Network such as NaiveBayes Classifier, Bayesian Networks, and Expert-elicited Bayesian Network, since only Bayes classifiers are included in the combination technique and this approach

offers less detection results with standard available data like kddcup99

Panda et al [17] have developed a discriminative multinomial NaiveBayes Classifier for NIDS with filtering analysis. The variation of the kddcup99 dataset namely NSL-kddcup99 is utilized. Two class classification which gives high classification rate and better accuracy with low false alarms is performed.

4. Proposed Work

In this chapter we will explain our approach "An Efficient Intrusion Detection Approach Utilizing Various WEKA Classifiers" which is proposed to enhance the competence of recognition of intrusion employing different WEKA classifiers on processed KDD cup 99 dataset. During the experiment we employed Adaboost, J48, JRip, NaiveBayes and Random Tree classifiers to categorize the different attacks from the processed KDD cup 99. The WEKA Classifiers are calculating Precision, recall, f-measures and ROC Curve Area performance during the experiment. A WEKA 3.8.1 workbench is employed for the experimental study purpose [18].

Figure 3: WEKA 3.8.1 Interfaces

4.1 Experimental Setup

Testing is carried out on the system having i3 Processors, 4 GB RAM, UBUNTU 14.10 Linux Operating System and WEKA 3.8.1 Learning Workbench developed by university of Waikato is frequently utilized for machine learning algorithms and the classification purpose.

WEKA (Waikato Environment for Knowledge Analysis) is a gathering of a variety of algorithms of Machine Learning which is coded in Java and they could be employed for solving troubles of data mining. Excluding these Machine Learning algorithms of WEKA (Fig.3) also furnishes alternatives for association rules, clustering, classification pre-processing, regression and visualization of the dataset. It could be broadened by the client to implement innovative algorithms [19].

4.2 Flow Graph

Figure 4 show the Flow graph of our proposed approach An Efficient IDS Detection Approach Utilizing Various WEKA Classifiers.

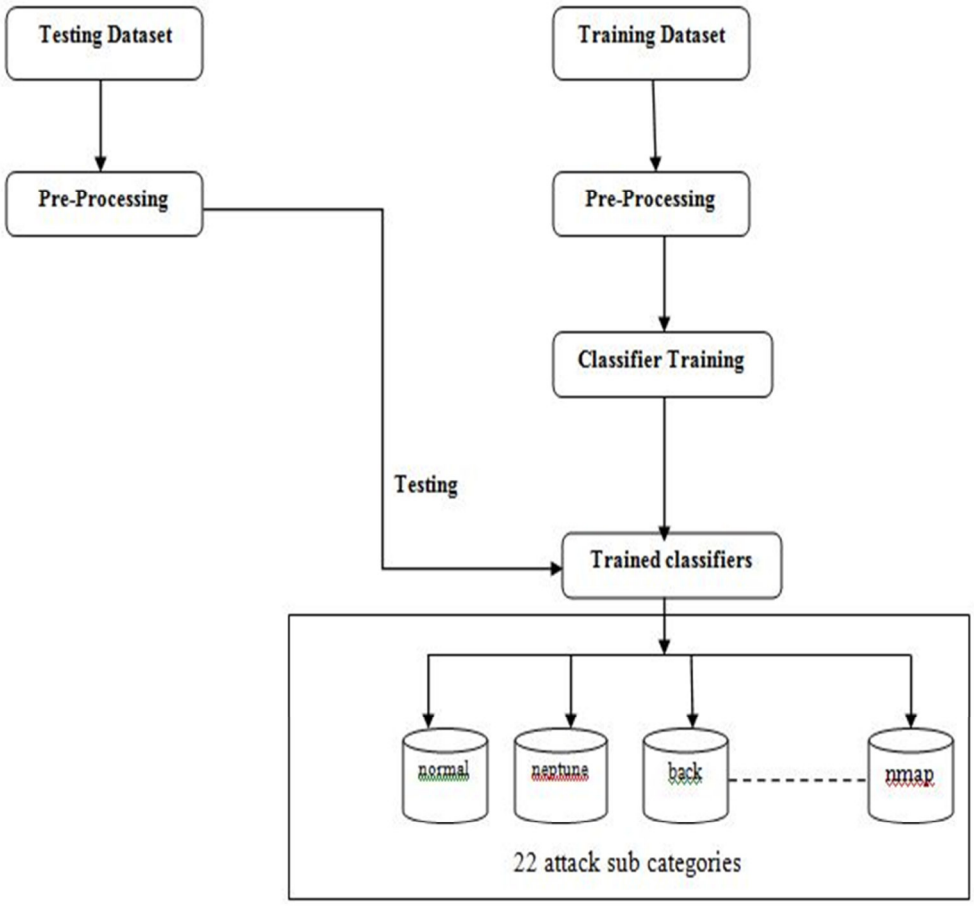

Figure 4: Flow graph of proposed approach

4.3 Procedure of Proposed Algorithm

Dataset Training: - The KDDCup99 dataset in ARFF file Format is employed for the purpose of experimentation study. The KDDCup99 dataset training is an assortment of 494,020 records. All dataset tuple is a solo attached vector expressed through 41 feature values and precisely one tag of either 'normal' or an 'attack' is given. The size of KDDCup99 is 51MB of which 70% is used for training.

Dataset Testing: - KDDCup99 dataset testing is discovered for the experimental study of proposed system. The dataset testing is separated into the individual attack. By defaults KDDCup99 dataset is arranged of five attack categories that are DOS, R2L, U2R, Probe and Normal. The size of KDDCup99 Test dataset is 45 MB of which 30% is used for testing.

Table 1: Attacks Present In the Kddcup'99 Datasets

Attack Name	Attacks in KDDCup99 Training set	Additional attacks in KDDCup99 Test set
DoS	back, neptune, smurf, teardrop, land, pod.	apache2, mailbomb, processtable.
Probe	satan, portsweep, ipsweep, nmap.	mscan, saint.
R2L	warezmaster, arezclient, ftpwrite, guesspassword, imap, multihop, phf, spy	sendmail, named, snmpgetattack, nmpguess, xlock, snoop, worm.
U2R	rootkit, bufferoverflow, loadmodule,perl.	httptunnel, ps, sqlattack

Pre-Processing: The dataset training and testing is separated into the individual attack label. By defaults KDDCup99 dataset is arrangement of 5 attack categories that are DOS, R2L, U2R, Probe and Normal however in our proposed work KDDCup99 dataset is processed as mentioned 5 attack categories. The attacks in KDDCup99 training dataset and attacks in KDDCup99 testing dataset are shown in the Table 1. The Number of samples in the kddcup99 dataset and distribution of attacks is shown in Table 2

Table 2: Number of Samples in the Kddcup99 Test Set and Distribution of Attacks

Attack Category	Number of Samples	Distribution of Attacks in %
Normal	60589	19.48
DoS	229853	73.90
R2L	16179	5.20
U2R	228	0.07
Probe	4165	1.4
Total	**311014**	**100**

Classification: Processed KDDCup99 dataset is tested with the various WEKA classifiers like Adaboost, J48, JRip, NaiveBayes, and Random Tree. Short rationalization of each employed classifier is precise below:

1. **ADABOOST:** Boosting is a family of methods for improving the performance of a "weak" classifier by using it within an ensemble structure, the most prominent member of which is AdaBoost. In Boosting methods, a set of weights is maintained across the objects in the data set, so that objects that have been difficult to classify acquire more weight, forcing subsequent classifiers to focus on them. These methods works by repeatedly running a learning algorithm on various distributions over the training data, and then combining the classifiers produced by the learner into the single composite classifier.
2. **J48: -** A decision tree is an analytical machine learning process that chooses the objective cost of a novel illustration founded on different attribute costs of the obtainable data. The interior leafs of a decision tree indicate the dissimilar attributes; the limbs among the nodes inform us the probable costs that these attributes could have in the experimental illustrations, whereas the terminal leafs notify us the concluding cost (categorization) of the relevant variable. The feature that is to be calculated is also recognized as the relying variable, as its cost relies upon, or is decided by, the costs of all the further features. The further features, which facilitate in expecting the cost of the relying variable, are identified as the autonomous variables in the dataset.
3. **JRIP** (Extended Repeated Incremental Pruning): JRip implements a propositional rule learner, "Repeated Incremental Pruning to Produce Error Reduction" (RIPPER), as proposed before. JRip is a rule learner alike in principle to the commercial rule learner RIPPER.
4. **NAIVEBAYES: -** A NaiveBayes classifier is a straightforward probabilistic classifier founded on pertaining Baye's theorem with strong (naive) freedom hypothesis. In easy words, a NaiveBayes classifier supposes that the occurrence (or absence) of an individual feature of a class is unconnected to the occurrence (or absence) of any other feature, specified the class variable.
5. **RANDOMTREE:** Random Tree is a supervised Classifier; it is an ensemble learning algorithm that produces many entity learners. It occupies a bagging scheme to create an arbitrary set of information for creating a decision tree. In ordinary tree every node is dividing utilizing the best divide amongst all variables. In a random forest, every node is dividing utilizing the best amongst the subset of predicators randomly selected at that node. Random trees have been initiated by Leo Breiman and Adele Cutler. The algorithm could agreement with both categorization and deterioration troubles. Random trees are a gathering (ensemble) of tree predictors that is known as forest. The categorization efforts as follows: the random trees classifier obtains the input feature vector, classifies it with each tree in the forest, and outputs the class tag that established the bulk of "votes". In case of deterioration, the classifier reaction is the average of the reactions over all the trees in the forest

5. Result Analysis

5.1 Evaluation Parameters

True Positive (TP) / Recall : True Positive in this perspective is described as the amount of true positives separated through the entire amount of parts that actually apply to the positive category (i.e. the addition of true positives and false negatives, which are articles which weren't tagged as applying to the positive category but should have been).

It is the measure of positive events that were properly categorized as positive, as computed utilizing by the following equation:

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

False Positive (FP): It is the quantity of negative cases that were wrongly categorized as positive, as calculated utilizing by the following equation:

$$FP = \frac{FP}{TN + FP}$$

True Negative (TN): It described as the quantity of negatives cases that were categorized appropriately, as computed utilizing by the following equation:

$$TN = \frac{TN}{TN + FP}$$

False Negative (FN): It is the quantity of positive cases that were mistakenly classified as negative, as computed utilizing by the following equation:

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

Precision: correctness for a class is the amount of true positives (i.e. the quantity of items properly marked as residing to the positive class) divided by the total quantity of components marked as residing to the positive class (i.e. the totaling of true positives and false positives, which are items wrongly marked as residing to the class). Precision (i.e., accuracy) is the measure of the total sum of attacks that are correctly determined. It is accomplished utilizing by the following equation:

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

F-Measure: - A enumerates that joins precision and recall is the harmonic mean of precision and recall, the usual F-measure or balanced F-score. F Measure that joins precision and recall is the harmonic mean of precision and recall is known as F-measure.

$$F - \text{measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

This is also known as the F1 measure, because recall and precision are equally loaded.

ROC: - Receiver operating characteristics (ROC) plans are supportive for systematizing classifiers and visualizing their result. Receiver Operating Characteristic (ROC), or ROC curve, is a graph plot that exhibits the result of a binary classifier technique as its intolerance threshold is various. The curve is created by plotting the true positive rate against the false positive rate at a range of threshold settings. Receiver Operator Characteristics (ROC) exhibits the tradeoff between sensitivity and specificity. ROC curves plot the true positive rate vs. the false positive rate, at varying threshold cutoffs. The ROC is also known as relative operating feature curve, since it is an estimate of two operating characteristics (TPR and FPR) as the criterion modifies.

5.2 Experimental Results and Discussion:

The Experiment Results study of the NaiveBayes, J48, JRip, Random Tree, and Adaboost classifiers is given away in Table 3, 4, 5, 6 & 7. As it could be seen the performance of NaiveBayes Classifier is lower average. For U2R and R2L attack is it's less than 41% score. The cause for this is by reason of the hypothesis of NaiveBayes approach that all parameters are self-governing. Nevertheless this is not forever the case. Many protection parameters are mutually dependent to one another. As an outcome NaiveBayes Classifier, even it takes a lesser amount of memory and is faster in calculation is avoided on account of poor results.

Table 3: Results of NaiveBayes Classifier

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.793	0.01	0.995	0.793	0.883	0.987	**dos**
	0.712	0.003	0.131	0.712	0.221	0.997	**u2r**
	0.983	0.138	0.087	0.983	0.159	0.994	**probe**
	0.966	0.075	0.411	0.966	0.576	0.976	**r2l**
	0.68	0.005	0.971	0.68	0.8	0.977	**normal**
Weighted Avg.	**0.782**	**0.014**	**0.948**	**0.782**	**0.841**	**0.985**	

To get better upon NaiveBayes Classifier we have utilized J48, Random Tree and Adaboost classifier in WEKA. These three classifiers have given away noteworthy enhancements in detection rate and accuracy. As it could be observed in table 4, 6 and 7 that average TP rate for J48, Random Tree & AdaBoost classifier is above 98% which is quite higher as compared to NaiveBayes and Jrip classifiers whose weighted average is 78.1% and 97.7%. Almost all the attacks have precision of exceeding 81% in J48, Random Tree and Adaboost classifier.

Table 4: Results of J48 Classifier

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	1	0.001	1	1	1	1	**Dos**
	0.773	0	0.879	0.773	0.823	0.947	**u2r**
	0.981	0	0.986	0.981	0.984	0.996	**Probe**
	0.83	0.011	0.807	0.83	0.819	0.994	**r2l**
	0.947	0.011	0.954	0.947	0.95	0.997	**Normal**
Weighted Avg.	**0.98**	**0.003**	**0.981**	**0.98**	**0.98**	**0.999**	

Table 5: Results of Jrip Classifier

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	1	0.011	0.996	1	0.998	0.994	**Dos**
	0.864	0	0.934	0.864	0.898	0.954	**u2r**
	0.982	0	0.985	0.982	0.984	0.996	**Probe**
	0.75	0.01	0.808	0.75	0.778	0.971	**r2l**
	0.952	0.013	0.946	0.952	0.949	0.998	**Normal**
Weighted Avg.	**0.977**	**0.011**	**0.977**	**0.977**	**0.977**	**0.994**	

Table 6: Results of Random Tree Classifier

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	1	0.001	1	1	1	1	**Dos**
	0.848	0	0.918	0.848	0.882	0.932	**u2r**
	0.991	0	0.981	0.991	0.986	0.995	**Probe**
	0.83	0.011	0.804	0.83	0.817	0.992	**r2l**
	0.945	0.011	0.954	0.945	0.95	0.997	**Normal**
Weighted Avg.	**0.98**	**0.003**	**0.98**	**0.98**	**0.98**	**0.999**	

Table 7: Results of Adaboost Classifier

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	1	0	1	1	1	1	**Dos**
	0.818	0	0.947	0.818	0.878	0.986	**u2r**
	0.994	0	0.99	0.994	0.992	1	**Probe**
	0.831	0.011	0.81	0.831	0.821	0.995	**r2l**
	0.948	0.011	0.955	0.948	0.951	0.999	**Normal**
Weighted Avg.	**0.981**	**0.003**	**0.981**	**0.981**	**0.981**	**0.999**	

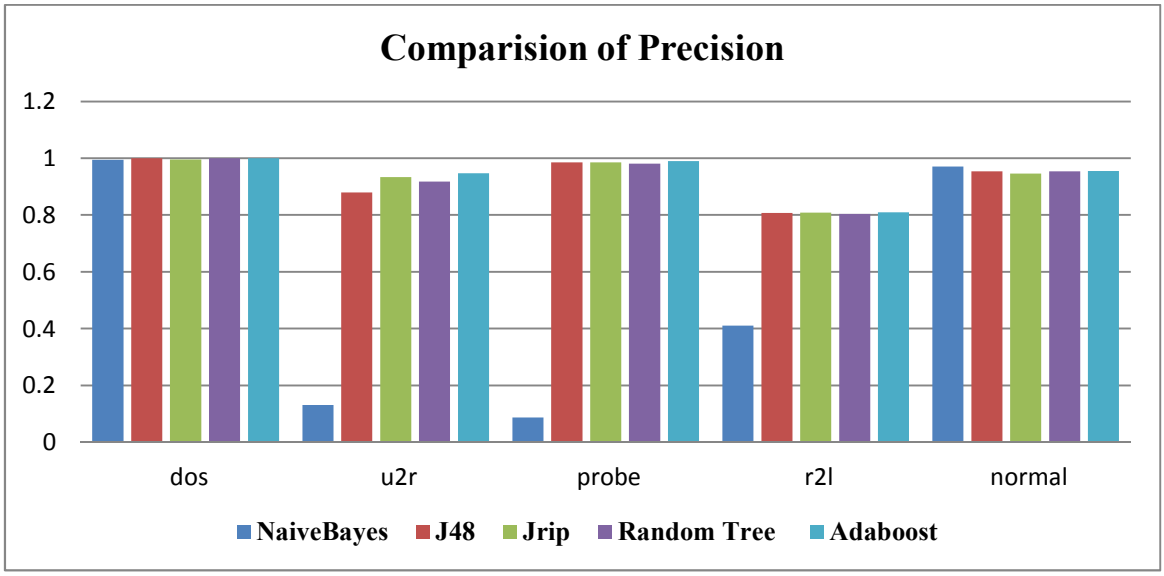

Figure 5: Comparison of Precision of Classifiers Utilized

6. Conclusion

Motivated by the shortcomings of prior approaches to the Intrusion Detection Problem, such as high false positive detection rates and poor detection performance on rarer but dangerous classes of network attacks, a new machine learning framework is introduced that leverage novel approach to intrusion detection. The proposed method "An Efficient Intrusion Detection Approach Utilizing Various WEKA Classifiers" is suitable for processing large multiclass network intrusion detection datasets such as the KDD Cup 99. The performance of this algorithm is compared to with the standard Adaboost, J48, JRip, NaiveBayes and Random Tree classifiers for the purpose of classification. Classifiers are evaluated based on Precision, recall, f-measures and ROC Curve Area performance criteria's. A WEKA 3.8.1 tool is utilized for the purpose of experimental study. It is observed that Adaboost is the best classifier among all utilized classifiers during the experiment. The conclusions of the all classifiers are appraised with other distinctive machine learning techniques. The implementation results of suggested algorithm demonstrate that the suggested machine learning technique offers maximum classification precision up to 99.76 %

In future this work could be extended in order to include more classifiers and could furthermore execute feature selection to improve classification accuracy & effectiveness.

Following idea could be improving our present work in future:

1) In order to test the accuracy of this model in real time, a network could be used which is able to introduce natural real time intrusions with numerous packets and diverse network scenarios.
2) Since there is no limit to the number of neural networks, we could implement this model using various neural networks which requires less pre-processing unlike the CC4 neural network, which requires input data in unary format and conversion to unary format requires lot of steps.
3) Regardless of the disputes to the problem of Detection of Intrusion, the introduction of new efficient and scalable techniques that combine a number of diverse classification decisions is still a relevant and vibrant area of research. Such research will ultimately help to strengthen the efficiency of detection and prevention of intrusion upcoming attacks.

References

1. http://www.jpost.com/Blogs/Unleavened-Media/Key-Tech-Trends-for-Q2-and-Beyond-393964
2. AlEroud, Ahmed. Contextual information fusion for the detection of cyberattack. UNIVERSITY OF MARYLAND, BALTIMORE COUNTY, Ph.D. Dissertation, 2014.
3. Sharma, Prayank. "A framework for Intrusion Detection Systems based on contextual semantics." PhD diss., UNIVERSITY OF MARYLAND, BALTIMORE COUNTY, 2012.
4. Liao, Hung-Jen, Chun-Hung Richard Lin, Ying-Chih Lin, and Kuang-Yuan Tung. "Intrusion detection system: A comprehensive review." Journal of Network and Computer Applications 36, no. 1 (2013): 16-24.
5. Hyunjin Kim, Hyejeong Hong, Hong-Sik kim and Sungho Kang. "A Memory-Efficient Parallel String Matching for Intrusion Detection Systems", IEEE communication letters, pp. 1004-1006, 2009
6. Ma, Wanli, Dat Tran, and Dharmendra Sharma. "A study on the feature selection of network traffic for intrusion detection purpose." In Intelligence and Security Informatics, 2008. ISI 2008. IEEE International Conference on, pp. 245-247. IEEE, 2008.
7. Yueai, Zhao, and Chen Junjie. "Application of Unbalanced Data Approach to Network Intrusion Detection." In Database Technology and Applications, 2009 First International Workshop on, pp. 140-143. IEEE, 2009.
8. Gupta, K.K., Nath, B. and Kotagiri, R. "Layered Approach Using Conditional Random Fields for Intrusion Detection", IEEE Trans.Dependable and Secure Computing, Vol. 7, No. 1, pp. 35 - 49, 2010.
9. Guiling Zhang, Yong Zhen Ke, Liankun Sun and Wei Xin Liu. "An Improvement of Payload-based Intrusion Detection Using Fuzzy Support Vector Machine", in Proc. of the International Conference on Information Security, pp. 1-4, 2010
10. Juan Wang, Qiren Yang and Dasen Ren. "An Intrusion Detection Algorithm Based on Decision Tree Technology", in Proc. of the International Conference on Information Processing, pp. 333-335, 2009.
11. Xiaodan Wang, Zhaohui Shi, Chongming Wu and Wei Wang. "An Improved Algorithm for Decision-Tree-Based SVM", in Proc. of the International Conference on Information Security, pp. 4234-4238, 2006.
12. Haddadi, F., Khanchi, S., Shetabi, M. and Derhami, V. "Intrusion Detection and Attack Classification Using Feed-Forward Neural Network", in Proc. of the International Conference on Computer and Network Technology, pp. 262-266, 2010
13. Demidova, Y. and Ternovoy, M. "Neural Network Approach of Attacks Detection in the Network Traffic", in Proc. of the International Conference on CAD Systems in Microelectronics, pp. 128-129, 2007

14. AI Islam, A.B.M.A. and Sabrina, T. “Detection of various Denial of Service and Distributed Denial of Service Attacks using RNN Ensemble”, in Proc. of the twelfth International Conference on Computers and Information Technology, pp. 603-608, 2009
15. Sang-Jun Han and Sung-Bae Cho. “Evolutionary Neural Networks for Anomaly Detection Based on the Behaviour of a Program”, IEEE Trans.on Systems, Man, and Cybernetics, Part B: Cybernetics, Vol. 36, No. 3,pp. 559-570, 2005.
16. Kok-Chin Khor, Choo-Yee Ting and Amnuaisuk S.P. “Comparing Single and Multiple Bayesian Classifiers Approaches for Network Intrusion Detection”, in Proc. of the Computer Engineering and Applications, pp. 325-329, 2010
17. Panda, M., Abraham, A. and Patra, M.R. “Discriminative Multinomial Naïve Bayes for Network Intrusion Detection”, in Proc. of the International Conference on Information Assurance and Security, pp. 5-10, 2010.
18. Hall, Mark, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. "The WEKA data mining software: an update." ACM SIGKDD explorations newsletter 11, no. 1 (2009): 10-18.
19. Holmes, Geoffrey, Andrew Donkin, and Ian H. Witten. "Weka: A machine learning workbench." In Intelligent Information Systems, 1994. Proceedings of the 1994 Second Australian and New Zealand Conference on, pp. 357-361. IEEE, 1994.