Dimension Reduction in Intrusion Detection Features Using Discriminative Machine Learning Approach

Karan Bajaj[1], Amit Arora[2]

**[1] Computer Science and Engineering Department, Chitkara University
Baddi, Himachal Pradesh, India**

**[2] Computer Science and Engineering Department, Chitkara University
Baddi, Himachal Pradesh, India**

Abstract

With the growing need of internet in daily life and the dependence on the world wide system of computer networks, the network security is becoming a necessary requirement of our world to secure the confidential information available on the networks. Efficient intrusion detection is needed as a defence of the network system to detect the attacks over the network. Using feature selection, we reduce the dimensions of NSL-KDD data set. By feature reduction and machine learning approach, we are able to build Intrusion detection model to find attacks on system and improve the intrusion detection using the captured data.

Keywords: *Feature Selection, Weka, NSL-KDD Data Set, Accuracy.*

1. Introduction

As the growing need of internet in our daily life and our dependence on the world wide system of computer networks, the network security is becoming a necessary requirement of our world to secure the confidential information available on the networks. The precious information is always prone to maximum attacks over the network. Intrusion may occur due to system vulnerabilities or security breaches, such as system misconfiguration, user misuse or program defects. Attackers can also combine multiple security vulnerabilities into an intelligent intrusion. Intrusion detection plays an important role over the large network system. In a big network system there are large number of servers and on-line services running in the system while such networks may lure more attackers. Efficient intrusion detection model is needed as a defense of the network system [1].

2. Data Set

To test the Intrusion Detection, we use NSL-KDD [2] data set which consists of selected records of the complete KDD'99 [3] data set. In NSL-KDD [2] data set redundancy of the data in the original KDD'99 [3] data set is reduced, which makes the data more realistic for attack detection. The data set contains 41 features and 1 class labeled as Normal and Anomaly.

Data set is divided into separate Train and Test data on which evaluation is performed, with a total of 24 training attack types in the train set [8], with an additional 14 types [8] in test data set. This makes the detection more realistic because now the model is also checked for the unknown attacks.

3. Related Research

The Lightweight Network Intrusion Detection System, LNID, is proposed system for intrusion detection. The filtering scheme proposed consists of two packet filters: Tcpdump Filter and LNID Filter. The former one processes initial packet filtering with tcpdump tool, extracting TCP packets towards Telnet servers of internal local area network [1]. In [4], the authors purpose Intrusion detection using several Decisions Trees and Decision Rules. The prediction accuracy of classifiers was evaluated using 10-fold cross validation, due to cross validation the obtained accuracy was only for the known attacks. Extended security for intrusion detection system using data cleaning in large database [5], this process works on matching policies in database with anomalous information. So, it works well when the policy is matched, therefore technique is good for known attacks whose policies are already defined. Light weight agents for

intrusion detection, this approach is designed and implemented for intrusion detection system (IDS) prototype based on mobile agents [6], but limited for only mobile agents. IP Flow-Based Intrusion Detection [7], this approach find the attack contents by monitoring every packet. However, packet inspection cannot easily be performed at high-speeds. Mahbod Tavallaee, Ebrahim Bagheri, Wei Lu, and Ali A. Ghorbani [8], demonstrated the use of multiple machine learning algorithms on their proposed NSL-KDD [2] data set which was free from the redundant data which was in KDDCUP'99 [3]. They use separate training and test set which makes the detection more accurate for unknown attacks.

Literature survey showed that, for the practical, most of the researchers had used KDDCUP'99[3] which suffers from drawback of redundant data, which leads to the biasing in detection of attacks which are more frequent in data sets like DOS and PROBE attacks. Some researchers had applied single algorithm to detect all the attack types or they had used cross validation on data set which is good only for the detection of already known attacks. The researchers that have used the NSL-KDD [2] data set with multiple machine algorithms [8] did not try further any attribute selection measures to improve the accuracy. This motivated us for our assumption that using NSL-KDD [2] data set with different training and test set separately and attribute selection with different machine learning algorithms will yield good performance and improve prediction for detection of attacks including unknown attacks as well.

4. Attribute Selection Measures

For the attribute selection, different feature selection algorithms are used they find the contribution of the 41 features in NSL-KDD [2] data set in intrusion detection. Feature selection reduces the features from the data set without affecting the effective indicators of system attacks.

4.1. Information Gain Attribute Evaluation:

Information Gain Attribute Evaluation evaluates the worth of an attribute by measuring the information gain with respect to the class [9].

$$\text{Info (G)} = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

Here Information gain G is calculated by calculating the probability of occurrence of class over total classes in data set.

4.2. Gain Ratio Attribute Evaluation:

It uses an extension to the information gain uses the gain ratio [9]

$$\text{Gain Ratio (A)} = \text{Gain (A)}/\text{Split Info (A)}$$

This value represents the potential information generated by splitting the training data set.

4.3. Correlation Attribute Evaluation:

Correlation specifies dependence of feature on each other. It represents the linear relationship between the variables or features.

$$r_{A,B} = \frac{\sum_{i=1}^{N}(a_i - \bar{A})(b_i - \bar{B})}{N\sigma_A\sigma_B} = \frac{\sum_{i=1}^{N}(a_i b_i) - N\bar{A}\bar{B}}{N\sigma_A\sigma_B},$$

Here N is the number of tuples, a_i and b_i is the respective values of A and B in tuple i, $\overline{A}$ and $\overline{B}$ are the respective mean values of A and B, σ_A and σ_B are the respective standard deviations of A and B [9]. The value of $r_{A,B}$ lies between -1 and 1. If A and B are completely correlated, $r_{A,B}$ takes the value of 1, if A and B are inversely correlated then $r_{A,B}$ takes value of -1and if A and B are totally independent then $r_{A,B}$ is zero.

5. Implementation Setup & Methodology

From feature selection and machine learning algorithms we will be able to collect the result data through which we can identify and predict the machine learning techniques that helps to distinguish between alerts, attacks and normal data. Our purpose is to suggest a learning model to reduce the false alarms and improves detection of attacks.

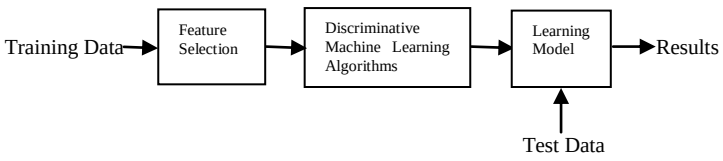

Fig.1 Implementation Setup Model

5.1. Feature Selection

For the attribute selection, we use different feature selection methods.

5.1.1. Information Gain Attribute Evaluation, we process the NSL-KDD [2] train set and retrieves the results. This algorithm use rankers method on features and evaluate the feature by ranking them from most important to least important.

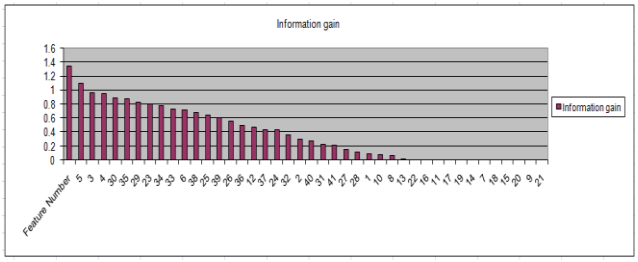

Fig.2 Information Gain Attribute Evaluation

Developed By: Karan Bajaj Date:24/04/2013 Aid:Weka(3.6.9)

5.1.2. We use training data and apply Gain Ratio Attribute Evaluation algorithm on data, this algorithm use rankers method on features and evaluate the feature by ranking them.

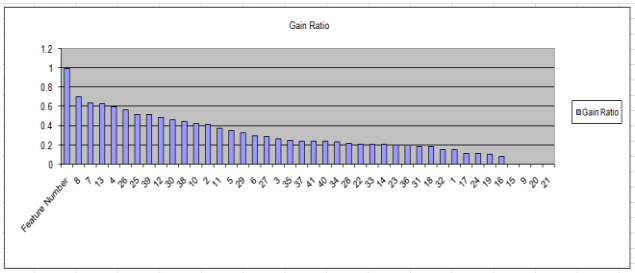

Fig.3 Gain Ratio Attribute Evaluation

Developed By: Karan Bajaj Date:24/04/2013 Aid:Weka(3.6.9)

Information gain measure is biased towards tests with many outcomes. Gain Ratio prefers to select attributes having a large number of values. It uses an extension to the information gain.

5.1.3. Correlation Attribute Evaluation, this algorithm rank the features in NSL-KDD [2] train set based on their correlation with each other, correlation specify dependence of feature on each other.

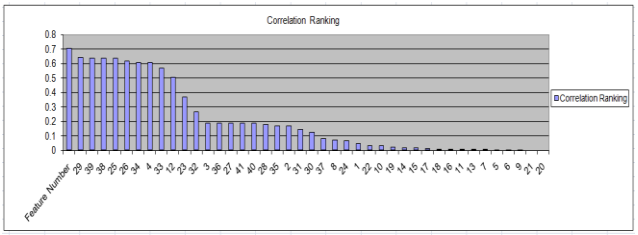

Fig.4 Correlation Attribute Evaluation

Developed By: Karan Bajaj Date:24/04/2013 Aid:Weka(3.7.9)

5.2. Dimension Reduction:

From the three feature selection methods applied on NSL-KDD [2] training data set (Fig 2,3 and 4), we come to find that feature number 9, 20 and 21 (urgent, num_outbound_cmds and is_host_login) have no role in detection of any attack and further 15, 17, 19, 32 and 40 (su_attempted, num_file_creations, num_access_files, dst_host_count and dst_host_rerror_rate) have minimum role in detection of attack.

On the basis of analysis of results of feature selection, we reduce the NSL-KDD [2] data set. In both train and test data set the dimensions of data set is reduced by removing the feature numbers 9, 15, 17, 19, 20, 21, 32 and 40.

5.3. Discriminative Machine Learning Algorithms

On the reduced data set, we applied several discriminative machine learning algorithms, now training sets are given to train the machine learning algorithms and test data set is given separately. Using separate train and test set give us advantage to check the accuracy of detection of attacks even on unknown attacks, because training set contain 24 attack types [8] and test set contain additional 14 attacks with previous 24 attacks [8]. This makes the detection more accurate because now the model is also checked for the unknown attacks.

5.3.1. NaïveBayes:

The NaiveBayes [10] classifier provides an approach to represent and learn the probalistic knowledge [11].

5.3.2. J48:

Is a tree classifier in Weka Tool [12], it is a version of C4.5 algorithm which was developed by Quinlan [13].

5.3.3. NB Tree:

NB Tree [14] builds a naive Bayes classifier on each leaf node of the built decision tree, which just integrates the advantages of the decision tree classifiers and the Naive Bayes classifiers [15].

5.3.4. Multilayer Perception:

It is a neural network classification algorithm [11].

5.3.5. LibSVM:

Support Vector Machines are supervised learning models with algorithms that analyze the data and recognize the patterns. LibSVM is integrated software for support vector classification, regression and distribution estimation. It supports multi-class classification [16].

5.3.6. SimpleCart:

Cart stands for classification and regression. Cart has the ability to generate the regression trees. It enables users to provide prior probability distribution [17].

6. Results & Performance Comparison

Table1 is representing the results, without feature selection on NSL-KDD [2] data set with 41 features and 1 class for Labels. The results are given in terms of accuracy of detection for various learning algorithms from previous benchmarks [8], except SimpleCart algorithm that is not used in previous benchmark paper.

Table 1 : Detection accuracy on NSS-KDD Test Data set

Classifier (Discrimitive Machine Learning Algorithms)	***Detection Accuracy (%)***	Incorrectly Classified Instances
J48 [8]	81.05	**
Naïve Bayes [8]	76.56	**
NB Tree [8]	82.02	**
Multi-layer Perception [8]	77.41	**
SVM [8]	69.52	**
SimpleCart	80.3229	19.6771

** Indicates information not provided by the author in their respective paper.

Table 2 is representing the results, after feature selection on NSL-KDD [2] data set. Now the feature is reduced from 41 features to 33 features and 1 class for Labels. The results shown in Table 2 are compared with previous benchmarks shown in Table 1.

Table 2 : Detection Accuracy on reduced Data set after dimension reduction

Classifier (Discrimitive Machine Learning Algorithms)	***Detection Accuracy (%)***	Incorrectly Classified Instances
J48	81.9375	18.0625
Naïve Bayes	75.7851	24.2149
NB Tree	80.6778	19.3222
Multi-layer Perception	73.5495	26.4505
LibSVM	71.0211	28.9789
SimpleCart	82.3235	17.6765

Result Analysis:

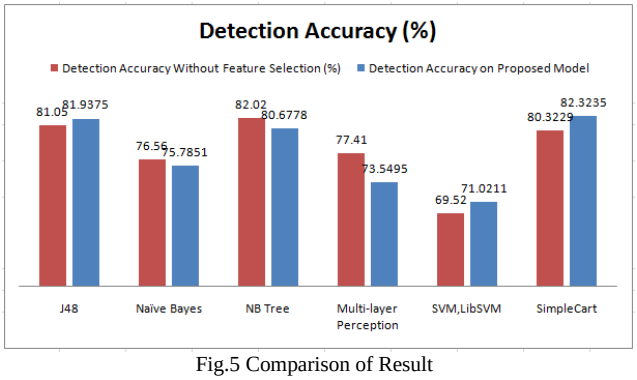

Fig.5 Comparison of Result

Fig.5 is representing the comparative analysis in terms of detection accuracy without feature selection and proposed model.

7. Conclusions

In this paper, we propose model for intrusion detection, which suggests that for the detection of intrusion, it is not necessary to perform the test on all the 41 features of NSL-KDD [2] data set. First by using feature selection the features are reduced to 33 features and further by removing them, the biasing of learning algorithms towards the frequent and easily detectable records in the data set is reduced. And the suggested machine learning algorithm after selection process is SimpleCart for the intrusion detection that leads to improve the computer security alerts from computer security incidents using machine learning techniques.

References

[1]Chia-Mei Chen, Ya-Lin Chen, Hsiao-Chung Lin, "An efficient network intrusion detection", Elsevier, Vol. 33, No. 4, 2010, pp. 477- 484.

[2]"Nsl-kdd data set for network-based intrusion detection systems. Available on: http://nsl.cs.unb.ca/NSL-KDD/, March 2009.

[3]KDD Cup (1999). Available on: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[4]G. Meera Gandhi, Kumaravel Appavoo , S.K. Srivatsa, "Effective Network Intrusion Detection using Classifiers Decision Trees and Decision rules" Int. J. Advanced Networking and Applications, Vol. 2, No. 3, 2010, pp. 686.

[5] R. Aarthy and P. Marikkannu, "Extended security for intrusion detection system using data cleaning in large database" International Journal of Communications and Engineering, Vol. 2, No. 2, 2012, pp. 56-60.

[6]Guy Helmer, Johnny S.K. Wong, Vasant Honvar, Les Miller, Yanxin Wang, "Lightweight agents for intrusion detection", Journal of systems and Software. Elsevier, Vol. 67, No. 2, 2010, pp.109-122.

[7]Anna Sperotto, Gregor Schaffrath, Ramin Sadre, Cristian Morariu, Aiko Pras and Burkhard Stiller, "An Overview of IP Flow-Based Intrusion Detection" IEEE communications surveys & tutorials, Vol. 12, No. 3, 2010, pp. 343.

[8]M. Tavallaee, E. Bagheri, W. Lu, and Ali A. Ghorbani, "A Detailed Analysis of the KDD CUP 99 Data Set" IEEE Symposium on computational intelligence in security and defence application, 2009.

[9]Jiawei Han and Micheline kamber, Data Mining Concepts and Techniques ,Publisher Elsevier, pp.67-69, 296-301, 2001.

[10] G. H. John, P. Langley, "Estimating Continuous Distributions in Bayesian Classifiers" In Proc. Of 11th Conference on Uncertainty in Artificial Intelligence, 1995.

[11] Huy Anh Nguyen and Deokjai Choi, "Application of Data Mining to Network Intrusion Detection: Classifier Selection Model", Springer-Verlag Berlin Heidelberg, LNCS 5297, 2008, pp. 399–408.

[12] "Waikato environment for knowledge analysis (weka) version 3.6.9. and 3.7.9" Available on :http://www.cs.waikato.ac.nz/ml/weka/
[13] J. Quinlan, : C4.5: Programs for Machine Learning, Publisher Morgan Kaufmann, San Mateo, 1993.
[14] R. Kohavi, "Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid," ser. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining. AAAI Press, 1996, pp. 202–207.
[15] Liangxiao Jiang and Chaoqun Li, "Scaling Up the Accuracy of Decision-Tree Classifiers: A Naive-Bayes Combination," Journal of Computers,Vol. 6, No. 4, 2011, pp.1325-1331.
[16] Chih-Chung Chang and Chih-Jen Lin, "LIBSVM : a library for support vector machines," ACM Transactions on Intelligent Systems and Technology, 2:27:1--27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm
[17] Lior Rokach and Oded Maimon, "DECISION TREES," Department of Industrial Engineering, Tel-Aviv University, Chapter-9, pp.181.

First Author: Karan Bajaj, Btech(Computer Science & Technology) from Himachal Pradesh University and pursuing M.E(Computer Science & Engineering) from Chitkara University. He is presently working as Assistant Professor in Department of Computer Science & Engineering at Chitkara University. He has more than 3 years of teaching experience to his credit. He has attended various workshops and short-term courses in different domains.

Second Author: Amit Arora, ME(Computer Science & Technology) from IIT Madras. He is presently working as Assistant Professor in Department of Computer Science & Engineering at Chitkara University. His Research areas are Machine Learning, Artificial Intelligence and Data Structures and Algorithms.