Classification Techniques for Intrusion Detection – An Overview

P.Amudha
Faculty of Engineering
Avinashilingam University
Coimbatore, Tamilnadu, India

S.Karthik
SNS College of Technology
Coimbatore, Tamilnadu, India

S.Sivakumari
Faculty of Engineering
Avinashilingam University
Coimbatore, Tamilnadu, India

ABSTRACT

Security is becoming a critical part of organizational information systems and security of a computer system or network is compromised when an intrusion takes place. In the field of computer networks security, the detection of threats or attacks is nowadays a critical problem to solve. Intrusion Detection Systems (IDS) have become a standard component in network security infrastructures and is an essential mechanism to protect computer systems from many attacks. In recent years, intrusion detection using data mining have attracted researchers more and more interests. Different researchers propose a different algorithm in different categories. Classifier construction is another research challenge to build an efficient intrusion detection system. KDDCup 1999 intrusion detection dataset plays a key role in fine tuning intrusion detection system and is most widely used by the researchers working in the field of intrusion detection. This paper presents an overview of intrusion detection, KDDCup'99 dataset and detailed analysis of classification techniques used in intrusion detection.

General Terms

Network Security, Data Mining

Keywords

Intrusion Detection, classification, KDDCup

1. INTRODUCTION

With the enormous growth of computer networks usage and internet accessibility, more organizations are becoming susceptible to a wide variety of attacks and threats. The conventional intrusion prevention techniques such as firewalls, access control or encryption have failed to protect networks and systems from increasingly complicated attacks and malwares. As a result, Intrusion Detection System (IDS) proposed by Denning [9] have become an essential element of security infrastructure which is useful to detect, identify these threats and track the intruders. Since then, many research works have been focused on, how to effectively and accurately construct detection models.

As the existing intrusion detection systems require input from human which is expensive to determine effective models for normal behavior, learning algorithms can be used as an alternative to discover appropriate behavior as normal and attack. Recently, there has been an increased interest in data mining-based approaches to build intrusion detection models. Dokas et al. [11] and subsequently Wu and Yen [43] used data mining approaches for IDS in which intrusion detection was considered as a classification problem, identifying normal and other types of intrusive behavior. Hence accurate intrusion detection model can be built by choosing an effective classification approach. Most of the researchers conduct experiments on the most popular benchmark dataset, Knowledge Discovery and Data Mining – KDD'99 [23], which was developed by Massachusetts Institute of Technology (MIT) during the International Competition on data mining in 1999.

In this paper, the current status of research on classification techniques and its applications in intrusion detection system is reviewed. With a rich content of literature on this theme, the remaining of the article is organized as follows: section 2 describes intrusion detection and the data set, section 3 briefs about data mining, section 4 presents detailed insight of classification and the techniques used in intrusion detection. Finally conclusion is given in section 5.

2. INTRUSION DETECTION

An Intrusion Detection system can be defined as a combination of software and/or hardware components which monitors computer systems and makes an alarm when an intrusion occurs [6]. The basic architecture of IDS is shown in Figure 1.

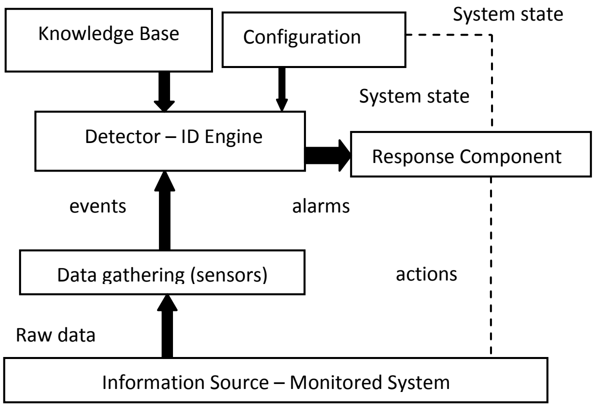

Fig 1: Basic Architecture of IDS [7]

The components in the architectural framework are:

Data Gathering Device: responsible for collecting the data from monitored system.

Detector – ID Engine: processes the data collected from sensors to identify intrusive behaviour and send an alarm signal to response component if there is an intrusion.

Knowledge Base: contains pre-processed information provided by network experts and collected by sensors.

Configuration Device: provides information about the current state of IDS.

Response Component: initiates response (active or inactive) when intrusion is detected.

An IDS is generally categorized as misuse detection and anomaly detection. The misuse detection can detect intrusions with low false alarm rate, but it fails to detect new attacks. IDS analyze the information it gathers and matches with the large databases of intrusive behaviour or attack signatures. It is also known as signature-based detection. Anomaly detection has the capability of detecting new types of attacks and is classified as static and dynamic. It determines whether deviation from the established normal usage patterns and is stated as intrusions.

2.1 Validation Parameters

The parameters that are used widely for validation of IDS are as follows:

Predictive accuracy: The two measures used for evaluating the predictive performance of IDS are: (i) detection rate and (ii) false alarm rate. Detection Rate (DR) also known as True Positive Rate (TPR) is defined as the ratio of number of attacks correctly detected to the total number of attacks, while the False Alarm (false positive) Rate (FAR) is the ratio of the number of normal connections that are incorrectly classified as attacks to the total number of normal connections. The evaluation of intrusions can be depicted using a confusion matrix which is shown in Table 1.

Table 1. Confusion Matrix

Confusion Matrix		**Predicted Class**	
		Attack	**Normal**
Actual Class	Attack	TP	FN
	Normal	FP	TN

- True Positive (TP): IDS producing an alarm when a legitimate attack occurs.
- False Positive (FP): IDS producing an alarm when no attack occurs.
- False Negative (FN): IDS producing no alarm when actual attack occurs.
- True Negative (TN): IDS producing no alarm when no attack occurs.

Receiver Operating Characteristics (ROC): Evaluation of IDS can also be performed using Receiver Operating Characteristics (ROC). ROC graphs depicts trade-offs between detection rate and false alarm rate. For visualizing the classifier performance, ROC space is used, in which FAR is represented on X-axis and DR on Y-axis. The classifier can be represented by the point in the ROC space corresponding to its (FAR, DR) pair. The resulting curve is called ROC curve as shown in Figure 2. In the graph, the point that corresponds to 0% false alarm rate and 100% detection rate represents the perfect IDS (Foster Provost, Tom Fawcett)[15].

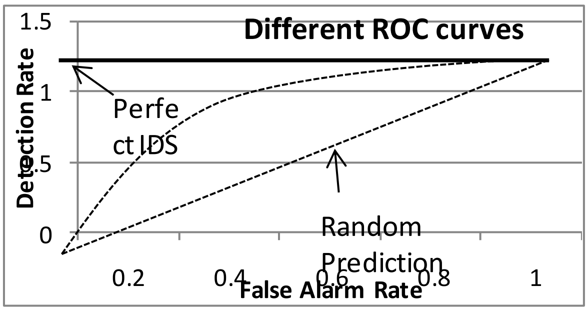

Fig 2: ROC Graph for different classifiers

Performance Time: The performance time of IDS is the total time taken by IDS to detect the intrusion.

2.2 Intrusion Detection Dataset

In this section, brief description of KDD Cup 1999 dataset [5] which was derived from the 1998 DARPA Intrusion detection Evaluation program is provided. It is the most widespread dataset collected over a period of nine weeks for a LAN simulating a typical U.S. Air Force LAN. The dataset contains a collection of simulated raw TCP dump data, where, multiple intrusions attacks was introduced and widely used in the research community. From seven weeks of network traffic, four gigabytes of compressed binary TCP dump training data was processed into five million connection records. Similarly, two weeks of test data yielded about two million connection records. The dataset contains 4,898,430 labeled and 311,029 unlabeled connection records. The labeled connection records consist of 41 attributes.

In network data of KDD99 dataset, each instance represents feature values of a class, where each class is categorized either normal or attack. The classes in dataset are characterized into one normal class and four main intrusion classes: Denial of Service (DoS), Probe, User-to-Root (U2R), Remote-to-Login (R2L).

- Normal: connections are generated by simulating user behaviour.
- DoS attacks: use of resources or services is denied to authorized users.
- Probe attacks: information about the system is exposed to unauthorized entities.
- User to Remote attacks: access to account types of administrator is gained by unauthorized entities.
- Remote to Local attacks: access to hosts is gained by unauthorized entities.

In KDD99 dataset the four attack classes (DoS, U2R, R2L, and probe) are divided into 22 different attack classes. The KDD99 intrusion detection benchmark dataset consists of three components namely: 10% KDD, Corrected KDD, Whole KDD as shown in Table 2. In the International Knowledge Discovery and Data Mining Tools Competition, only 10% KDD dataset was employed for the purpose of training where it is a more concise version of Whole KDD dataset. It contains more records of attacks than normal connections and the attack types are not distributed equally. Most of the researchers perform experiments using 10% of the overall KDD Cup'99 labeled dataset which contains 4, 94,020 records having 41 features.

Table 2. Number of attacks in training KDD99 dataset

Attacks	Dataset		
	10% KDD	Corrected	Whole KDD
Normal	97277	60593	972780
DoS	391458	229853	3883370
U2R	52	70	50
R2L	1126	11347	1126
Probe	4107	4106	41102

3. WHY DATA MINING?

Data mining is the process of discovering interesting knowledge from large amounts of data stored either in databases, data warehouses, or other information repositories [10]. Chen et al. [6]; Fayyad et al. [14] have viewed data mining as one of the step in the Knowledge Discovery Process (KDD) as shown in Figure 3.

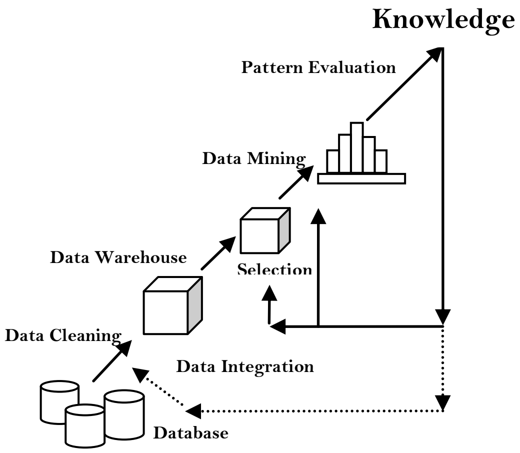

Fig 3: Steps in Knowledge discovery [10]

Frawley et al. [16] described that data mining aims to apply machine learning algorithms to large datasets. Ganti et al. [14] described algorithms that addressed three classical data mining problems: classification, clustering and association analysis which are the most important topic in data mining research and development. The main approach of data mining is classification, which maps a data item into several predefined categories. This approach outputs classifiers which has the ability to classify new data in the future.

In recent years, data mining techniques have been attracted by the researchers in the intrusion detection domain as they aim to reduce the great burden of analyzing huge volumes of audit data and producing optimization of detection rules.

4. CLASSIFICATION

Classification is a data mining technique, which arranges the data into predefined groups. The goal of predictive classification is to predict the target class accurately for each record in a set of new data, that is, data that is not in the historical data. A classification task begins with constructing data model (also known as training data) for which the target values (or class assignments) are known.

Classification or supervised learning models have been proposed by many researchers. Frawley et al. [13]; Fayyad et al. [12] defined classification as a key data mining technique whereby database tuples, acting as training samples, were analyzed in order to produce a model of the given data. In the classifier model construction, different classification algorithms use different techniques for finding relations between the predictor attributes values and the target attributes values. The classification algorithm learns from the training set and builds a predictive model to classify and distinguish between “bad” and “good” connections.

4.1 Classification Techniques for Intrusion Detection

Intrusion detection can be thought of as a classification problem where each audit record can be classified into one of a discrete set of possible categories, normal or a particular kind of intrusion. Intrusion detection using data mining have attracted more and more interests in recent years. As an important application of data mining, they aim to meliorate the great burden of analyzing huge volumes of audit data and realizing performance optimization of detection rules. Lee, W. et al. [15] described a data mining framework for adaptively building Intrusion Detection (ID) models. Data mining programs was applied to audit data to compute misuse and anomaly detection models, according to the observed behaviour in the data.

Classification techniques commonly used for classifying intrusion detection datasets are: Decision Trees, Bayesian Classification, Neural Network, Support Vector Machines, Associative Classification, k-Nearest Neighbor Classifiers, Rule Induction Methods. Most of these approaches directly apply standard methods to the publicly available intrusion detection datasets. Standard classification algorithms do not perform well when the computer intrusions are much rarer than normal behavior. In such scenarios, researchers have developed special algorithms and applied to intrusion detection problems. Moreover, the classification accuracy of the existing algorithms or techniques has to be improved as it is very difficult to detect new attacks. Classifier is a challenge to build an efficient intrusion detection system.

4.1.1 Single and Hybrid classifier approaches

4.1.1.1 Naive Bayes

Langley and Sage [25]; Domingos and Pazzani [12] found that Naïve Bayes (NB) can perform very well when moderate dependencies exist in the data. It has been shown that the performance of Naïve Bayes classifier improves when redundant features are removed. Ben Amor *et al.* [1] conducted an empirical investigation on the KDD Cup ’99 data set, comparing the performance of NB and a Decision Tree (DT). The DT obtained a higher accuracy (92.28% compared with 91.47%), but NB obtained better detection rates.

Mrutyunjaya Panda and Manas Ranjan Patra [28] proposed a framework of network intrusion detection system based on data mining algorithm, Naïve Bayes. The experiments were conducted on 10% of the KDD99 dataset and 10-fold cross validation was used for evaluation The results showed that the detection rate was 95%, with an error rate of 5%. Moreover, it performed faster (1.89 seconds) to build the model, efficient and cost effective.

Huy Anh Nguyen and Deokjai Choi [20] proposed the model for classifier algorithm selection for each attack category. They verified the effectiveness of ten distinct widely used classifier algorithms that represent a wide variety of fields: Bayesian approach, decision trees, rule based models and lazy learner for the field of intrusion detection using KDD99 dataset. They noted that no single algorithm could detect all attack categories with high detection and low false alarm rate.

Dewan Md. Farid et al. [10] introduced a new hybrid algorithm for adaptive network intrusion detection using Naive Bayesian classifier and ID3 algorithm, which analyzes the large volume of network data and considers the complex properties of attack behaviours to improve the performance of detection speed and detection accuracy. To evaluate the performance of proposed algorithm for network intrusion detection, 5-class classification was performed using 10% of KDD Cup'99 dataset. The results showed that the attacks of KDD99 dataset detected 99% accuracy using proposed algorithm.

Muda Z., et al. [29] proposed a hybrid learning approach through combination of K-Means clustering (KM) and Naive Bayes (NB) classification to improve current anomaly-based detection capabilities. The input dataset was partitioned into k-clusters according to an initial value known as the seed points into each cluster's centroids or cluster centers. The results showed that KM+NB performed better than single classifier NB in detecting normal, probe and DoS instances.

Oyebode E.O et al. [31] presented classification techniques, Naive Bayes(NB), Radial Basis function(RBF) and proposed Rotation Forest for intrusion detection to examine their accuracy in detecting network access patterns using KDDCup'99 dataset. Rotation Forest proposed by Juan J Rodriguez et al. [22] was based on rotation of feature space through Principal Component Analysis (PCA). The results showed that Rotation Forest outperformed other algorithms by yielding detection accuracy of 95.11% and 94.13% and false positive rate of 0.0489 and 0.598 for the first and second approaches respectively.

4.1.1.2 Decision trees

Quinlan [34] proposed decision tree to reduce the probability of over fitting the training data. Decision trees (DTs) are popular in misuse detection systems, as they yield good performance and offer some benefits over other machine learning techniques. Sabhnani and Serpen [35] have examined the performance of several machine learning techniques including C4.5 DT. The DT obtained good accuracy, but does not perform as well as other techniques on some classes of intrusion, particularly *U2R* and *R2L* attacks, both of which are minor classes and include a large proportion of new attack types. Similar observations have been made by Gharibian and Ghorbani [17] and demonstrated that DTs are very sensitive to the training data and do not learn well from imbalanced data. Furthermore, they found that DTs and Random Forests (ensemble of DTs) are very sensitive to the data selected for training, *i.e.*, the performance varied significantly on different folds (subsets) of the data.

DTs do suffer from the drawback of not being able to deal well with unseen data. New attacks may be classified as some default class, such as 'normal', as for the C4.5 DT employed in an investigation done by Bouzida and Cuppens [3]. Therefore, Bouzida and Cuppens developed a modified C4.5 DT, which classifies new/unseen data as a new 'unknown' class. By doing this, they avoid a significant amount of misclassifications of new attacks as normal connections, particularly U2R attacks. Ohta et al. [30] also proposed a modification to the C4.5 DT classifier, aimed at reducing the false positive rate. They changed the way in which the trees are built, by taking into account the type of errors that may be produced, choosing attributes that are less likely to produce false positives. The modified C4.5 DT outperformed the original DT and the sampling approach.

4.1.1.3 Support Vector Machine

Support vector Machine (SVM), a new promising pattern classification technique, proposed by Vapnik , is an effective classification method and supervised learning algorithms, which have been applied increasingly to misuse detection in the last decade (Burges , Cortes and Vapnik) [8]. Cortes and Vapnik defined SVM as a binary classifier algorithm that looks for an optimal hyper plane as a decision function in a high dimensional space and Figure 4 shows the SVM margins and support vectors. Furthermore, they are trained very quickly compared with MLPs. SVM and kernel methods are the popular tools for data mining tasks such as classification, regression and novelty detection (Bennett and Campbell) [2].

Srinivas Mukkamala, Guadalupa Janoski [39] proposed Neural Networks (NN) and Support Vector Machine (SVM) for intrusion detection system (IDS). The two main reasons for using SVM for intrusion detection are: speed and scalability. The experiments were carried out using DARPA 1998 dataset. SVM IDS was developed by performing training and testing on the dataset. The trained set achieved a runtime of 17.77 seconds and testing set received 99.50% accuracy with a runtime of 1.63 seconds. The performance of SVM showed that SVM IDS have slightly higher rate of making the correct detection.

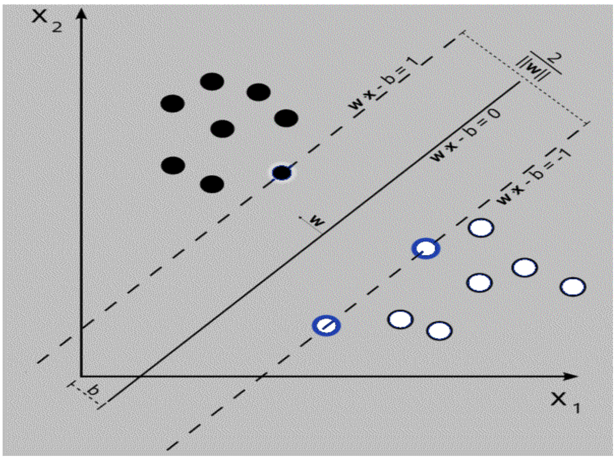

Fig 4: SVM—Margins and Support Vectors(Samples on the margin are called the support vectors)

Another modified SVM was proposed by Qing Song [33] referred to as a *Robust* SVM (RSVM), developed to better deal with noise. The RSVM was applied to host based intrusion detection by Hu *et al*. (2003) [19] analysing a subset of BSM audit data from the DARPA98 data set. On the noisy data, the SVM performed very poorly, obtaining merely 60%

true positives at 10% false positives. The RSVM, however, obtained 100% true positives at 8% false positives. Another benefit of the RSVM was that it produced less support vectors, which makes it a quicker algorithm.

Peddabachigari et al. [37] conducted an empirical investigation of SVMs and DTs, in which they analysed their performance as stand alone detectors and as hybrids. The hybrid DTSVM performed better or equally as good as the SVM alone. However, the DT performed better on *Probing*, *U2R* and *R2L*. SVM and DT-SVM performed poorly on *U2R* and *R2L* compared with the DT and ensemble.

Khan et al., (2007) [24] proposed a hybrid of SVM and clustering to shorten the training time. Khan et al. evaluated their hybrid SVM / clustering algorithm on the DARPA98 data set. Their algorithm was trained in 13.18 hours, which was approximately 5 hours shorter than a basic SVM algorithm. They also improved the accuracy, mainly due to correctly classifying more *DoS* attacks. However, the FPR increased by approximately 3%.

Su-Yun Wu, Ester Yen [40] proposed to sample different ratios of normal data to achieve better accuracy rate and to compare the efficiency of machine learning methods (decision tree and SVM) in intrusion detection system. The performance of C4.5 and SVM were compared with KDD winner. They found that C4.5 was superior to SVM in accuracy and detection; but in false alarm rate, SVM was better.

Wang Hui et al. [41] proposed an improved SVM by combining Principal Component Analysis (PCA) and Particle Swarm Optimization (PSO). PCA was an effective data mining technique and was used to reduce dimensionality of data. Then PSO was used to optimize the kernel parameters. The experimental results showed that the intrusion detection rate (97.752%) of improved SVM by combining PCA and PSO was higher than those of PSO-SVM (95.635%) and of standard SVM (90.476%).

The disadvantages of Single Classifier are:

- If the output of selected classifier is wrong the final decision may be wrong.

- The trained classifier may not be capable enough to handle the problem.

Hence combining a number of trained classifiers lead to a better performance than any single classifier.

4.1.2 Ensemble Classifier approach

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. The use of numerous data mining methods is commonly known as an ensemble approach, and the process of learning the correlation between these ensemble techniques is known by names such as multi-strategy learning, or meta-learning. Lee et al. (2000)[26] called the actual application of this learned correlation as meta-classification.

Bagging and boosting are two of the most well-known ensemble learning methods. 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. The bagging and boosting algorithms primarily operate on the data level, performing sampling. Bagging performs random sampling (with replacement) to train different classifiers, while boosting performs sampling based on a distribution that is continuously updated to increase the chances of sampling instances that are often misclassified. The bagging algorithm is given below.

```
Algorithm: Bagging.
 Input:
D, a set of d training tuples;
k, the number of models in the ensemble;
a learning scheme
Output:
 A composite model
Method:
(1) for  $i$ = 1 to $k$ do // create $k$ models:
(2) create bootstrap sample, $D_i$, by sampling $D$ with replacement;
(3) use $D_i$ to derive a model, $M_i$;
(4) end for
To use the composite model on a tuple, X:
(1) if classification then
(2) let each of the $k$ models classify $X$ and return the majority vote;
(3) if prediction then
(4) let each of the $k$ models predict a value for $X$ and return the average predicted value;
```

In boosting, weights are assigned to each training tuple. A series of k classifiers is iteratively learned. After a classifier M_i is learned, the weights are updated to allow the subsequent classifier$_{i+1}$, to pay more attention to the training tuples that were misclassified by M_i. The final boosted classifier combines the votes of each individual classifier, where the weight of each classifier's vote is a function of its accuracy [10] as shown in Figure 5.

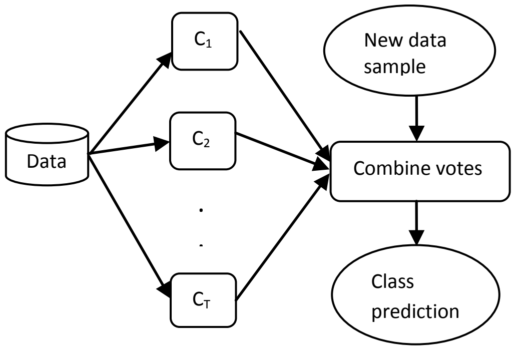

Fig 5: Combining the accuracy of claasifiers

The pros and cons of bagging and boosting methods are given in Table 4.

Table 4. Pros and cons of bagging and boosting

Technique	Pros	Cons
Bagging	Stable against noise	Needs many comparable classifiers
Boosting	Improves margins	Unstable against noise

Although the instability of DT is considered as a drawback, Breiman [4] exploited this as a beneficial trait to construct successful ensembles of DTs and Gharibian and Ghorbani [24] had shown that their performance is sensitive to the training data. Peddabachigari et al. [32] created classifier ensembles of different techniques and showed that ensemble classifiers outperformed the individual classifiers. Shelly Xiaonan Wu and Wolfgang Banzhaf [37] stated that both the hybrid and ensemble systems indicate the future trends of developing intrusion detection systems and the application of ant colony optimization to the intrusion detection domain is limited.

Breiman [42] proposed ensemble approach, Random Forest (RF) which was first applied to intrusion detection by Zhang and Zulkernine [44] to perform network based misuse and anomaly detection. On a small subset of the KDD Cup ’99 data set, the hybrid systems obtained 94.7% TPR and 2% FPR.

Weiming Hu [42] proposed an intrusion detection algorithm based on the AdaBoost algorithm. AdaBoost algorithm combines the weak classifiers (decision stumps) for continuous features and the weak classifiers for categorical features into a strong classifier. Experimental results showed that the algorithm had low computational complexity and error rates, as compared with algorithms of higher computational complexity, when tested on the benchmarked KDDCup99 dataset. The results revealed that the proposed algorithm provided false alarm rate of 0.31% - 1.79% and detection rate of 90.04% - 90.88%.

Jiong Zhang et al. [21] proposed a new systematic framework that employed the random forests algorithm for network intrusion detection. The random forests algorithm is an ensemble classification and regression approach, which is one of the most effective data mining techniques. The experiment was carried out by using the default values of the parameters for the random forests algorithm (66% samples as training data, 34% samples as test data, ten trees in the forest, and six random features to split the nodes). Results showed that overall error rate for classification (original dataset 1.92% and balanced dataset 0.05%), time to build pattern (original dataset 1975 seconds and balanced dataset 65 seconds).

Christine Dartigue et al.[7] proposed a new data-mining based technique for intrusion detection using an ensemble of binary classifiers with feature selection and multiboosting. A binary classifier for each type of attack was generated by applying different features (applying information gain and gain ratio) for different classes. Based on the accurate binary classifiers, they applied a new ensemble approach using C4.5 which aggregated each binary classifier’s decision for the same input to decide which class was most suitable for a given input. Also, they used multiboosting (wagging) for reducing bias and variance. The results showed that 98.3% normal data was correctly classified. The overall accuracy obtained was 92.30% and a cost of 0.2184.

Mrudula Gudadhe et al. [27] proposed new ensemble boosted decision tree approach for intrusion detection system. The proposed boosted decision trees algorithm was tested on 10% of KDDCup’99 dataset with 12 features and compared to that of a Naïve Bayes, k-NN, eClass0, eClass1 and the Winner (KDDCup’99) in terms of accuracy or detection rate. Boosted decision trees outperformed the compared algorithms on real world intrusion dataset, KDDCup’99 and concluded that boosted decision trees may be a competitive alternative to these techniques in intrusion detection system.

Sheng Chen et al. [38] presented ranked Minority Oversampling in Boosting (RAMOBoost) which is an integration of ensemble learning methodology with RAMO technique. RAMOBoost adjusts the sampling weights of minority class examples according to their data distribution. Moreover, RAMOBoost adopts an iterative learning procedure that assesses the hypothesis at every boosting iteration by shifting the decision boundary towards the difficult-to-learn instances of both the majority and minority classes. The results showed that oversampling ratio for minority class was increased and RAMOBoost outperformed SMOTEBoost.

Hany M. Harb, Abeer S. Desuky [18] presented a fast learning algorithm using AdaBoost ensemble with simple genetic algorithms (GA) for intrusion detection system. he strong classifiers comprises more weak classifiers requiring more time to evaluate and more memory to occupy, which affect the performance of IDS and slow down the detection process. So, to address this issue GA was proposed as a post optimization procedure for the classifiers and their coefficients, which removes the redundancy classifiers and leads to shorter final classifiers and to speed up the classification. For experimentation, randomly two separated training and testing datasets were selected from the NSL-KDD dataset. The results showed that the number of weak classifiers of the boosted strong classifiers trained by standard AdaBoost was reduced by 42% due to GA. The average classification time of the boosted strong classifier with the GA was about 60% faster where accuracy was increased by 0.64.

Taghi M. Khoshgoftaar et al. [41] presented a comprehensive empirical evaluation and comparison of boosting and bagging techniques SMOTEBoost, RUSBoost, Exactly Balanced Bagging (EBBag), Roughly Balanced Bagging (RBBag) in the context of learning from imbalanced and noisy data. There are various algorithms used for handling class imbalance including AdaCost (Fan et al. [13]), RareBoost (Joshi et al. [57]), SMOTEBoost (Chawla et al., [5]) and RUSBoost (Seiffert C et al. [36]). Among these techniques, RUBoost and SMOTEBoost performed the best in previous works and hence the authors have used for comparison. The datasets used for the experiment was from the UCI repository and the learners used were: J48, NB and RIPPER. The experiments show that the bagging techniques outperformed boosting, and hence bagging is recommended for imbalanced data.

Much research on machine learning or data mining treats the intrusion detection problem as a classification task, adopting techniques such as DTs, SVM, ANNs and Naïve Bayes (NB). Combining classifiers is a popular approach to improve the accuracy of an individual classifier. Furthermore, classifier ensembles can provide additional security from an adversary. Popular classifier combination approaches such as AdaBoost and Random Forests (RFs) have been applied to intrusion detection, as well as more specialised combinations based on observations in the literature that different classifiers perform well on different classes of intrusion. However, the results reported in the literature, which makes it difficult to determine which classifiers are indeed best suited for detecting different classes of intrusion.

5. CONCLUSION

Data mining techniques have been attracted by the researchers in the intrusion detection domain recently and they aim to reduce the great burden of analyzing huge volumes of audit data. There is an imbalance among the classes in the KDD Cup'99 data set, which has been recognized as an issue in intrusion detection that may cause poor detection of minor classes and is a major challenge to data mining. Achieving high detection rate and reducing false alarm rates are the significant challenges in designing an intrusion detection system. Using different classification techniques, it could be possible to improve the detection rate and reduce false alarm rate and need to be studied. In this paper, various classification techniques used by the researchers in evaluating the performance of intrusion detection model are reviewed. From the empirical study performed, this work identified that different researchers propose different algorithms for the intrusion detection domain in different categories, but still, it has to be explored.

REFERENCES

[1] Ben Amor, Benferhat, Elouedi, Naive Bayes vs. Decision Trees in Intrusion Detection Systems, Proc. of the 2004 ACM symposium on applied computing, 2004, pp. 420–424.

[2] Bennett K.P and Campbell C., Support Vector Machines: Hyper plane, SIGKDD Explorations, vol.2, issue 2, 2000, pp.1-13.

[3] Bouzida Y, Cuppens F, Neural networks vs. decision trees for intrusion detection, In IEEE / IST Workshop on Monitoring, Attack Detection and Mitigation, 2006.

[4] Breiman L, Random Forests, Machine Learning, vol. 45, no. 1, 2001, pp. 5-23.

[5] Chawla N.V, Bowyer K.W, Hall L.O, Kegelmeyer W.P, Smote: Synthetic minority oversampling technique, Journal of Artificial Intelligence Research, vol.16, 2002, pp.321–357.

[6] Chen M.S., Han J and Yu Philip S., Data Mining: An Overview from a Database Perspective, IEEE Transactions on Knowledge and Data Engineering, vol.8,No.6,1996,pp.866-883.

[7] Christine Dartigue, Hyun IK Jang, Wenjun Zeng, A New data-mining based approach for network Intrusion detection, Proc. of Seventh Annual Communication Networks and Services Research Conference, 2009, pp.372-377.

[8] Cortes, Vapnik, Support-vector networks, Machine Learning, vol.20, 1995, pp.273–297.

[9] Denning D. E, An intrusion-detection model, IEEE Transactions on Software Engineering, vol. SE-13, no. 2, pp.222-232.

[10] Dewan Md. Farid, Nouria Harbi, Mohammad Zahidur Rahman , Combining Naive Bayes and Decision Tree for Adaptive Intrusion Detection, Proc. of Intl. Journal of Network Security & Its Applications (IJNSA), Volume 2, Number 2, 2010, pp.12-25.

[11] Dokas, P, Ertoz, L, Lazarevic A, Srivastava J, Tan P. N ,Data mining for network intrusion detection, Proceeding of NGDM, 2002, pp.21–30.

[12] Domingos P. and Pazzani M., Beyond Independence: Conditions for the optimality of the simple Bayesian Classifier, In proceedings of the 13th Intnl. Conference on Machine Learning, 1996, pp.105-110.

[13] Fan, W., Stolfo, S., Zhang, J., & Chan, P., Adacost: Misclassification cost-sensitive boosting, ICML, 1999.

[14] Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, From Data Mining to Knowledge Discovery in Databases, American Association for Artificial Intelligence, 1996, pp.37-54.

[15] Foster Provost, Tom Fawcett, Robust Classification for Imprecise Environment, 2000, pp.1-38, Kluwer Academic Publishers.

[16] Frawley, Gregory Piatetsky-Shapiro, Christopher J Matheus, Knowledge Discovery in Databases: an Overview, AI Magazine Vol.13 No.3, 1991, pp.57-70.

[17] Gharibian F, Ghorbani A.A , Comparative Study of Supervised Machine Learning Techniques for Intrusion Detection, Proc. of the Fifth Annual Conference on Communication Networks and Services Research, 2007, pp. 350–358.

[18] Hany M. Harb, Abeer S. Desuky, AdaBoost Ensemble with Genetic Algorithm Post Optimization for Intrusion Detection, Intl. Journal of Computer Science, vol.8, issue 5, no.1, 2011, pp. 28-33.

[19] Hu W, Liao Y, Vemuri V.R , Robust Anomaly Detection Using Support Vector Machines, Proc. of Intl. Conference on Machine Learning and Applications, 2003, Morgan Kaufmann.

[20] Huy Anh Nguyen, Deokjai Choi, Application of Data Mining to Network Intrusion Detection: Classifier Selection Model, 2008, pp.399-408, Springer-Verlag.

[21] Jiong Zhang, Mohammad Zulkernine, Anwar Haque, Random-Forests-Based Network Intrusion Detection Systems, IEEE Transactions on Systems, Man, And Cybernetics—Part C: Applications and Reviews, vol. 38, no. 5, 2008, pp.649-659.

[22] Juan J Rodriguez, Ludmila I Kuncheva, Carlos J Alonso, Rotation forest: A new classifier ensemble method, IEEE Transactions on Pattern Analysis and Machine Intelligence,2006, pp. 1619-1630

[23] KDD99, KDDCup 1999 data, 1999, http://kdd.ics.uci.edu/ Databases/kddcup99/10 percent.gz.

[24] Khan L, Awad M, Thuraisingham B, A new intrusion detection system using support vector machines and hierarchical clustering, The VLDB Journal, vol.16, 2007, pp.507–521.

[25] Langley P, Sage S, Induction of selective Bayesian classifiers, Proc. of the Tenth Conference on Uncertainty in Artificial Intelligence , 1994,pp. 399-406, Seattle, WA: Morgan Kaufmann.

[26] Lee W, Stolfo S.J, Mok K.W. , A Data Mining Framework for Building Intrusion Detection Models, In

Proc of IEEE Symposium on Security and Privacy, 1999, pp.120-132.

[27] Mrudula Gudadhe, Prakash Prasad, Kapil Wankhade, A New Data Mining Based Network Intrusion Detection Model, Proc. of Intl. conference on Computer & communication technology, 2010, pp.731-735.

[28] Mrutyunjaya Panda, Manas Ranjan Patra, Network Intrusion Detection Using Naïve Bayes, International Journal of Computer Science and Network Security, vol.7 no.12, 2007, pp.258-262.

[29] Muda Z, Yassin W, Sulaiman M.N, Udzir N.I , Intrusion Detection based on k-means clustering and Naive Bayes classification, Proc. of 7th Intl. Conference on IT in Asia, 2011, pp.1-6.

[30] Ohta S, R. Kurebayashi and K. Kobayashi. , Minimizing false positives of a decision tree classifier for intrusion detection on the internet, Journal of Networks System Management, vol.16, 2008, pp.399–419. ISSN 1064-7570.

[31] Oyebode E.O, Fashoto S.G, Ojesanmi O.A, Makinde O.E, Intrusion Detection System for Computer Network Security, Australian Journal of Basic and Applied Sciences, vol.5,no,12, 2011, pp.1317-1320.

[32] Peddabachigari S, Abraham A, Grosan C, Thomas J, Modelling Intrusion Detection Systems Using Hybrid Intelligent Systems, Journal of Network and Computer Applications, vol.30, 2007, pp.114–132.

[33] Qing Song, Robust Support Vector Machine with Bullet Hole Image Classification, IEEE Transactions on Systems, Man, And Cybernetics—Part C: Applications And Reviews, Vol. 32, No. 4, 2002, pp.440-448.

[34] Quinlan, C4.5: Programs for Machine Learning, 1993, Morgan Kaufmann Publishers, San Mateo, CA.

[35] Sabhnani M, Serpen G(2003), Application of Machine Learning Algorithms to KDD Intrusion Detection Dataset within Misuse Detection Context, In Proc. of the Intl. Conference on Machine Learning, Models, Technologies and Applications, vol. 1, pp. 209–215.

[36] Seiffert C, Taghi M. Khoshgoftaar, RUSBoost: A Hybrid Approach to Alleviating Class Imbalance, IEEE Transactions on Systems, Man, And Cybernetics—Part A: Systems And Humans, Vol. 40, No. 1, 2010, pp.185-197.

[37] Shelly Xiaonan Wu, Wolfgang Banzhaf, The use of computational intelligence in intrusion detection systems: A review, Applied Soft Computing, 2010, pp.1-35, Elsevier Publication

[38] Sheng Chen, Haibo He, Edwardo A. Garcia, RAMOBoost: Ranked Minority Oversampling in Boosting, IEEE Trans. On Neural Networks, vol.21, no.10, 2010, pp.1624-1642.

[39] Srinivas Mukkamala, Guadalupe Janoski, Andrew Sung, Intrusion Detection: Support Vector Machines and Neural Networks, In Proceedings of the IEEE International Joint Conference on Neural Networks, 2002, pp. 1702-1707.

[40] Su-Yun Wu, Ester Yen, Data Mining-based intrusion detectors, Expert Systems with Applications, vol.36, 2009, pp.5605-5612, Elsevier.

[41] Taghi M. Khoshgoftaar, Jason van Hulse, Amri Napolitano, Comparing Boosting and bagging Techniques with Noisy and imbalanced data, IEEE Trans. On Systems, Man & Cybernetics-Part A: Systems and Humans, vol.41, no.3, 2011, pp.552-568.

[42] Weiming Hu, Wei Hu, Steve Maybank, AdaBoost-Based Algorithm for Network Intrusion Detection, IEEE Transactions on Systems, Man, And Cybernetics—Part B: Cybernetics, vol. 38, no. 2, 2008, pp.577-583.

[43] Wu S, Yen E, Data mining-based intrusion detectors, Expert Systems with Applications, vol.36, no.3, 2009, pp.5605–5612.

[44] Zhang J, Zulkernine M, A Hybrid Network Intrusion Detection Technique Using Random Forests, Proc. of the First International Conference on Availability, Reliability and Security, 2006, pp. 262–269.