International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 10 (2018) pp. 7249-7255
© Research India Publications. http://www.ripublication.com

Feature Classification and Outlier Detection to Increased Accuracy in Intrusion Detection System

Nachiket Sainis
M.Tech Research Scholar
Department of Computer science & Engg.
BRCM CET, Bahal, Haryana India.

Durgesh Srivastava
Research scholar
IKGPTU, Jalandhar

Dr. Rajeshwar Singh
Professor
DOABA Group of Colleges Rahon,
SBS Nagar, Punjab India.

Abstract

The day by day targeted network attacks is steadily increasing and evolving, forcing businesses to revamp their network security systems due to possible data and capital losses. Intrusion Detection Systems is a very important element for almost any security system. The key feature of IDS is the active detection of unauthorized access that tries to compromise the confidentiality, availability and Integrity of computer or computer networks. Many researchers have already developed security and advanced techniques to explore technologies to detect cyber attacks with all DARPA 1998 dataset for Intrusion Detection and improved versions of this KDD Cup'99, NSL-KDD Cup and GureKDDcup data set.

In this research, we evaluate the use of five ML classification algorithm to deal with the attack classification problem. They are SVM, Naive Bayes, KNN and the Decision Tree based C4.5 (J48) and Random Forest Algorithm. The project objective is to compare if some of the newer dataset and the most advanced, such as NSL-KDD and GurKDDcup dataset, could be much better candidates than older and overused DARPA KDD Cup 1999. The exact number of available features in a dataset clearly shows the relevance of detecting specific types of attacks. Another objective of this research, is to provide a suitable way to reduce the intricacies of the developed classification models in the first phase by reducing the features domain.

Keywords : Intrusion Detection system, Machine Learning Algorithm, KDD Cup'99, GureKDDcup, NSL-KDD Cup, Feature Selection, Outlier Detection.

INTRODUCTION

Network security is a primary problem nowadays since the internet usage is increasing in multi-dimensions mainly because of a lot more usage of portable gadgets. Intrusion Detection Systems will be helpful to identify malign intentions of network users without compromising the security of the computer system and the network.

Intrusions are referred to as attempt to compromise the integrity, availability and confidentiality of a computer system or network [1]. IDS are hardware or software systems that start process automatically for analyzing activities in a computer network or network evaluating them for warning signs of security problems [2].

Feature selection is the approach of eliminating the features of the actual dataset that are unnecessary with reference to the task being performed. Therefore, not only is the execution time of the classifier processing the data decreases, but also the accuracy improved, mainly because there are unnecessary or redundant features consist of noisy data that badly affect the classification accuracy [3].

In this paper, we propose a different feature selection technique that uses the dataset that contains no outliers that means outlier free dataset available in real time. The classification algorithms C4.5, Random Forest, SVM, Naive Bayes and K-NN will be evaluated using the dataset KDD cup'99, NSL-KDD and GureKDDcup to identify four types of attacks: Dos, Probe, R2L and U2R. The reduction of features is implemented using common feature selection techniques correlation-based Feature Selection (CFS). The final results of the machine learning classifier are computed to compare feature reduction techniques to show that our suggested model works efficiently for network intrusion detection.

DETECTION TECHNIQUES FOR IDS

1. Anomaly based Intrusion detection

The anomaly-based intrusion detection specified as outliers, peculiarities or exceptions are the data pattern which functions abnormally. The anomaly detection technique is used to discover patterns that are far from normal, while others are marked as intrusion. Anomaly detections is categories as dynamic and static detectors. The static anomaly detector is considered part of the monitored system, which remains constant. The static part contains two parts, the system code and the system data. The static parts of the system can be represented as one bit. In case of deviation from its original form, the error was reported or the burglar rebuilt the part of the system.

The dynamic detector contains the definition of the system behavior. The system behavior is defined as a series of different pairs [4]. If unsafe actions is considered abnormal, system administrators can be warned by false positives [6].

2. Signature based Intrusion detection

Signature-based attack detection is known as misuse detection. Here the dataset has several instances and all data must be marked as normal or abnormal. To train the KDD data, Machine learning classification algorithms are used as per their label. This method is used to keep record that automatically maintains the signature pattern to detect the intruder in network computer. The misuse detection technique is created automatically and the jobs are more complicated and accurate than manual. Based on the robustness and severity of an signature activated in the system, a notification or alarm response must be sent to the concern authorities [5].

Relying on rules, a Signature based Detection System will try correlate likely patterns to intrusion attempts. To access a system, viruses try numerous steps in a particular pattern.

RELATED WORK

Feature Selection

Feature selection is very important to increase the effectiveness of machine learning algorithms. This process use for selection of a subset from original features based on specific criteria, and this is a main and often used data mining approach to reduce dimensionality. Most data contains noisy, redundant and irrelevant features. Feature selection approach minimizes the number of features, by eliminating redundant, noisy and irrelevant features, and has significant effects on applications: accelerating a machine learning classification algorithm, improving learning accuracy and performance improvement [7].

Filtering methods are generally used as a pre-processing step. The feature selection is independent of a machine learning classification algorithm. Selection of features according to their scores in different statistical tests on their correlation with the result variables [8]. These approach is very efficient in computational time and robust to over-fitting.

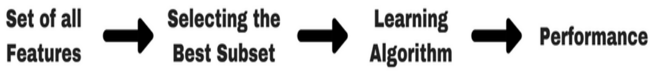

Figure 1. Filter Method for Feature Selection Process

There are two common ways of reducing features : A wrapper uses the expected learning algorithm to evaluate the usefulness of features, while a filter evaluates features according to the general data properties. The wrapper method is generally viewed as a better subset of features, but runs significantly slower than a filter [9].

DATASET DESCRIPTION

1. KDDCup'99

KDD Cup’99 intrusion detection datasets that are based totally on DARPA ’98 dataset [10] provides labelled dataset for researcher running within the area of intrusion detection and represent the publicly available labelled dataset. The detailed description of KDD dataset is given in the next phase. The KDD'99 dataset is created the usage of a simulation of an army network. In the end, there is a sniffer which records all transmitted network traffic data by using the Tcpdump format. KDD training [11] dataset contains around 4,900,000 single connection vectors, every one of which includes 41 attributes and is categories as either an attack or normal, with precisely one specified attack type. The simulated attacks classified amongst the subsequent four classes: Denial of Service (Dos), Remote to Local (r2l), Probe and User to Root (u2r) attacks [12]. Features are labelled into four listed types:

- **Basic Features:** These characteristics tend to be derived from packet headers while no longer analyzing the payload.
- **Content Features:** To analyze the actual TCP packet payload, Domain knowledge is used and this encompasses features which includes the large variety of unsuccessful login attempts.
- **Time-based Traffic Features:** These features are created to acquire properties accruing over a 2 second temporal window. An example of such a feature will be the wide range of connections to the exact same host over the interval of 2 second.
- **Host-based Traffic Features:** Make use of a historical window calculated over the numerous connections and it is 100 in this case. Thus Host based attributes are created to analyze attacks, which time frame longer than 2 seconds [13].

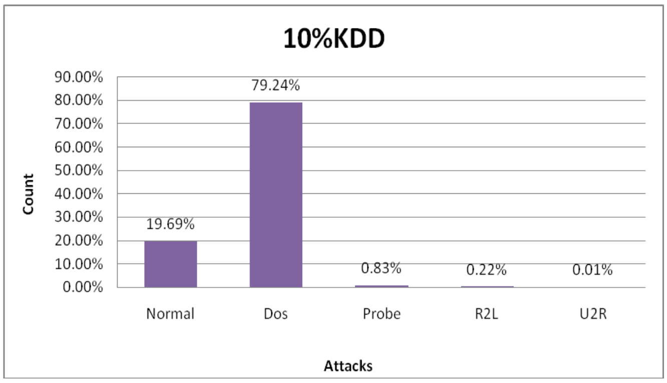

Figure 2. Statistics for Normal and Attack type in 10%KDD CUP'99

2. NSL KDD

The NSL KDD dataset is offline network data based totally on KDD'99 dataset [14]. The NSL-KDD data set trained to solve many different immanent issues of the KDDCUP'99 data set. KDD CUP’99 is said to be broadly used data set for anomaly detection [15] for locating accuracy in intrusion detection [16]. The deficiency found in the KDD CUP’99 [10] data set is the extensive quantity of duplicate record of approximately 78% in train set and 75% in test set, respectively. That makes the learning algorithm rule biased, that makes U2R much more vulnerable to the network. To resolve these types of problems, the latest version of KDD dataset NSL-KDD is offered.

Advantages of NSL-KDD dataset over the original KDD cup dataset:

- No duplicate data found within the NSL-KDD train data set, then the classifiers do not produce the result biased.
- The proposed test sets does not contain any duplicate records, due to this the learners’ performance will not be prevented and gives better detection rates.
- The small number of records selected by each level of difficulty is inversely proportional to the proportion of records in the KDD dataset.
- The dataset contains a reasonable number of samples by train and test sets that makes it convenient to run experiments on whole data sets without any requirement to randomly consider a small part [17].

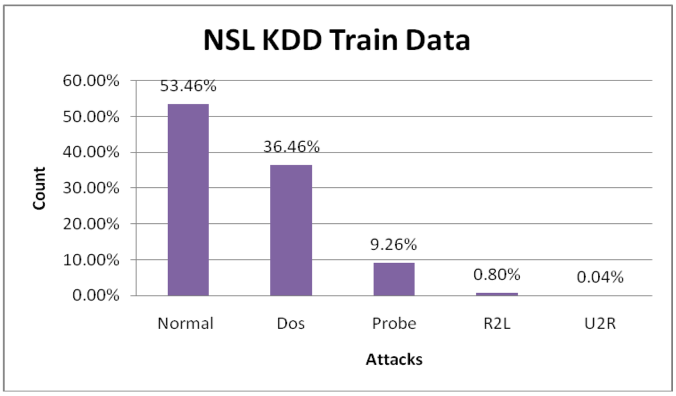

Figure 3. Statistics for Normal and Attack type in NSL-KDD Cup'99

3. *GureKDD Cup*

GureKDDcup dataset is consist of kddcup99 connections (UCI repository database) also payload added to (network packets contents) each and every connections. The GureKDDCup capture group employs the similar methods implemented to create kddcup99 [18]. They processed tcpdump data files with bro-ids and also obtained every connection with its proper features. And finally, the dataset is labeled each and every connection according to the connections-class files (tcpdump.list) which provided by MIT. The Original dataset size is too large i.e 9.3 GB and the size of 6 percent dataset is 4.2 GB respectively.

GureKddcup (and gureKddcup6percent) contains 41 features same as the KDDcup'99. The gureKddcup is too big to be utilized in any learning process. Most of the research projects with *kddcup* database are performed by using the 10% of the database available in UCI [18]. A reduced sample: gureKddcup6percent which contain only no-flood attacks matched with tcpdump.list along with a arbitrary subsample of normal connections paired with tcpdump.list. Figure 4 shows abnormal and normal class.

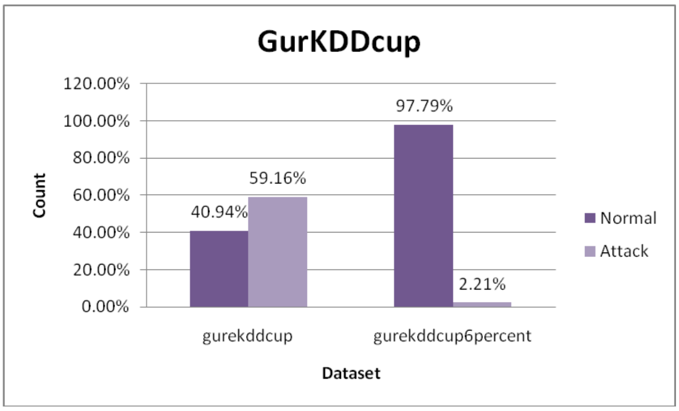

Figure 4. Statistics for Normal and Attack type in GureKDDcup

EXPERIMENTAL SETUP

This experiment has been designed and implemented to evaluate the KDD dataset with various machine learning algorithm, based on generic metrics such as accuracy, completeness, and other performance metrics; but also based on the level of details the datasets provide, notably with respect to attack classes, results are provided for every ML algorithms.

For this experiment, a 10% subset of the KDD Cup 1999, a 6% subset of the GureKDDCup dataset and full NSL-KDD dataset is loaded into *Weka*, and all of its features are selected. Each dataset will be run against the Random Forest, C4.5 (J48), K-nearest Neighbour (KNN), Support vector Machines (SVM) and Naive Bayes algorithms in Weka, with all features selected at all time. In order to choose the best quality dataset candidate for the later experiments when classifying certain attributes.

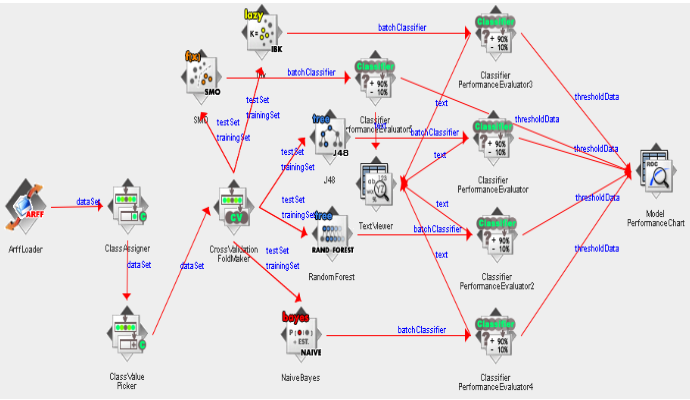
ArffLoader
dataSet
ClassAssigner
dataSet
ClassValue Picker
dataSet
CrossValidation FoldMaker
testSet
trainingSet
SMO
IBk
J48
RandomForest
NaiveBayes
batchClassifier
Classifier PerformanceEvaluator3
Classifier PerformanceEvaluator5
Classifier PerformanceEvaluator
Classifier PerformanceEvaluator2
Classifier PerformanceEvaluator4
text
TextViewer
thresholdData
Model PerformanceChart

Figure 5. Experimental Setup in Knowledge Flow

Interquartile Range

The Interquartile range (IQR) is used to define the propagation of a distribution. It's just the range where the middle half of the data points are located. In simple terms it is always the distance between the two quartiles IQR = Q3-Q1. The quartiles of a classified list of data values are three points that separate the data into exactly four equal parts, with each and every part containing quarterly data [19].

- **Q1** is referred to as the middle number between the smallest number and the median of the dataset.
- **Q2** is the median of the dataset.
- **Q3** is the middle value between the median and the highest value of the dataset.

Outlier Calculation Using Interquartile Range

- Put the data in the correct order.
- Calculate the first quartile (Q1)
- Calculate the third quartile (Q3)
- Calculate the inter quartile range (IQR) = Q3-Q1
- Calculate the lower limit = Q1-(1.5*IQR)
- Calculate upper limit = Q3+(1.5*IQR)

Everything outside the upper and lower boundary is an outlier.

Correlation Based Feature Selection (CFS)

CFS is a simple filtering algorithm that belongs to subsets of features and has discovered the merits of a feature or subset of features of a correlation based heuristic evaluation feature. The reason behind CFS is to discover subsets consisting of features that strongly correlate with class and do not correlate with each other. The remaining of the features must be ignored. Irrelevant features must be omitted simply because they strongly correlate with one or more of the remaining features. The acceptance of selected features depends entirely on how well classes in the instance range are predicted that were not previously predicted by other features. CFS‟ s feature evaluation function is shown as below [20].

$$Merit_s = \frac{k\overline{r_{cf}}}{\sqrt{k + (k-1)\overline{r_{ff}}}}$$

Main steps for feature selection process can be presented as below

- Start weka and select the tab explorer from the Application menu.
- Click on open file and load the outlier removed dataset. Click on select attributes tab
- Choose attribute evaluator as cfsSubsetEval and search method as Best First search (BFS) then click on start.
- Most relevant features based on Best First Search feature selection algorithms were selected.
- The data with the selected features was used as input to the three types of classifiers, C4.5, RF, SVM, NB and KNN. Each classifier was tested and trained in separate experiment.
- The results of the classifiers were illustrated and analyzed using evaluation criterion.

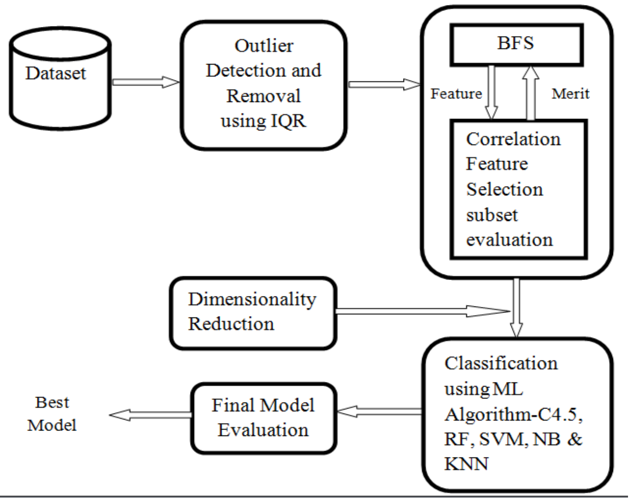
Dataset
Outlier Detection and Removal using IQR
BFS
Feature
Merit
Correlation Feature Selection subset evaluation
Dimensionality Reduction
Classification using ML Algorithm-C4.5, RF, SVM, NB & KNN
Final Model Evaluation
Best Model

Figure 6. Block Diagram for Proposed Methodology

The selected features by Outlier Removed dataset +CFS+BFS algorithms for KDD cup'99, NSL-KDD cup and GureKDDcup dataset are shown in Table I, Table II and Table III, most relevant features selected from total 42 features.

Table I. CFS + BFS Selected Features for KDD cup'99

Feature No.	Description	Type
14	lroot_shell	Continuous
8	wrong_fragment	Continuous
30	diff_srv_rate	Continuous
5	src_bytes	Continuous
7	land	Discrete
4	flag	Discrete
2	protocol_type	Discrete
23	count	Continuous
36	dst_host_same_src_port_rate	Continuous
6	dst_bytes	Continuous
3	service	Discrete

Table II. CFS + BFS Selected Features for NSL-KDD Cup

Feature No.	Description	Type
4	flag	Discrete
5	src_bytes	Continuous
6	dst_bytes	Continuous
26	srv_serror_rate	Continuous
12	logged_in	Discrete
30	diff_srv_rate	Continuous

Table III. CFS + BFS Selected Features for GureKDDcup

Feature No.	Description	Type
7	land	Discrete
8	wrong_fragment	Continuous
10	hot	Continuous
3	service	Discrete
11	num_failed_logins	Continuous

RESULT ANALYSIS

Each dataset is evaluated with machine learning classification algorithm C4.5, K-Nearest Neighbour, Naive Bayes, SVM and Random Forestwere tested before and after feature selection.

Table IV. Different dataset Accuracy & AUC with classification algorithm

Dataset	Classification Algorithm	Proposed Meth. Accuracy	ROC Curve (AUC)%
KDD Cup'99	C4.5	99.94%	1.0
NSL-KDD cup	C4.5	99.38%	0.95
GureKDD cup	C4.5	99.06%	0.75
KDD Cup'99	K-NN	99.90%	1.0
NSL-KDD cup	K-NN	99.43%	0.95
GureKDD cup	K-NN	99.08%	1.0
KDD Cup'99	Naive Bayes	96.16%	0.87
NSL-KDD cup	Naive Bayes	81.88%	0.79
GureKDD cup	Naive Bayes	99.03%	0.86
KDD Cup'99	Random Forest	99.94%	1.0
NSL-KDD cup	Random Forest	99.38%	0.95
GureKDD cup	Random Forest	99.08%	1.0
KDD Cup'99	SVM	99.94%	0.92
NSL-KDD cup	SVM	88.09%	0.83
GureKDD cup	SVM	99.06%	0.93

The Receiver Operating Characteristic curve (ROC) and its Area under the Curve (AUC) are used for the performance analysis of the classifier. It ranges from 0 to 1 and an attribute that is perfectly correlated to the class provides a value of 1. Here, 1 is the best possible value. The KNN and Random Forests algorithm are shown preeminent results almost among all the datasets i.e. 100% (accuracy).

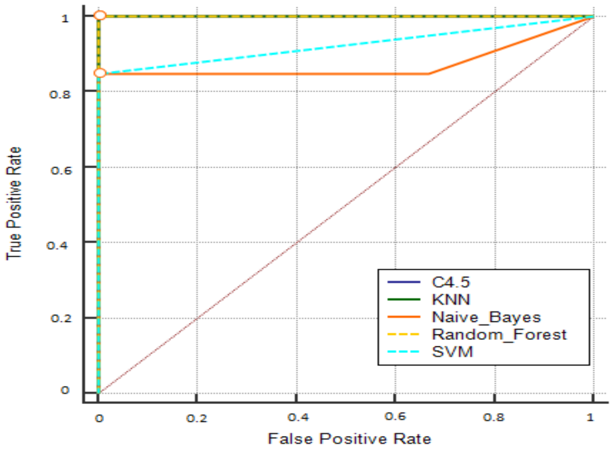

Figure 7. ROC Curve for KDD Cup'99

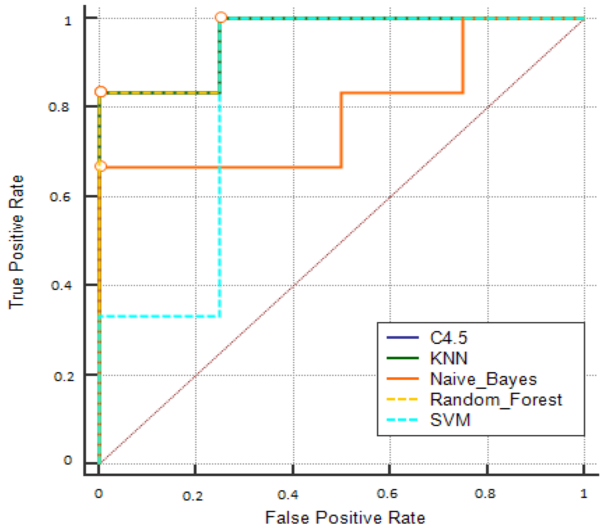

Figure 8. ROC Curve for NSL-KDD Cup

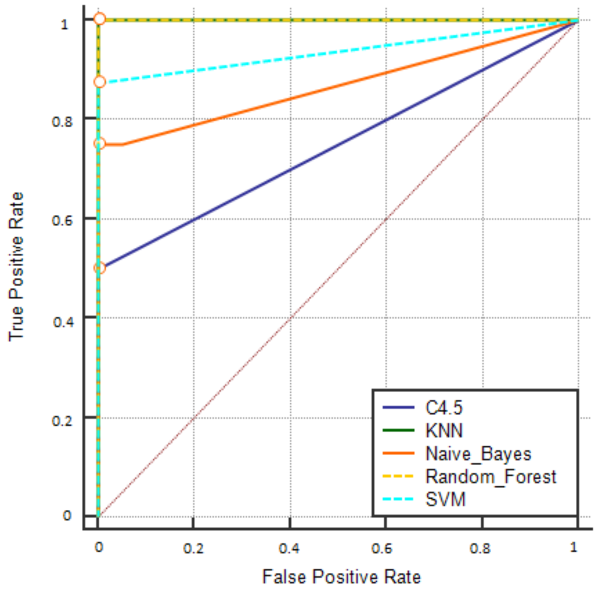

Figure 9. ROC Curve for GureKDD Cup

The obtained results are displayed in Table IV, and along with AUC (Area under curve) for all five classifiers with three different datasets Figure 7, Figure 8 and Figure 9 represents the Area under curve.

Building time for three dataset was very less with 11, 6 and 5 features was selected by BFS for KDD cup'99, NSL-KDD and GureKDDcup respectively as shown in Table V. The accuracy slightly decreased but the model building time dropped down which is a great deal of time efficiency level. By applying feature selection process it is clear that the time of all models dropped down in to less than 50%, That means also the reduction of computations size; i.e less computation complexity. This reduction was a result of number of selected features.

Table V. Model building Time with different Classifiers before and after OR+FS

Dataset	Classification Algorithm	Time Taken before OR+FS	Time Taken After OR+FS
KDD Cup'99	C4.5	119.53 sec	23.15 sec
NSL-KDD cup	C4.5	46.95 sec	7.14 sec
GureKDD cup	C4.5	55.93 sec	1.28 sec
KDD Cup'99	K-NN	0.37 sec	0.23 sec
NSL-KDD cup	K-NN	0.15 sec	0.04 sec
GureKDD cup	K-NN	0.13 sec	0.03 sec
KDD Cup'99	Naive Bayes	5.63 sec	1.36 sec
NSL-KDD cup	Naive Bayes	1.76 sec	0.22 sec
GureKDD cup	Naive Bayes	1.86 sec	0.23 sec
KDD Cup'99	Random Forest	554.63 sec	205.97 sec
NSL-KDD cup	Random Forest	157.93 sec	83.73 sec
GureKDD cup	Random Forest	166.48 sec	27.6 sec
KDD Cup'99	SVM	689.07 sec	186.53 sec
NSL-KDD cup	SVM	3126.7 sec	1356.73
GureKDD cup	SVM	302.44 sec	98.91 sec

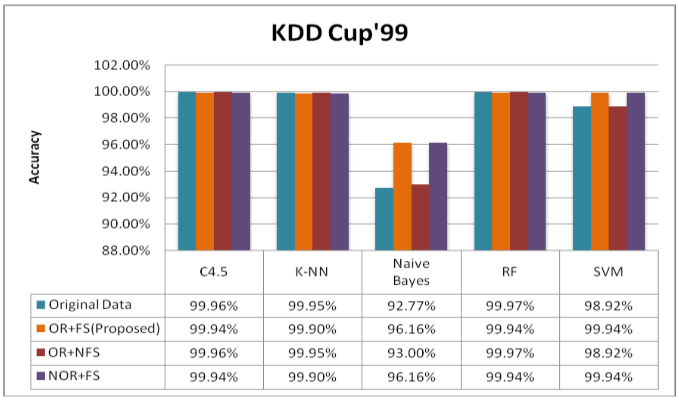

Figure 10. Overall Accuracy analysis of proposed technique in KDD cup'99

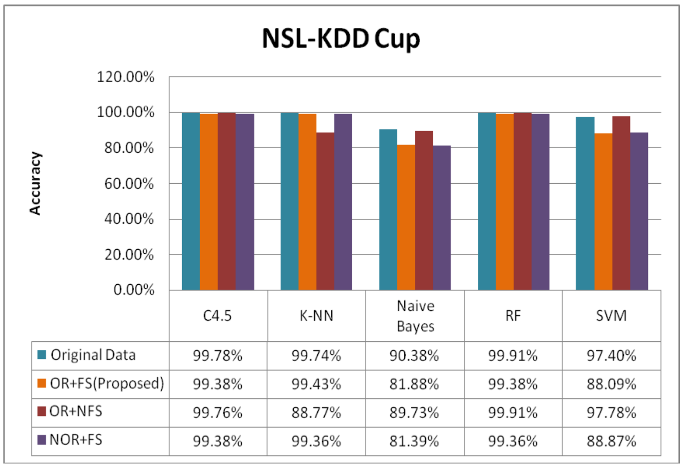

Figure 11. Overall Accuracy analysis of proposed technique in NSL-KDD cup

In accuracy analysis graph OR+FS means outliers removal and feature selection which is proposed methodology and next is OR+NFS means outliers removal with no feature selection and last is NOR+FS means no outliers removal with feature selection.

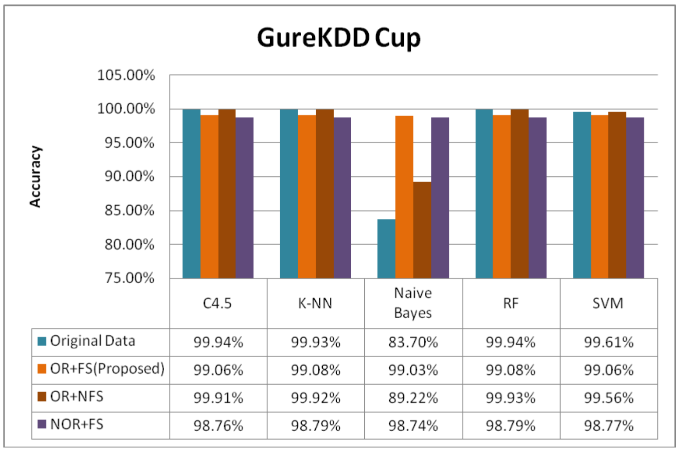

Figure 12. Overall Accuracy analysis of proposed technique in GureKDDc**up**

Finally, the complexity of the models was reduced in to a respectable amount, the process of finding the class was a function of (41 features), it reduced to a function of (11, 6 and 5 features) in the proposed method.

CONCLUSION

The usage of a network intrusion detection system in order to detect cyber-attacks and determine defense configuration, we are using Machine Learning (ML) techniques with different classification methods for the dataset. In this study, we have developed five models to solve the IDS issue using SVM, KNN, Naive Bayes and the decision tree based C4.5 (J48) and Random Forest algorithm. Number of attacks were classified using the above mentioned five methods. The importance of choosing a relevant amount of features for classification.

Although it is beneficial to know what specific cyber-attack you are exposed to, the large number of classes for classification decreases the accuracy of all the machine learning classification algorithms. To increase the possibility to render properly created decisions regarding defense configuration, the best features for classification are categories of cyber- attacks such as DoS, U2R, Probing and R2L attacks.

To enhance the efficiency of the proposed models and speeding up the detection process, first we detect outliers using Interquartile Range (IQR) and apply a filter called Remove with values to remove outliers and after that set of features were selected using the CFS subset Eval and Best First Search method (BFS). A comparison between the developed models without feature selection and with feature selection was provided. The developed models were designed for decreasing the complexity while maintaining the appropriate detection accuracy. The decision tree based Random Forest, K- Nearest Neighbour and C4.5 algorithm achieved the maximum classification accuracy compared to other search techniques explored in this work. The Random Forest and KNN algorithm work best with all three dataset with ROC of 1.000, which shows algorithm fully representative by training data.

REFERENCES

[1] Shilpa Lakhina, Sini Joseph, Bhupendra verma, Feature Reduction using Principal Component Analysis for Effective Anomaly–Based Intrusion Detection on NSL-KDD, *International Journal of Engineering Journal of Engineering Science and Technology*

[2] R.Bace and P. Mell, *NIST Special Publication on Intrusion Detection Systems*, 2001.

[3] Y Yang, JO Pedersen, A comparative study on the effect of feature selection on classification accuracy, *Procedia Technology 1,* 2012, pp.323– 327.

[4] Karthikeyan .K.R and A. Indra- “Intrusion Detection Tools and Techniques a Survey”

[5] Vera Marinova-Boncheva-“A Short Survey of Intrusion Detection Systems”-. Bulgarian academy of sciences.

[6] Application and Signature based IDS Retrived from http://www.idconline.com/technical_references/pdfs/data_communications/Application_and_Signature_Based_IDS.pdf

[7] Liu H ,Setiono R, Motoda H, Zhao Z, Feature Selection: An Ever Evolving Frontier in Data Mining, *JMLR: Workshop and Conference Proceedings 10*, 2010, pp. 4-13

[8] Website https://www.analyticsvidhya.com/blog/2016/12/introduction-to-feature-selection-methods-with-an- example-or-how-to-select-the-right-variables/

[9] Y. Kim, W. N. Street, F. Menczer, and G. J.Russell, *“Feature selection in data mining” in Data Mining*: Opportunities and Challenges: J. Wang, Ed. Hershey, PA: Idea Group Publishing, 2003, pp. 80–105.

[10] KDD Cup 1999 Intrusion Detection Dataset. [Online]. Available:http://kdd.ics.uci.edu/databases/kddcup99/kdd cup99.html

[11] Ebrahim Bagheri, Wei Lu, Mahbod Tavallaee and Ali A. Ghorbani , “A Detailed Analysis of the KDD CUP 99 Data Set”, in IEEE Symposium on Computational Intelligence for Security and Defense Applications, 2009. CISDA 2009.

[12] P. GiftyJeya, M. Ravichandran, C. S. Ravichandran “Efficient Classifier for R2L and U2R Attacks” in International Journal of Computer Applications (0975 – 8887) Volume 45– No.21, May 2012

[13] H. GünesKayacık, A. NurZincir-Heywood, Malcolm I. Heywood, “Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets” in Third Annual Conference on Privacy, Security and Trust, October 12-14, 2005

[14] P. Bhoria, K. Kanwal Garg. “Determining feature set of DOS attacks”, in International Journal of Advanced Research in Computer Science and Software Engineering, vol.3 issue 5, May 2013, pp. 875-878.

[15] Hee-su Chae, Byung-oh Jo, Sang-Hyun Choi, Twae-kyung Park, “Feature Selection for Intrusion Detection using NSL-KDD” in Recent Advances in Computer Science.

[16] S. Revathi, Dr. A. Malathi, “A Detailed Analysis on NSL-KDD Dataset Using Various Machine Learning Techniques for Intrusion Detection” in International Journal of Engineering Research & Technology (IJERT) Vol. 2 Issue 12, December 2013

[17] http://www.unb.ca/research/iscx/dataset/iscx-NSL-KDD-dataset.html

[18] https://addi.ehu.es/bitstream/handle/10810/20608

/20160601_Txostena_gurekddcup_InigoPeronaBalda.pdf?sequence=1

[19] Website https://www.geeksforgeeks.org/interquartile -range-iqr/

[20] Hall, M. (1999). *Correlation-based Feature Selection for Machine Learning*. Ph. D. thesis, University of Waikato.