ISSN:0975-9646
Vivek P.Kshirsagar et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (5) , 2015, 4825-4828

Comparative Analysis of Various Classifiers for Performance Improvement in Intrusion Detection System by Reducing the False Positives

Vivek P.Kshirsagar[1],Dr.Madhuri Joshi[2]

Research Scholar,Department of Electronics and Telecommunications
Government Engineering College ,Railway Station Road,Aurangabad,Maharashtra,India

Professor,Department of Computer Science and Engineering,
Jawahalal Nehru Engineering College,Cidco,Aurangabad,Maharashtra,India

***Abstract*— With the growth of cyber-attacks as observed over the last couple of decade, safety, protection and privacy of information has become a major concern for organizations across the globe. Intrusion detection systems (IDSs) have thus gained important place and play a key role in detecting large number of attacks. There are a number of intrusion detection systems in market and most of them have the problem of having a relatively large number of false positives. Hence a need has arisen in the networking society of addressing the issue of false alarm and false positives and has resulted in an interest for researchers in IDS area. The main motivation of this research is in enhancing the performance of different data mining techniques to handle the alerts, reduce them and classify real attacks and reduce false positives .In this paper, the authors propose the use of algorithms C4.5 and Naïve Bayesian algorithms to lower the rate of false positives. The algorithms are first trained for detecting attacks on KDD99 Dataset and then are tested on live traffic to classify whether the flow is normal or there are attacks. The results established that C4.5 algorithm with a factor of .75 efficiently detects and classifies the attacks with significantly reduced false positives. Naive Bayesian algorithm statistically validates the experimental results.**

***Keywords*- C4.5, Detection rate, False Positives, Naive Bayes Classifier, Network intrusion detection**

I. INTRODUCTION

The objective of intrusion detection system is to detect and try to prevent hostile attacks in the network by malicious users (hackers).It relies on the ability to provide views of unusual activity and issuing alerts accordingly. The administrators can then take suitable actions by blocking or removing from network suspicious connections. As discussed in [1] all computer systems are vulnerable to all kinds of attacks and threats and most of the time these goes unnoticed. Hence the aim is to build an intrusion detection system that can capture live traffic, store it in the form of packets and analyse whether it is attack or normal packet. Machine learning or intelligent approach first came into forefront for audit data which were mined using the technique of association rule mining.[2]

Bayesian probability approach was used in [3] to reduce the false alarm rate. Misclassification of packets is common in any intrusion detection system and many researchers have focussed their interest in reducing the false positive rates and for the KDD dataset in [4] an approach of rough set theory was implemented to select the features best suitable for classification. The intrusion detection system should run continuously requiring minimal human supervision and withstand targeted malicious attacks. [5] It functions to monitor and resist local intrusion by utilizing minimal resources. It also adapts so as to function in large and fast networks. One key feature of the intrusion detection system is to have lower rate of false positives.

II. INTRUSION DETECTION OVERVIEW

The data mining algorithm framework as shown in fig.1 computes activity patterns from system audit data and extract predictive features from the patterns.[6][7] Machine learning algorithms C4.5 and Naïve Bayes algorithms are then applied to the KDD Dataset for training purposes. Raw data is first captured in the form of packet and interpreted in the form of connection records containing a number of features, such as service, duration, source IP address, destination IP address etc. The anomaly detector detects intrusions. On classification of the packet or traffic by the selected classification algorithm, Alarm Manager signals an alarm to the appropriate action taking entity to perform

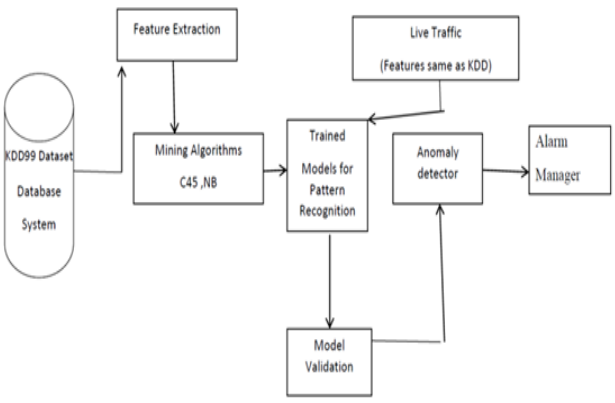

Fig. 1 Architecture of intrusion detection system

III. MATERIALS

The KDD Cup 1999 is being made use of in order to train the data mining algorithms. The algorithms are trained to be able to recognize the following attacks that are grouped into four major categories:

1. DOS: Denial of service
2. Probing
3. U2R
4. R2L

IV. PROPOSED SYSTEM

The proposed system consists of various modules like Packet Capture, Feature selection, Data Mining algorithms and evaluation metrics. The functions of each module are explained below:-

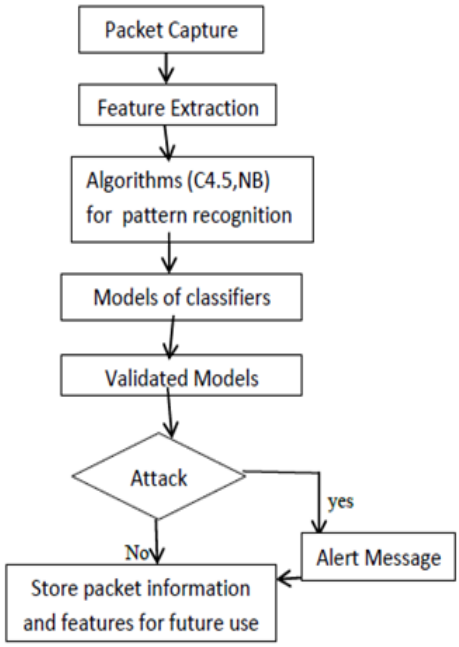

Fig. 2 Flow of Proposed System

A) PACKET CAPTURE

Capture of packets is carried out by using Open Source Package named JPcap. Jpcap is a Java library that uses the C library libpcap, for capturing and sending network packets. The traffic is logged in database for pattern matching by comparing those with the already defined signatures for labeled classification in an offline environment. The classification algorithm is implemented using NetBeans IDE, Java, and Weka. WinPcap is a tool available under windows for link-layer network access [9] the classified packets are indicated by providing separate color coding for valid and invalid packets. The authors have used MYSQL for offline storage. In our system, patterns are labeled based on criteria (TCP RFC standards) presented in following Table 1:

Table 1 Flag conditions, Packet validation and recommended action

If	Validation	Action
ACK = 0 & FIN = 1	Invalid	DROP
ACK = 0 & PUSH = 1	Invalid	DROP
ACK = 0 & RST = 0 & SYN = 1	Invalid	DROP
ACK = 0 & URG = 1	Invalid	DROP
FIN = 1 & SYN = 1	Invalid	DROP
RST = 1 & SYN = 1	Invalid	DROP
ACK_VALUE ≠ 0 & ACK = 0	Invalid	DROP

B) Feature Selection

The data available for constructing the system consists of a large amount of packets of trained data and test data. The connections are in chronological order. Each connection is described by 40+ features. The features are categorized as follows:[2][4][8]

i) TCP features

These features include the duration, protocol type, and service of the connection, as well as the amount of data transferred.

Login features

These features were derived from the payload of the TCP packets using domain knowledge. They include features like the number of failed login attempts and whether or not root access was obtained.

ii) Time stamp features

Calculated over a two second time interval, these features include things like the number of connections to the same host as the current connection and the number of connections to the same service as the current connection.

iii) Host traffic features

Similar to the time based traffic features, catching attacks of more than 2 seconds.

From the available features, eight were selected for use in the system. Features selected for Experimental Analysis:-

1. Intrusiontype{BSDtype,PING1MicrosoftWindows1,Ping3-O-MeterWindows,Pinger Windows2,ICMPPINGWindows,AlternateAddress,UnreachableHost,DestinationNetworkUnknown,PrecedenceViolation,Reply,Echoundefinedcode,I-Am-Here,IPV6Where-Are-You}
2. Month {Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec}
3. Day {1 to 31}
4. Sip{IP address of source machine}
5. Dip {IP address of destination machine}
6. Detect {yes, no}
7. Protocol type {ICMP,UDP,TCP}
8. Intrusion class {DOS, Normal, Probe, U2R, R2L}

V. Algorithms and Techniques used for Experimentation

As seen in the work of many researchers [9] for automatically tuning the intrusion detection system ,the authors here have employed suitable data mining techniques to classify attacks from live traffic and enhance the performance of the system.

1) Naïve Bayesian algorithm

The Bayesian IDS is built out of a naïve Bayesian classifier. The classifier is anomaly based. It works by recognizing that features have different probabilities of occurring in attacks and in normal TCP traffic. The algorithm is trained by giving it classified traffic. It then adjusts the probabilities for each feature. After training, the algorithm calculates the probabilities for each TCP connection and classifies it as either normal TCP traffic or an attack. [10]

2) C4.5

It builds decision trees from a set of instances used as training data. For building the tree it incorporated the concept of information entropy. The instances from the training data are classified into one of the five classes... Each instance has different attributes. For building the tree C4.5 [11] chooses any one attribute with the highest gain value that best splits the instances into subsets as belonging in one of the classes. The tree is pruned by applying various confidence factors.

3) C4.5 with Multiboosting

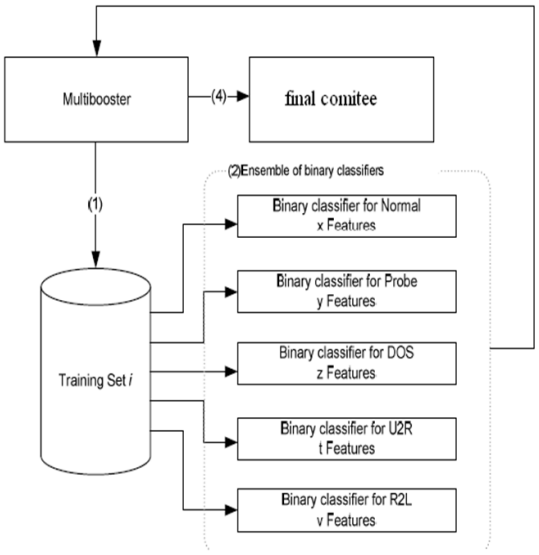

Fig 3 System Model for IDS using C4.5 and Multiboosting

As seen from fig 3,initially five classifiers are defined one for each of KDD dataset attack types. The classifiers are trained using C4.5 algorithm to detect attack from normal packet. The errors during classification are again back-propagated to the classifiers and this continues until accuracy is improved. This results in a most accurate final classification. Then the technique of multiboosting is applied to form a decision committee. As real time-traffic is captured hence dynamic multiboosting technique is employed. In this method, the packets that are classified as attacks are stored in the database. Hence the IDS in real – time will check the contents of the database for the packets. If it detects an attack it issues an alert. If any match is not found for the various attacks then a normal classification is done for the packet.

VI. Evaluation Metrics

True positive: It is defined that the attack is correctly classified.

$$TPR = TP/ (TP+FN)$$

False Negative: It occurs when the attack is incorrectly predicted as negative when it is actually positive.

False positive: It occurs when the attack is incorrectly predicted as yes when in reality it should be no. [12]

$$FPR = FP / (TN + FP)$$

Accuracy: It is defined by the following formula:-

$$Accuracy=TP/ (TP+FP)$$

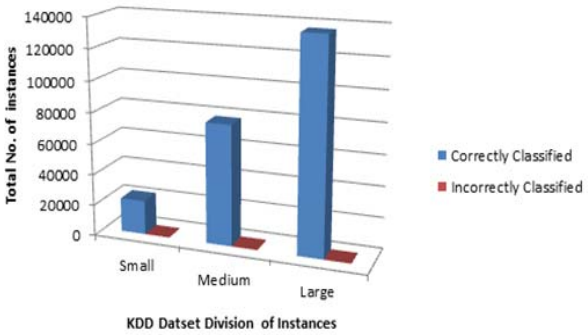

Fig 4 Accuracy of C4.5 on factor of .75 when the KDD Dataset is divided into small, medium and large instances

Fig 4 illustrates graphically the results on applications of the metrics and shows that when used with a factor of .75 and on division of the dataset into three ranges of instances the correctly and incorrectly classified instances.

VII. Experimentation And Results Discussions

Table 2 compares the performance of the algorithms C4.5 NB and AB after they are trained to detect the five classes of attacks from the KDD Dataset.

KDD Attack	KDD Count	C4.5	NB	AB
U2R	70	67	76	73
R2L	14745	5636	5621	5550
PROBE	4156	4129	4714	4323
NORMAL	80593	64747	67885	68726
DOS	231455	232450	232733	232357

Table 2 displays the five classes from the KDD Dataset with instances for each class and the performance results of the algorithms C4.5 and NB.As seen in fig 4. On application of different factors for C4.5 algorithm the performance gets slightly improved for a factor for .75 for the five attack classes.

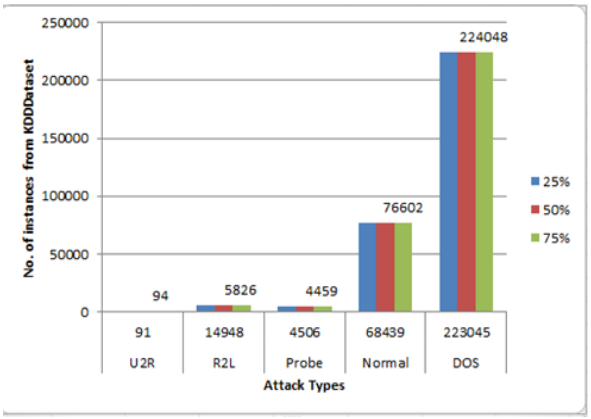

Fig 5 Attacks as classified by C4.5 algorithm on various confidence factor (.25, .50 and .75)

Table 3 Percentage of accuracy and error rate of C4.5 algorithm on large dataset for various confidence factors.

Value	Total Instances	Correctly Classified Instances	Mis-classified instance	% Accuracy	% Error Rate
0.25	136650	136548	102	99.9254	.0746
.40	136650	136551	99	99.9276	.0724
.75	136650	136558	92	99.9327	.0673
.80	136650	136557	93	99.9319	.0681

The error rate on the same number of instances for the algorithm C4.5 when tested on four distinct factor values gets decreased on the factor value of .75 which means less number of false positives. Fig 5 illustrate in numbers the significant decrease in false positives generated for attack classes U2R and Probe when C4.5 algorithm is applied against Naïve Bayes Classifier.

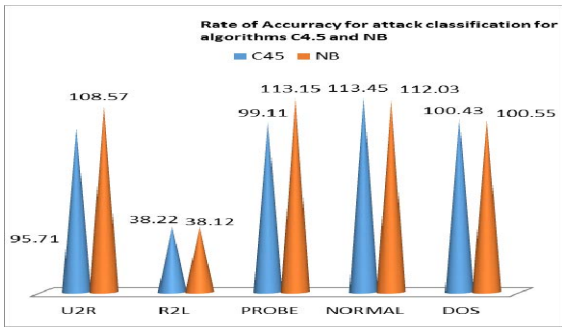

Fig. 6 Comparative Analysis of algorithms C4.5 and NB for accurately classifying various attacks

VIII. Conclusions

The entire network intrusion detection framework was developed using Matlab environment with java packages. The KDD dataset was used to train the algorithms for the 5-classes (normal, dos, probe, u2r and r21).C4.5 constructs decision trees by using features to try and split the training set into positive and negative active examples until it achieves high accuracy on the training set. NB tree segments the data using a univariate decision tree. Each leaf is a naïve bayes classifier class with a probabilistic summary, and finds the most likely class for each example it is asked to classify. Once the algorithms were trained they were used to detect attacks form live traffic. For a duration of 20 minutes C4.5 classified live traffic as (R2L:123Probe:2, Normal: 6754, DOS: 110) and Naïve Bayes classified it as (Normal: 6947, DOS: 42). From the results it is inferred that the algorithm C4.5 achieves a high accuracy at detecting attacks for a confidence factor of .75 and thus preventing false positives to a greater extent. Naïve Bayesian algorithm statistically validates the results.

References

[1] Mahbod Tavallaee, Ebrahim Bagheri, Wei Lu, and Ali A. Ghorbani," A Detailed Analysis of the KDD CUP 99 Data Set", in *PROC CISDA,*, 2009

[2] Adetunmbi A.Olusola., Adeola S.Oladele. and Daramola O.Abosede," Analysis of KDD '99 Intrusion Detection Dataset for Selection of Relevance Features", *World Congress on Engineering and Computer Science*, Vol.1,2010

[3] Hesham Altwaijry, Saeed Algarny," Bayesian based intrusion detection system", *Journal of King Saud University - Computer and Information Sciences*, Vol. 249(1), pp 1–6,2012

[4] Suthaharan, S,, Panchagnula, T.," Relevance feature selection with data cleaning for intrusion detection system", in *Proc IEEE*, p. 1–6,2012

[5] Dahlia Asyiqin Ahmad Zainaddin, Zurina Mohd Hanapi ," Hybrid of fuzzy clustering Neural network over nsl dataset for intrusion detection system ", *Journal of Computer Science*, vol.9 (3), pp. 391-403, 2013

[6] Zhan Jiuhua ; Leshan Teachers Coll., Leshan**,"** Intrusion Detection System Based on Data Mining", First International Workshop on Knowledge Discovery and Data Mining,23-24 Jan. 2008, Adelaide, SA, pp. 402 - 405,2008

[7] Yusufovna, S.F.," Integrating Intrusion Detection System and Data Mining", International Symposium on Ubiquitous Multimedia Computing, Hobart, ACT, Oct 13-15 ,pp. 256 – 259,2008

[8] H. Günes Kayacık, A. Nur Zincir-Heywood, Malcolm I. Heywood," Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets", Dalhousie University, Faculty of Computer Science, 6050 University Avenue, Halifax, Nova Scotia.

[9] Zhenwei Yu, Weigert, T. ," An Automatically Tuning Intrusion Detection System", *IEEE Transactions on Systems ,Man, and Cybernetics*, Vol.37(2),pp 373-384,2007

[10] Nong Ye, Xiangyang Li,, Qiang Chen, Syed Masum Emran, Mingming Xu," Probabilistic Techniques for Intrusion Detection Based on Computer Audit Data",*IEEE transactions on systems, man, and cybernetics—part a: systems and humans*, vol. 31(4), pp. 266-274, July 2001

[11] V.Balaji,Varalakshmi K,"Differentiating network attacks using C4.5 algorithm with multiboosting",*International journal of emerging technologies in computational and applied sciences*, vol.4(3),pp.231-235,March-May 2013

[12] Asieh Mokarian, Ahmad Faraahi, Arash Ghorbannia Delavar," False Positives Reduction Techniques in Intrusion Detection Systems-A Review", *International Journal of Computer Science and Network Security*,vol.13(10),pp.128-134,2013