An Intrusion Detection System Based on KDD-99 Data using Data Mining Techniques and Feature Selection

Pratibha Soni, Prabhakar Sharma

Abstract: ***Internet and internet users are increasing day by day. Also due to rapid development of internet technology, security is becoming big issue. Intruders are monitoring computer network continuously for attacks. A sophisticated firewall with efficient intrusion detection system (IDS) is required to prevent computer network from attacks. A comprehensive study of literatures proves that data mining techniques are more powerful technique to develop IDS as a classifier. Performance of classifier is a crucial issue in terms of its efficiency, also number of feature to be scanned by the IDS should also be optimized. In this paper two techniques C5.0 and artificial neural network (ANN) are utilized with feature selection. Feature selection techniques will discard some irrelevant features while C5.0 and ANN acts as a classifier to classify the data in either normal type or one of the five types of attack.KDD99 data set is used to train and test the models ,C5.0 model with numbers of features is producing better results with all most 100% accuracy. Performances were also verified in terms of data partition size.***

Index Terms: ***Decision tree, Feature Selection, Intrusion Detection System, Partition Size, Performance measures.***

I. INTRODUCTION

Information or network security is becoming an important issue for any organization to protect data and information in their computer network against various types of attack with the help of an efficient and robust Intrusion Detection System (IDS). IDS can be developed using various machine learning techniques. IDS act as a classifier which classifies the data as normal or attack. Classification is a process of putting different categories of data together. Classification is one of the very common applications of the data mining in which similar type of samples are grouped together in supervised manner. Su-Yu Wua et al. [1] used SVM and classification tree to compares accuracy, detection rates and false alarm rate. The result show that C4.5 is superior to SVM in accuracy and detection but in false alarm rate SVM is better. Gang Wang et al. [2] have proposed a new intrusion detection approach FC-ANN using fuzzy clustering and artificial neural network. The model gives effectiveness result for R2L and U2R attacks in terms of accuracy. V. Balon Canedo et al. [3] proposed a new KDD winner method consisting of discretizations, filters and various classifiers like Naive Bayes (NB) and C4.5 to develop robust IDS. The proposed classifier gives high accuracy i.e. 99.45% compare to others. Reda M. Elbasiony et. al [4] have suggested hybrid technique with combination of random forest with k-mean algorithm.This hybrid framework achieves detection rates and false positive rates better than other techniques. Zubair A Baig et al. [5] used supervised neural network and proposed network intrusion detection model GMDH yields high attack detection rule nearly 98%. Bin Luo et al. [6] proposed FASVFG based classifier that achieves a high generalization accuracy of 94.355 in validation experiment and average Mathews correlation coefficient reaches 0.8858. In this study a decision tree technique: C5.0 and artificial neural network (ANN) based techniques are explored in terms of partition size and feature selection. C5.0 is comparatively new decision tree technique suggested by Quinlan. The performance of this technique is better than its predecessor techniques like ID3 and C4.5 suggested by Quinlan for many applications. The techniques were used by many researchers in different problem domain for data classification and achieved very high accuracy. On the other hand ANN is good classifiers, which classify the data by presenting input-output pair. EBPN is most widely used ANN.

II. MATERIAL AND METHODS

A. Material (Dataset)

For any machine learning techniques we need historical data to be learned. Appropriate size of the data is always required to train and test the models. KDD99 Data set is an intrusion related data with almost 50 lacks samples. Ten percent of this data is publically available in UCI repository site for the experimental purpose of the researcher's. This optimum size of data contains samples for all 22 classes. A higher sample size data will require more computational resources which are not possible with simple desktop computers. So relatively low sample size data of KDD99 (10% of KDD) is used in this research work as raw material for developing a model. This data set contains about 5 million records as TCP/IP connection with 41 features, some of which are qualitative while others are continuous. Twenty two samples are categorized into five broader categories along with normal as DoS, R2L, U2R and Prob.

B. Methods

Manuscript Received on July 2014.

Pratibha Soni, M tech. (CSE), Raipur Institute of Technology, Raipur (CG), India.

Prabhakar Sharma, Asst. Prof. Department of CSE, Raipur Institute of Technology, Raipur (CG), India.

C5.0 is a decision tree based classifier developed by Ross Quinlan (rulequest.com/see5-info.html, 2010) and it is an extension of C4.5. It automatically extracts classification rules in the form of decision tree from given training data. C5.0 has many benefits over C4.5 in terms of time and memory space required, the tree generated by C5.0 is also very small as compared to C4.5 algorithm which ultimately improves the classification accuracy. ANN is another widely used technique for data classification. ANN [7] is known as best classifier and is able to mine huge amount of data for classification. They were originally developed in the field of machine learning to try to imitate the neurophysiology of the human brain through the combination of simple computational elements (neurons) in a highly interconnected system. A neural network is composed of a set of elementary computational units, called neurons, connected together through weighted connections. These units are organized in layers so that every neuron in a layer is exclusively connected to the neurons of the preceding layer and the subsequent layer. Every neuron, also called a node, represents an autonomous computational unit and receives inputs as a series of signals that dictate its activation. Following activation, every neuron produces an output signal. All the input signals reach the neuron simultaneously, so the neuron receives more than one input signal, but it produces only one output signal. Every input signal is associated with a connection weight. The weight determines the relative importance the input signal can have in producing the final impulse transmitted by the neuron. The connections can be exciting, inhibiting or null according to whether the corresponding weights are respectively positive, negative or null. The weights are adaptive coefficients that, by analogy with the biological model, are modified in response to the various signals that travel on the network according to a suitable learning algorithm.

III. MODEL DEVELOPMENT PROCESS

A process flow diagram for classification of intrusion data is depicted in Fig1, this figure can be viewed as four different parts: First is collection of 10% KDD99 data from UCI repository site and labeling them according to 5 different types of attack as data preprocessing. A total of 494,021 samples are then randomly divided into five different partitions in second phase as below:

Partition 1: 50% training and 50 % testing
Partition 2: 60% training and 40% testing
Partition 3: 70% training and 30% testing
Partition 4: 80% training and 20 % testing
Partition 5: 90% training and 10% testing

A random sampling of training and testing partition may produce different results in different runs. Best result out of 10 runs is considered for analysis of the model.

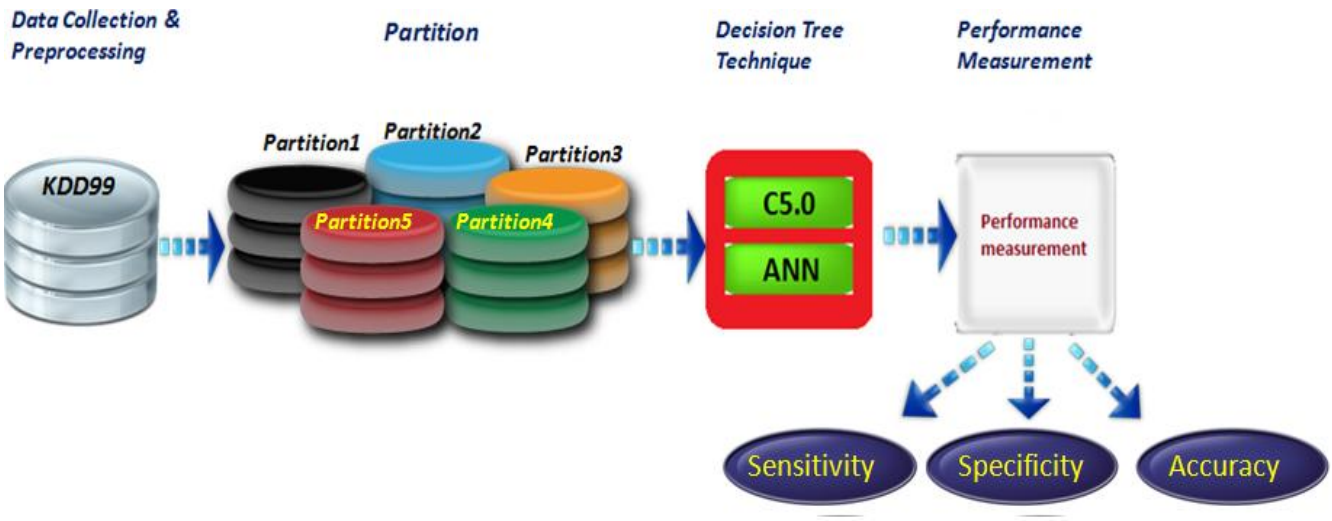

Figure 1. Process of Developing and Testing Models using C5.0 and ANN Technique.

In third phase decision tree technique as C5.0 and ANN technique as EBPN are used to develop models using Clementine software version 12.0 under Windows environment and i3 processor. Models are measured in terms of following statistical formulae as given below:

- **Accuracy** of a classifier on a given test set is the percentage of test set tuples that are correctly classified by the classifier [8].
- **Sensitivity** is the proportion of positive tuples that are correctly identified. Sensitivity is also referred to as true positive rate [8].
- **Specificity** is the proportion of negative tuples that are correctly identified. Specificity is also referred to as true negative rate [8].

These measures are defined as:

$$\text{Sensitivity} = \frac{t\_pos}{pos} \quad (1)$$

$$\text{Specificity} = \frac{t\_neg}{neg} \quad (2)$$

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

Where t_pos is the number of true positives, pos is the number of positive tuples (i.e. pos = t_pos + f_neg), t_neg is the number of negatives and neg is the number of negative tuples (i.e. neg = f_pos + t_neg).

It can be shown that accuracy is a function of sensitivity and specificity:

$$\text{Accuracy} = \text{sensitivity}\frac{pos}{(pos+neg)} + \text{specificity}\frac{neg}{(pos+neg)} \quad (3)$$

IV. RESULT AND DESCUSSION

Experimental work is carried out using Clementine data mining tool under windows environment for five different partitions. A confusion matrix for these partitions is shown in table I for C5.0 technique. Diagonal of the table in case of each partition clearly reflects that model is self sufficient to identify five different categories of samples with minimum number of misclassification, say for example in case of partition1 195,581 samples are correctly classified of DoS category while 15 samples under this category falls under normal category. We can also observe that numbers of misclassified samples are either changing due to partition size. Similarly different partitions of data sets are also applied to ANN and confusion matrix obtained in this case is shown in table II. One can observe from this table that model is not performing well, samples related to U2R category of attack are not well classified, and all the samples related to this category falls in other category of attack .Similarly most of the samples related to Probe category are distributed to other categories. Situations are almost same in case of all other partitions. However model will be better in terms of accuracy but it will be not performing well in terms of other measures. From table I, values of all parameters like t_pos (True positive), t_neg (True negative), f_pos (False positive) and f_neg (False negative) are obtained for all five classes i.e. Dos, R2L, U2R, Normal and Probe. With the help of above values we have then calculated error measures in terms of accuracy, sensitivity and specificity using formulae discussed in section III. Results are shown in table III and IV respectively for C5.0 and ANN. Results obtained are really promising and almost 100% .If we will observe the table minutely we can see that there is little variations due to partition size, however C5.0 is performing better than ANN.

TABLE I. CONFUSION MATRIX OF DECISION TREE TECHNIQUE C5.0 AT TESTING STAGE

Partition	Actual Vs Predicted	DoS	R2L	U2R	Normal	Probe
Partition1 50:50	DoS	195,581	0	0	15	0
	R2L	6	523	1	18	4
	U2R	0	3	10	9	1
	Normal	12	1	6	48,767	6
	Probe	79	0	0	23	1971
Partition2 60:40	DoS	156,563	0	0	10	4
	R2L	1	441	1	12	3
	U2R	0	2	9	7	0
	Normal	8	1	5	39,028	6
	Probe	54	3	0	13	1,574
Partition3 70:30	DoS	117,460	0	0	7	1
	R2L	1	337	2	10	3
	U2R	0	0	8	6	0
	Normal	6	4	0	29,349	5
	Probe	40	0	0	5	1,197
Partition4 80:20	DoS	78,347	0	0	6	0
	R2L	0	214	1	7	1
	U2R	0	0	4	6	0
	Normal	3	1	1	19,465	3
	Probe	21	1	0	3	780

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

Partition5 90:10	DoS	39,338	0	0	6	0
	R2L	0	112	0	1	0
	U2R	0	0	1	3	0
	Normal	1	0	0	9,871	2
	Probe	13	0	0	1	377

TABLE II. CONFUSION MATRIX OF DECISION TREE TECHNIQUE ANN AT TESTING STAGE

Partition	Actual Vs Predicted	DoS	R2L	U2R	Normal	Probe
Partition1 50:50	DoS	194,460	93	02	1,039	02
	R2L	0	481	0	71	0
	U2R	0	09	0	14	0
	Normal	1	196	0	48,593	02
	Probe	52	302	272	114	1,333
Partition2 60:40	DoS	156,512	0	0	53	12
	R2L	176	139	0	143	0
	U2R	10	0	0	8	0
	Normal	87	23	0	38,926	12
	Probe	06	0	0	104	1,534
Partition3 70:30	DoS	117,416	0	0	45	07
	R2L	146	75	0	132	0
	U2R	05	0	0	09	0
	Normal	56	13	0	29,283	12
	Probe	07	0	0	62	1,173
Partition4 80:20	DoS	77,910	68	0	375	0
	R2L	0	189	0	34	0
	U2R	0	05	0	05	0
	Normal	11	60	0	19,398	04
	Probe	37	123	93	45	507
Partition5 90:10	DoS	39,338	0	0	4	02
	R2L	0	98	0	15	0
	U2R	01	02	0	01	0
	Normal	02	19	0	9,840	03
	Probe	20	01	20	96	254

TABLE II. VARIOUS MEASURES OF C5.0 MODEL AT TESTING STAGE

Partition	Accuracy					Sensitivity					Specificity				
Training: Testing	*DoS*	*R2L*	*U2R*	*Normal*	*Probe*	*DoS*	*R2L*	*U2R*	*Normal*	*Probe*	*DoS*	*R2L*	*U2R*	*Normal*	*Probe*

50:50	99.95	99.98	99.99	99.96	99.95	99.99	94.75	43.48	99.95	95.08	99.81	99.99	99.99	99.97	99.99
60:40	99.96	99.99	99.99	99.97	99.96	99.99	96.29	50.00	99.95	95.74	98.85	99.99	99.99	99.97	99.99
70:30	99.96	99.98	99.99	99.97	99.96	99.99	95.47	57.14	99.95	96.38	99.85	99.99	99.99	99.97	99.99
80:20	99.97	99.98	99.99	99.97	99.97	99.99	95.96	40.00	99.96	96.89	99.88	99.99	99.99	99.97	98.06
90:10	99.96	99.99	99.99	99.97	99.97	99.98	99.12	25.00	99.97	96.42	99.86	99.99	100	99.97	99.99

TABLE III. VARIOUS MEASURES OF ANN MODEL AT TESTING STAGE

Partition	**Accuracy**					**Sensitivity**					**Specificity**				
Training: Testing	***DoS***	***R2L***	***U2R***	***Normal***	***Probe***	***DoS***	***R2L***	***U2R***	***Normal***	***Probe***	***DoS***	***R2L***	***U2R***	***Normal***	***Probe***
50:50	99.52	99.77	99.88	99.42	99.70	99.42	87.14	0	99.60	64.30	99.89	99.76	99.89	99.38	99.99
60:40	99.83	99.83	99.99	99.78	99.93	99.99	30.35	0	99.67	93.31	99.99	99.99	100	99.81	99.99
70:30	99.82	99.80	99.99	99.78	99.94	99.95	21.25	0	99.72	94.44	99.31	99.99	100	99.79	99.99
80:20	99.50	99.78	99.89	99.46	99.69	99.43	84.75	0	99.61	62.98	99.76	99.74	99.90	99.42	99.99
90:10	99.94	99.92	99.95	99.72	99.71	99.98	86.73	0	99.76	64.96	99.77	99.95	99.96	99.71	99.99

V. FEATURE SELECTION

Feature subset selection [9] is an important problem in knowledge discovery, not only for the insight gained from determining relevant modeling variables, but also for the improved understandability, scalability, and, possibly, accuracy of the resulting models. In the Feature selection the main goal is to find a feature subset that produces higher classification accuracy. Feature selection [10] is an optimization process in which one tries to find the best feature subset, from the fixed set of the original features, according to a given processing goal and feature selection criteria, without feature transformation or construction. The existing feature selection methods depending on feature selection criterion used two main streams: first are open-loop methods and second are closed-loop methods. The open-loop methods, also called the filter, present bias, or the front end methods, are based mostly on selecting features using between-class separability criteria. These methods do not consider the effect of the selected features on the entire processing algorithm's performance. Instead, they select these features for which the resulting reduced data set has maximal between–class separability, defined usually based on between–class and between-class covariance (or scatter matrices) and their combination. Ignoring the effect of the selected feature subset on the performance of classifier is a weak side of the open-loop methods. The closed-loop methods called also the wrapper, performance bias, or classifier feedback methods, are based on the feature selection using a classifier performance as criterion of feature subset selection. The closed-loop methods will generally provide a better selection of subset, since they based on the unlimited goal of optimal feature selection, which is providing the best classification. Feature selection technique with feature raking is applied to select best feature subset. The simple feature selection procedure is based on evaluate of classification power of individual features, then ranking such evaluated features, and eventually selecting the first best m features. A criteria applied to an individual feature could be of either of the open-loop or closed-loop type. This is also relies on an assumption that the final selection criterion can be expressed as a sum or product of the criteria evaluated for each feature independently. We can expect that a single feature alone have a low classification power. However, this feature when put together with others may exhibit substantial classification power. Features reduced in case of various partitions is shown in table V .C5.0 with 36 number of features in case of 90:10 partition is performing well with almost 100% accuracy .

TABLE IV. FEATURE SELECTION WITH C 5.0 AND ANN

Technique	**Partition**	**Feature**	**Accuracy**

			Training	Testing
C5.0	50:50	41	99.94	99.93
	60:40	41	99.94	99.93
	70:30	41	99.94	99.94
	80:20	41	99.94	99.95
	90:10	41	99.94	99.95
C5.0	50:50	36	99.94	99.93
	60:40	36	99.94	99.94
	70:30	36	99.94	99.94
	80:20	36	99.94	99.94
	90:10	36	99.95	99.95
C5.0	50:50	34	99.93	99.91
	60:40	34	99.92	99.91
	70:30	34	99.93	99.92
	80:20	34	99.93	99.93
	90:10	34	99.94	99.94
C5.0	50:50	32	99.92	99.90
	60:40	32	99.93	99.91
	70:30	32	99.93	99.91
	80:20	32	99.93	99.92
	90:10	32	99.94	99.93
ANN	50:50	41	99.11	99.12
	60:40	41	99.69	99.68
	70:30	41	99.67	99.67
	80:20	41	99.12	99.13
	90:10	41	99.38	99.40
ANN	50:50	36	98.76	98.76
	60:40	36	99.30	99.31
	70:30	36	99.24	99.25
	80:20	36	98.99	99.00
	90:10	36	99.03	99.06
ANN	50:50	34	99.15	99.16
	60:40	34	98.98	99.01
	70:30	34	99.01	99.04
	80:20	34	99.23	99.25
	90:10	34	99.36	99.36
ANN	50:50	32	98.87	98.88
	60:40	32	99.02	99.02
	70:30	32	99.01	98.99
	80:20	32	99.01	98.99
	90:10	32	98.48	99.49

VI. CONCLUSION

Intrusion detection is necessary for transmission of huge amount of data and information over public network and at the same time security of data is important. In order to protect data and information from various types of attack a novel intrusion detection system is required. This study explores use of C5.0 decision tree technique and ANN technique to classify intrusion data based on their partition size. Five different partitions are made to check the performance of model after feeding KDD99 data set. A comprehensive result show that C5.0 is performing better in case of 90-10 partition as error measures are almost near to 100% in this case. Feature selection technique is also applied in case of both the techniques. A comparative result proves that C5.0 is performing better than ANN and produces best result with 36 features.

REFERENCES

1. S.Y. Wua, and E. Yen, "Data Mining based intrusion detectors", Expert Systems with Applications, 36, 2009, 5605-5612.
2. G. Wang, J. Hao, J. Ma, and L. Huang, " A new approach to intrusion
3. detection using Artificial Neural Networks and Fuzzy Clustering", Expert Systems with Applications, 37, 2010, 6225-6232.
4. V.B. Canedo, N.S. Marono, and A.A. Betanzos , "Feature Selection and
5. Classification in Multiple Class Databases: An Application to KDDcup99 dataset", Expert Systems with Application, 38, 2011, 5947-5957.
6. R.M. Elbasionary, E.A. Sallam, T.E. Eltobely, and M.M. Fahmi,"A Hybrid network intrusion detection framework based on random Forest and weighted k-means", Aim Shams Engineering Journal,4, 2013,753-762
7. Z.A Baig., S.M Sait., and A.R. Shaheen, "GMDH-based networks for intelligent intrusion detection", Engineering Applications of Artificial Intelligence, 26, 2013, 1731-1740.
8. B. Luo, and J. Xia, " A novel intrusion detection system based on feature generation with visualization strategy",Expert Systems with Applications, 41, 2014, 4139-4147.
9. P. Giudici, and S. Figini, "Applied Data Mining for Business and Industry", 2nd ed., John Wiley & Sons, April 2009.
10. J. Han, and M. Kamber, "Data Mining Concepts and Techniques", 2nd ed., Morgan Kaufmann Publishers, USA, 2006.
11. J. Wang. "Data Mining: opportunities and challenge", Idea Group, USA, 2003.
12. K, J. Cios, W.Pedrycz, R.W. Swiniarski, and L. Kurgan. " Data mining
13. methods for knowledge discovery", 3rd printing, kluwer academic Publishers, USA, 2000.
14. SPSS Clementine help file http//www.spss.com last accessed on Oct 2012.
15. UCI Machine Learning Repository of machine learning databases (2010). University of California, school of Information and Computer Science, Irvine. C.A.
16. http://www.ics.uci.edu/~mlram,?ML.Repositary.html.