A Data Mining Approach for Intrusion Detection System Using Boosted Decision Tree Approach

[1]Priyanka B Bera, [2]Ishan K Rajani,
[1]P.G. Student, [2]Professor,
[1]Department of Computer Engineering,
[1]D.I.E.T, Rajkot, India

***Abstract* - Data mining is the technique to handle the large amount of data. These data must be secured from the suspicious things. New intelligent Intrusion Detection Systems (IDSs) which are based on sophisticated algorithms rather than current signature-base detections are in demand. There is often the need to update an installed Intrusion Detection System (IDS) due to new attack methods or upgraded computing environments. Since many current Intrusion Detection Systems are constructed by manual encoding of expert knowledge, changes to them are expensive and slow. In data mining-based intrusion detection system, we should make use of particular domain knowledge in relation to intrusion detection in order to efficiently extract relative rules from large amounts of records. This paper proposes new ensemble boosted decision tree approach for intrusion detection system. Experimental results shows better results for detecting intrusions as compared to others existing methods.**

***IndexTerms* – boosted decision trees, data mining, ensemble approach, network intrusion detection system**

I. Introduction

Security of our information is one of the cornerstones of information security. As number of network attacks increase over the past few years an encroachment catching method is becoming a critical component to protect the network. The biggest problem of the abuses is the intrusion that abuses weaknesses of network and systems. These attacks can destroy systems and network and overspread rapidly through the Internet. Encroachment catching method have undergone rapid growth in power, scope and complexity in their short history. In recent year, intrusion detection system has been most selected topic in the information security field. An encroachment catching method is a system for detecting intrusion and reporting them to the manager. This system are usually specific to the operating system that they operate in and are an important tool in the overall implementation of organization’s statement. This system is the issue of identifying unauthorized use of computer system . it detects the misuse of the our information of our computer or machine and diagnoses the abuse of machine. There are various system to classifies the methods and attacks are available for the detection.

There are various classified method generally use for the detection method:

– Misuse based detection system: misuse detection is one of an approach used in detecting different kind of an attack. In misuse detection approach, we have to first define the normal or abnormal behavior of the system. It stands against reverse of anomaly based detection. In misuse based detection data are compare in the dataset with other parameters. Like, a various detection system, misuse detection system is the good method for the abnormal data.

– Network based detection system: NIDS can be able to stand against large amount of network traffic to remain effective. As network traffic increases exponentially NIDS must collect all the traffic and evaluate in a timely manner. A network-based intrusion detection system (NIDS) is used to monitor and analyze network traffic to protect a system from network-based threats. A NIDS scans all inbound packets and look for any doubtful patterns. When threats are discovered, based on its grimness, the system responses by taking action such as notifying administrators, or barring the source IP address from accessing the network.

– Anomaly based detection system: Anomaly based detection system supervise the behavior in order to identify intrusion by detecting anomaly. It work on assumption of the other behavior differs from normal user behavior so, it can be detected easily. Events in an anomaly detection engine are caused by any behaviors that fall outside the predefined or accepted model of behavior. .Anomaly-based IDS systems can cause heavy processing overheads on the computer system. In this case we have two possibilities: (1) False positive: Anomalous activities that are not intrusive but are flagged as intrusive. False Negative: Anomalous activities that are intrusive but are flagged as non intrusive .

There were so many attack types available in the dataset. But these attacks fell in exactly four categories:

– Denial of service(Dos): attacker tries to prevent legitimate from using a service:

– Remote to local(R2L): Attacker does not have an account on the victim machine, hence tries to gain access.

– User to Root(U2R): Attacker has local access to the victim machine and tries to gain super user privileges.

– Probe: Attacker tries to gain information about target host.

There are various aspects of intrusion detection or encroachment catching method, due to which a researcher can become more familiar or quickly familiar with every aspect of intrusion detection. There are large number of technique use for the detection system.

II. RELATED WORK

Data mining is disciplines works to finds the major relations between collections of data and enables to discover a new and anomalies behavior. Data mining based intrusion detection techniques generally fall into one of two categories; misuse detection and anomaly detection. In misuse detection, each instance in a data set is labeled as ‘normal’ or ‘intrusion’ and a on learning algorithm is trained over the labeled data. These techniques are able to automatically retrain intrusion detection models different input data that include new types of attacks, as long as they have been labeled appropriately. Data mining are used in different field such as marketing, financial affairs and business organizations in general and proof it is success. The main approaches of data mining that are used including classification which maps a data item into one of several predefined categories. This approach normally output “classifiers” has ability to classify new data in the future, for example, in the form of decision trees or rules. An ideal application in intrusion detection will be together sufficient “normal” and “abnormal” audit data for a user or a program. The second important approach is clustering which maps data items into groups according to similarity or distance between them.

Anomaly detection techniques thus identify new types of intrusions as deviations from normal usage [7, 8]. In statistics - based outlier detection techniques [4] the data points are modeled using a stochastic distribution and points are determined to be outliers depending upon their relationship with this model. However, with increasing dimensionality, it becomes increasingly difficult and in-accurate to estimate the multidimensional distributions of the data points [1]. However, recent outlier detection algorithms that we utilize in this study are based on computing the full dimensional distances of the points from one another [9, 16] as well as on computing the densities of local neighborhoods [6].

Classifier construction is another important research challenge to build efficient IDS. Nowadays, many data mining algorithms have become very popular for classifying intrusion detection datasets such as decision tree, naïve Bayesian classifier, neural network, genetic algorithm, and support vector machine etc. However, the classification accuracy of most existing data mining algorithms needs to be improved, because it is very difficult to detect several new attacks, as the attackers are continuously changing their attack patterns. Anomaly network intrusion detection models are now using to etect new attacks but the false positives are usually very high. The performance of an intrusion detection model depends on its detection rates (DR) and false positives (FP).

Ensemble approaches [14, 17] have the advantage that they can be made to adopt the changes in the stream more accurately than single model techniques. Several ensemble approaches have been proposed for classification of evolving data streams. Ensemble classification technique is advantageous over single classification method. It is combination of several base models

and it is used for continuous learning. Ensemble classifier has better accuracy over single classification technique. Bagging and boosting are two of the most well-known ensemble learning methods due to their theoretical performance guarantees and strong experimental results. Boosting has attracted much attention in the machine learning community as well as in statistics mainly because of its excellent performance and computational attractiveness for large datasets.

III. PROPOSED APPROACH

This proposed model uses boosted decision tree i.e. hoeffding tree classification techniques to increase performance of the intrusion detection system. Boosted Decision Tree- The underlying idea of boosting is to combine simple rules to form an ensemble such that the performance of the single ensemble member is improved, i.e. boosted. Let h1, h2, …. hN be a set of hypotheses and consider the composite ensemble hypothesis,

$$f(x) = \sum_{n=1}^{N} \alpha_n h_n(x) \qquad (1)$$

Here n denotes the coefficient with which the ensemble member hn is combined; both n and the learner or hypothesis hn are to be learned within the boosting procedure. The boosting algorithm initiates by giving all data training tuples the same weight w0. After a classifier is built, the weight of each tuple is changed according to the classification given by that classifier. Then, a second classifier is built using the reweighted training tuple. The final classification of a intrusion detection is a weighted average of the individual classifications over all classifiers. There are several methods to update the weights and combine the individual classifiers. After the kth decision tree is built, the total misclassification error Epsilon k of the tree, defined as the sum of the weights of misclassified tuples over the sum of the weights of all tuples, is calculated:

$$\varepsilon_k = \sum_{i(miscl)} w_i^k \Big/ \sum_i w_i^k \qquad (2)$$

where i loops over all instances in the data sample. Then, the weights of misclassified tuples are increased

$$w_i^{k+1} \rightarrow w_i^{k+1} \Big/ \sum_i w_i^{k+1} \qquad (4)$$

$$w^{k+1} = \frac{1-\varepsilon_k}{\varepsilon_k} w_i^k \qquad (3)$$

and the tree k+1 is constructed. Note that, as the algorithm progresses, the predominance of hard-to-classify instances in the training set is increased. The final classification of tuple i is a weighted sum of the classifications over the individual trees. Furthermore, trees with lower misclassification errors "k is given more weight when the final classification is computed.

In decision tree i.e. hoeffding tree, each node contains a test on an attribute, each branch from a node corresponds to a possible outcome of the test and each leaf contains a class prediction. A decision tree is learned by recursively replacing leaves by test nodes, starting at the root. The attribute to test at a node is chosen by comparing all the available attributes and choosing the best one.

For classifying examples in the dataset, the prior and conditional probabilities generated from the dataset are used to make the prediction. This is done by combining the effects of the different attributes values from the example. Suppose the example ej has independent attribute values { ai1, ai2,…, aip}, we know P(aik | cj), for each class cj and attribute ajk and then estimate P(ej|cj) by

$$P(e_i \mid c_j) = P(c_j)\prod\nolimits_{k=1\rightarrow p} P(a_{ij} \mid c_j) \qquad (5)$$

To classify an example in the dataset, the algorithm estimates the likelihood that ei is in each class. The probability that ei is in a class is the product of the conditional probabilities for each attribute value with prior probability for that class. The posterior probability P(cj | ei) is then found for each class and the example classifies with the highest posterior probability for that example. The algorithm will continue this process until all the examples of sub-datasets or sub-sub-datasets are correctly classified. When the algorithm correctly classifies all the examples of all sub or sub-sub datasets, then the algorithm terminates and the prior and conditional probabilities for each sub or sub-sub-datasets are preserved for future classification of unseen examples.

In this proposed scheme boosting method improves ensemble performance by using adaptive window and adaptive size hoeffding tree as base learner. Because of this algorithm woks faster and increases performance. It uses dynamic sample weight assignment technique. In this algorithm adaptive sliding window is parameter and assumption free in the sense that it automatically detects and adapts to the current rate of change. Its only parameter is a confidence bound. Window is not maintained explicitly but compressed using a variant of the exponential histogram technique. It keeps the window of length W using only O (log W) memory & O (log W) processing time per item, rather than the O (W) one expects from a naïve implementation. It is used as change detector since it shrinks window if and only if there has been significant change in recent examples, and estimator for the current average of the sequence it is reading since, with high probability, older parts of the window with a significantly different average are automatically dropped.

IV. EXPERIMENT AND RESULT

The proposed boosted decision trees algorithm is tested on KDDCup'99 dataset [11] and compared to that of a Naïve Bayes, kNN, eClass0 [2], eClass1 [2] and the Winner (KDDCup'99).

A. Evaluation of Anomaly Detection

There are generally two types of attacks in network intrusion detection: the attacks that involve single connections and the attacks that involve multiple connections (bursts of connections). The standard metrics in Table 1 treat all types of attacks similarly thus failing to provide sufficiently generic and systematic evaluation for the attacks that involve many network connections.

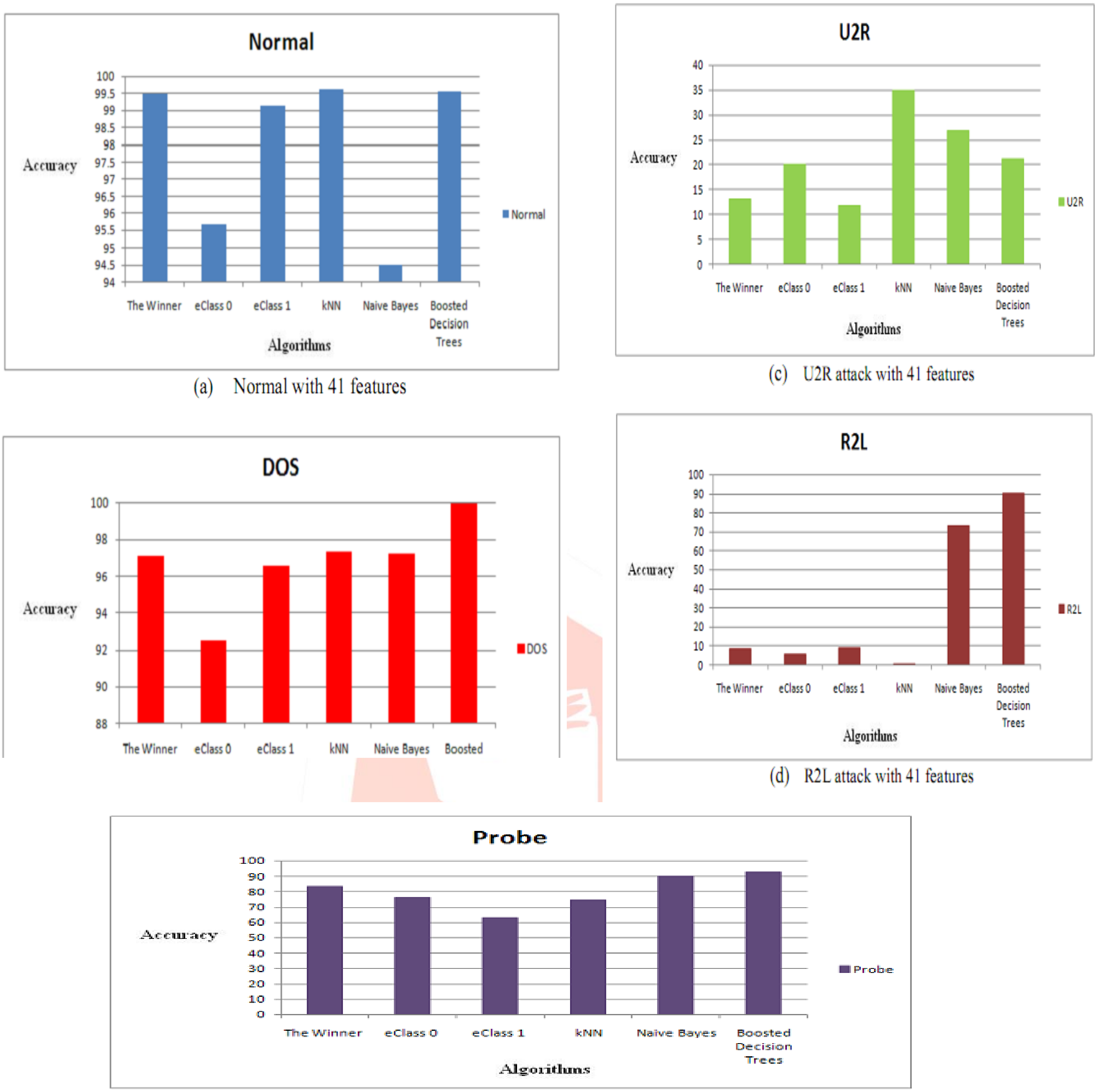
Confusion Matrix		Predicted Class	
		Normal	Intrusion Detected
Actual Class	Normal	True Negative	False Positive
	Intrusion Detected	False Negative	Correct

Interleaved Test-Then-Train - In this method each individual example can be used to test the model before it is used for training and from this the accuracy can be incrementally updated. The intension behind using this method is that, the model is always being tested on examples it has not seen. The advantage over holdout method being that holdout set is not needed for testing and ensures a smooth plot of accuracy over time as each individual example will become increasingly less significant to the overall average.

B. Evaluation on KDDCup'99 Data Set

The experiment is carried out on an intrusion detection real data stream which has been used in the Knowledge Discovery and Data Mining (KDD) 1999 Cup competition. In KDD99 dataset the input data flow contains the details of the network connections, such as protocol type, connection duration, login type etc. Each data sample in KDD99 dataset represents attribute value of a class in the network data flow, and each class is labeled either as normal or as an attack with exactly one specific attack type. In total, 41 features have been used in KDD99 dataset and each connection can be categorized into five main classes as one normal class and four main intrusion classes as DOS, U2R, R2L and Probe. There are 22 different types of attacks that are grouped into the four main types of attacks DOS, U2R, R2L and Probe tabulated in

Table 2.



(a) Normal with 41 features

(c) U2R attack with 41 features

(d) R2L attack with 41 features

(e) Probe attack with 41 features

Figures 1(a) - 1(e) show graphical comparison of boosted decision trees algorithm with SVM, kNN and Naïve Bayes with feature selection. 12 features are selected from 41 features.

The experimental setting is for the KDD99 Cup, taking 10% of the whole real raw data stream (494021 data samples) and 12 features are selected as per proposed algorithm.Figures 1(a) - 1(e) show graphical comparison of boosted decision trees algorithm with the Winner (KDDCup'99), eClass0, eClass1, kNN, C4.5 and Naïve Bayes in terms of accuracy or detection rate.

V. CONCLUSION

This paper introduced a network intrusion detection model using boosted decision trees: a learning technique that allows combining several decision trees to form a classifier which is obtained from a weighted majority vote of the classifications given by individual trees. The generalization accuracy of boosted decision trees has compared with Naïve Bayes, kNN, eClass0, eClass1 and the Winner (KDDCup'99). Boosted decision trees outperformed the compared algorithms on real world intrusion dataset, KDDCup'99. On the basis of these results, it can be concluded that boosted decision trees may be a competitive alternative to these techniques in intrusion detection system.

REFERENCES

[1] C. C. Aggrawal, P. Yu, "Outlier Detection for High Dimensional Data", Proceedings of the ACM SIGMOD Conference, 2001.

[2] P. P. Angelov, X. Zhou, “Evloving fuzzy rule based classifiers from data streams”, IEEE Transaction on Fuzzy Systems, Vol 16, No. 6, pp. 1462-1475, 2008.

[3] R. Bane, N. Shivsharan, “Network intrusion detection system (NIDS)”, pp. 1272-1277, 2008.

[4] V. Barnett, T. Lewis, “Outliers in Statistical Data”, John Wiley and Sons, NY, 1994.

[5] S. T. Brugger, “Data mining methods for network intrusion detection”, pp. 1-65, 2004.

[6] M. M. Breunig, H. P. Kriegel, R. T. Ng, J. Sander, “LOF: Identifying Density-Based Local Outliers”, Proceedings of the ACM SIGMOD Conference , 2000.

[7] D. E. Denning, “An Intrusion Detection Model”, IEEE Trans-actions on Software Engineering, SE-13, pp. 222-232, 1987.

[8] H. S. Javitz, A. Valdes, “The NIDES Statistical Component: Description and Justification”, Technical Report, Computer Science Laboratory, SRI International, 1993.

[9] E Knorr, Ng, R.: Algorithms for Mining Distance-based Outliers in Large Data Sets. Proceedings of the VLDB Conference (1998)

[10] W. Lee, S. J. Stolfo, “Data Mining Approaches for Intrusion Detection”, Proceedings of the 1998 USENIX Security Symposium, 1998.

[11] R. P. Lippmann, D. J. Fried, I. Graf, J. W. Haines, K. P. Kendall, D. McClung, D. Weber, S. E. Webster, D. Wyschogrod, R. K. Cunningham, M. A. Zissman, “Evaluating Intrusion Detection Systems: The 1998 DARPA Off-line Intrusion Detection Evaluation. Proceedings DARPA Information Survivability Conference and Exposition (DISCEX) 2000”, Vol 2, pp. 12--26, IEEE Computer Society Press, Los Alamitos, CA, 2000.

[12] W. Lee, S. J. Stolfo, “Data mining approaches for intrusion detection” Proc. of the 7th USENIX Security Symp.. San Antonio, TX, 1998.

[13] W. Lee, S. J. Stolfo, K. W. Mok, “A data mining framework for building intrusion detection models”, Proc. of the 1999 IEEE Symp.on Security and Privacy, pp. 120--132. Oakland, CA, 1999.

[14] M. Masud, J. Gao, L. Khan, J. Han, “Classifying evolving data streams for intrusion detection”.

[15] M. Panda, M. Patra, “Ensemble rule based classifiers for detecting network intrusions”, pp 19-22, 2009

[16] S. Ramaswami, R. Rastogi, K.Shim, “Efficient Algorithms for Mining Outliers from Large Data Sets”, Proceedings of the ACM SIGMOD Conference, 2000.

[17] H. Wang, W. Fan, P. Yu, J. Han, “Mining concept-drifting data streams using ensemble classifiers”, In Proceedings of the ACM SIGKDD, pp. 226-235, Washington DC, 2003.

[18] Z. Yu, J. Chen, T. Q. Zhu, “A novel adaptive intrusion detection system based on data mining”, pp.2390-2395, 2005.