World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:10, No:6, 2016

Evaluation of Ensemble Classifiers for Intrusion Detection

M.Govindarajan

***Abstract*—**One of the major developments in machine learning in the past decade is the ensemble method, which finds highly accurate classifier by combining many moderately accurate component classifiers. In this research work, new ensemble classification methods are proposed with homogeneous ensemble classifier using bagging and heterogeneous ensemble classifier using arcing and their performances are analyzed in terms of accuracy. A Classifier ensemble is designed using Radial Basis Function (RBF) and Support Vector Machine (SVM) as base classifiers. The feasibility and the benefits of the proposed approaches are demonstrated by the means of standard datasets of intrusion detection. The main originality of the proposed approach is based on three main parts: preprocessing phase, classification phase, and combining phase. A wide range of comparative experiments is conducted for standard datasets of intrusion detection. The performance of the proposed homogeneous and heterogeneous ensemble classifiers are compared to the performance of other standard homogeneous and heterogeneous ensemble methods. The standard homogeneous ensemble methods include Error correcting output codes, Dagging and heterogeneous ensemble methods include majority voting, stacking. The proposed ensemble methods provide significant improvement of accuracy compared to individual classifiers and the proposed bagged RBF and SVM performs significantly better than ECOC and Dagging and the proposed hybrid RBF-SVM performs significantly better than voting and stacking. Also heterogeneous models exhibit better results than homogeneous models for standard datasets of intrusion detection.

***Keywords*—**Data mining, ensemble, radial basis function, support vector machine, accuracy.

I. Introduction

TRADITIONAL protection techniques such as user authentication, data encryption, avoiding programming errors and firewalls are used as the first line of defense for computer security. If a password is weak and is compromised, user authentication cannot prevent unauthorized use; firewalls are vulnerable to errors in configuration and suspect to ambiguous or undefined security policies [1].They are generally unable to protect against malicious mobile code, insider attacks, and unsecured modems. Programming errors cannot be avoided as the complexity of the system and application software is evolving rapidly leaving behind some exploitable weaknesses. Consequently, computer systems are likely to remain unsecured for the foreseeable future. Therefore, intrusion detection is required as an additional wall for protecting systems despite the prevention techniques. Intrusion detection is useful not only in detecting successful intrusions, but also in monitoring attempts to break security, which provides important information for timely countermeasures [2], [3]. Intrusion detection is classified into two types: misuse intrusion detection and anomaly intrusion detection. Several machine-learning paradigms including neural networks [4], linear genetic programming (LGP) [5], support vector machines (SVM), Bayesian networks, multivariate adaptive regression splines (MARS) [6],fuzzy inference systems (FISs) [7], etc. have been investigated for the design of IDS.

Data mining methods may be distinguished by either supervised or unsupervised learning methods. One of the most active areas of research in supervised learning has been to study methods for constructing good ensembles of classifiers. It has been observed that when certain classifiers are ensembled, the performance of the individual classifiers.

Recently, advances in knowledge extraction techniques have made it possible to transform various kinds of raw data into high level knowledge. However, the classification results of these techniques are affected by the limitations associated with individual techniques. Hence, hybrid approach is widely recognized by the data mining research community.

Hybrid models have been suggested to overcome the defects of using a single supervised learning method, such as radial basis function and support vector machine techniques. Hybrid models combine different methods to improve classification accuracy. The term combined model is usually used to refer to a concept similar to a hybrid model. Combined models apply the same algorithm repeatedly through partitioning and weighting of a training data set. Combined models also have been called Ensembles. Ensemble improves classification performance by the combined use of two effects: reduction of errors due to bias and variance [8].

This paper proposes new ensemble classification methods to improve the classification accuracy. The main purpose of this paper is to apply homogeneous and heterogeneous ensemble classifiers for standard datasets of intrusion detection problem to improve classification accuracy. Organization of this paper is as follows. Section II describes the related work. Section III presents proposed methodology and Section IV explains the performance evaluation measures. Section V focuses on the experimental results and discussion. Finally, results are summarized and concluded in Section VI.

II. Related Work

The Internet and online procedures is an essential tool of our daily life today. They have been used as an important component of business operation [9]. Therefore, network security needs to be carefully concerned to provide secure

M. Govindarajan is with the Annamalai University, Annamalai Nagar, Tamil Nadu, India (phone: 91-4144-221946; e-mail: govind_aucse@yahoo.com)

information channels. Intrusion detection (ID) is a major research problem in network security, where the concept of ID was proposed by Anderson in 1980 [10]. ID is based on the assumption that the behavior of intruders is different from a legal user [11]. The goal of intrusion detection systems (IDS) is to identify unusual access or attacks to secure internal networks [12] Network-based IDS is a valuable tool for the defense-in-depth of computer networks. It looks for known or potential malicious activities in network traffic and raises an alarm whenever a suspicious activity is detected. In general, IDSs can be divided into two techniques: misuse detection and anomaly detection [13], [14].

Misuse intrusion detection (signature-based detection) uses well-defined patterns of the malicious activity to identify intrusions [15], [16]. However, it may not be able to alert the system administrator in case of a new attack. Anomaly detection attempts to model normal behavior profile. It identifies malicious traffic based on the deviations from the normal patterns, where the normal patterns are constructed from the statistical measures of the system features [17]. The anomaly detection techniques have the advantage of detecting unknown attacks over the misuse detection technique [18]. Several machine learning techniques including neural networks, fuzzy logic [19], support vector machines (SVM) [17], [19] have been studied for the design of IDS. In particular, these techniques are developed as classifiers, which are used to classify whether the incoming network traffics are normal or an attack. This paper focuses on the Support Vector Machine (SVM) and Radial Basis Function (RBF) among various machine learning algorithms.

The most significant reason for the choice of SVM is because it can be used for either supervised or unsupervised learning. Another positive aspect of SVM is that it is useful for finding a global minimum of the actual risk using structural risk minimization, since it can generalize well with kernel tricks even in high- dimensional spaces under little training sample conditions.

In [20], it is shown how neural networks can be employed for the anomaly and misuse detection. The works present an application of neural network to learn previous behavior since it can be utilized to detection of the future intrusions against systems. Experimental results indicate that neural networks are “suited to perform intrusion state of art detection and can generalize from previously observed behavior” according to the authors.

Reference [21] suggested Application of SVM an ANN for intrusion detection. Reference [22] used flexible neural network trees for feature deduction and intrusion detection. Reference [23] combined multiple techniques for intrusion detection.

References [24], [25] proposed an algorithm the basis of which is to adaptively resample and **c**ombine (hence the acronym--arcing) so that the weights in the resampling are increased for those cases most often misclassified and the combining is done by weighted voting.

Previous work has demonstrated that arcing classifiers is very effective for RBF-SVM hybrid system [26]. A hybrid model can improve the performance of basic classifier [12].

In this paper, a hybrid intrusion detection system is proposed using radial basis function and support vector machine and the effectiveness of the proposed bagged RBF, bagged SVM and RBF-SVM hybrid system is evaluated by conducting several experiments on real and benchmark datasets of intrusion detection. The performance of the proposed bagged RBF, bagged SVM, and RBF-SVM hybrid classifiers are examined in comparison with standalone RBF and standalone SVM classifier and also heterogeneous models exhibits better results than homogeneous models for standard data sets of intrusion detection.

III. Proposed Methodology

A. Preprocessing

Before performing any classification method, the data has to be preprocessed. In the data preprocessing stage, it has been observed that the datasets consist of many missing value attributes. By eliminating the missing attribute, records may lead to misclassification because the dropped records may contain some useful pattern for Classification. The dataset is preprocessed by removing missing values using supervised filters.

B. Existing Classification Methods

1) Radial Basis Function Neural Network

Radial basis function (RBF) networks [27] combine a number of different concepts from approximation theory, clustering, and neural network theory. A key advantage of RBF networks for practitioners is the clear and understandable interpretation of the functionality of basic functions. In addition, fuzzy rules may be extracted from RBF networks for deployment in an expert system.

The RBF networks used here may be defined as follows.

1. RBF networks have three layers of nodes: input layer uI, hidden layer u_H and output layer u_0
2. Feed-forward connections exist between input and hidden layers, between input and output layers (shortcut connections), and between hidden and output layers. Additionally, there are connections between a bias node and each output node. A scalar weight wi, j is associated with the connection between nodes i and j.
3. The activation of each input node (fanout) $i \in u_I$ is equal to its external input

$$ai(k) \overset{def}{=} xi(k) \tag{1}$$

where $x_i(k)$ is the element of the external input vector (pattern) $X(k)$ of the network($k = 1,2,....$ denotes the number of the pattern).

4. Each hidden node (neuron) $j \in u_H$ determines the Euclidean distance between "its own" weight vector $\mathrm{W}j \overset{\text{def}}{=} (\mathrm{w}_{(1,j)},.....,\mathrm{w}_{(|u_I|,j)})^T$ and the activations of the input nodes, i.e., the external input vector

$$s_j(k) \overset{def}{=} \|Wj - X(k)\| \tag{2}$$

The distance $\mathrm{s}_j(\mathrm{k})$ is used as an input of a radial basis function in order to determine the activation $\mathrm{a}_j(\mathrm{k})$ of node j. Here, Gaussian functions are employed

$$a_j(k) \overset{def}{=} e^{(-s_j(k)^2 / r_j^2)} \tag{3}$$

The parameter r_j of node j is the radius of the basis function; the vector w_j is its center.

Localized basis functions such as the *Gaussian* or the *inverse multiquadric*are usually preferred.

5. Each output node (neuron) $l \in u_0$ computes its activation as a weighted sum

$$a_l(k) \overset{def}{=} \sum_{j=1}^{|u_H|} w(j,l).a_j(k) + \sum_{i=1}^{|u_I|} w(i,l).a_i(k) + w(B,l) \tag{4}$$

The external output vector of the network, $y(k)$ consists of the activations of output nodes, i.e., $y_l(k) \overset{def}{=} a_l(k)$. The activation of a hidden node is high if the current input vector of the network is "similar" (depending on the value of the radius) to the center of its basis function. The center of a basis function can, therefore, be regarded as a prototype of a hyperspherical cluster in the input space of the network. The radius of the cluster is given by the value of the radius parameter. In the literature, some variants of this network structure can be found, some of which do not contain shortcut connections or bias neurons.

2) Support Vector Machine

Support vector machines [28], [29] are powerful tools for data classification. Classification is achieved by a linear or nonlinear separating surface in the input space of the dataset. The separating surface depends only on a subset of the original data. This subset of data, which is all that is needed to generate the separating surface, constitutes the set of support vectors. In this study, a method is given for selecting as small a set of support vectors as possible which completely determines a separating plane classifier. In nonlinear classification problems, SVM tries to place a linear boundary between two different classes and adjust it in such a way that the margin is maximized [30]. Moreover, in the case of linearly separable data, the method is to find the most suitable one among the hyperplanes that minimize the training error. After that, the boundary is adjusted such that the distance between the boundary and the nearest data points in each class is maximal.

In a binary classification problem, its data points are given as:

$$D = \{(x^1, y^1),....(x^l, y^l)\},....x \in \Re^n, y \in \{-1,1\}, \tag{5}$$

Where y = a binary value representing the two classes and, x = the input vector.

As mentioned above, there are numbers of hyperplanes that can separate these two sets of data and the problem is to find the hyperplane with the largest margin. Suppose that all training data satisfy the following constraints:

$$\text{For } y_i = +1 \quad w \,.\, x \;+\; b \;\geq\; +1 \tag{6}$$

$$\text{For } y_i = -1 \; w.x + b \leq -1 \tag{7}$$

Where w = the boundary, x = the input vector, b = the scalar threshold (bias).Therefore, the decision function that can classify the data is:

$$f(y) = \mathrm{sgn}((w.x) + b) \tag{8}$$

Thus, the separating hyperplane must satisfy the following constraints:

$$y_i[(w.x_i) + b] \geq 1 \tag{9}$$

Where *l* = the number of training sets

The optimal hyperplane is the unique one that not only separates the data without error but also maximizes the margin. It means that it should maximize the distance between closest vectors in both classes to the hyperplane. Therefore, the hyperplane that optimally separate the data into two classes can be shown to be the one that minimize the functional:

$$\phi(w) = \frac{|w|^2}{2} \tag{10}$$

Therefore, the optimization problem can be formulated into an equivalent non-constraint optimization problem by introducing the Lagrange multipliers ($\alpha_I >= 0$) and a Lagrangian:

$$I(w,b,\alpha) = \frac{1}{2}\|w\|^2 - \sum_{t=1.l} \alpha_t(y_t((w.x_t)+b)-1) \tag{11}$$

The Lagrangian has to be minimized with respect to **w** and *b* by the given expressions:

$$w_0 = \sum y\alpha x \tag{12}$$

This expressions for w_0 is then substitute into (11) which will result in dual form of the function which has to be maximized with respect to the constraints $\alpha_I > 0$.

$$\text{Maximize } W(\alpha) = \sum \alpha_I - \frac{1}{2} \sum_{I,j=1..l} \alpha_i \alpha_j y_i y_j (x_i x_j) \quad (13)$$

Subject to $\alpha_I \geq 0, i = 1..l$ and $\sum \alpha_I y_i$. The hyperplane decision function can therefore be written as:

$$f(x) = sign(w_0 x + b_0) = sign(\sum y_i \alpha_i^0 (x_i . x) + b_0) \quad (14)$$

However, (14) is meant for linearly separable data in SVM. In a non-linearly separable data, SVM is used to learn the decision functions by first mapping the data to some higher dimensional feature space and constructing a separating hyperplane in this space.

C. Homogeneous Ensemble Classifiers

1) Dagging

This meta classifier creates a number of disjoint, stratified folds out of the data and feeds each chunk of data to a copy of the supplied base classifier. Predictions are made via majority vote, since all the generated base classifiers are put into the Vote meta classifier. It is useful for base classifiers that are quadratic or worse in time behavior, regarding number of instances in the training data.

2) ECOC

Error correcting output codes (ECOC) are commonly used in information theory for correcting bit reversals caused by noisy communication channels, or in machine learning for converting binary classifiers, such as support vector machines, to multi-class classifiers by decomposing a multi-class problem into several two-class problems [31].Dietterich and Bakiri introduced ECOC to be used within the ensemble setting [32]. The idea is to use a different class encoding for each member of the ensemble.

3) Proposed Bagged RBF and SVM Classifiers

Given a set D, of d tuples, bagging [33] works as follows. For iteration i (i =1, 2,…k), a training set, D_i, of d tuples is sampled with replacement from the original set of tuples, D. The bootstrap sample D_i, by sampling D with replacement, from the given training data set D repeatedly. Each example in the given training set D may appear repeated number of times or not at all in any particular replicate training data set D_i. A classifier model, M_i, is learned for each training set, D_i. To classify an unknown tuple, X, each classifier, M_i, returns its class prediction, which counts as one vote. The bagged RBF and SVM, M^*, counts the votes and assigns the class with the most votes to X.

Algorithm: RBF and SVM ensemble classifiers using bagging
Input:

- D, a set of d tuples.
- k = 2, the number of models in the ensemble.
- Base Classifiers (Radial Basis Function, Support Vector Machine)

Output: Bagged RBF and SVM, M^*
Method:

1. for i= 1 to k do // create k models
2. Create a bootstrap sample, D_i, by sampling D with replacement, from the given training data set D repeatedly. Each example in the given training set D may appear repeated times or not at all in any particular replicate training data set D_i
3. Use D_i to derive a model, M_i;
4. Classify each example din training data $D_i$and initialized the weight, $W_i$for the model, M_i, based on the accuracies of percentage of correctly classified example in training data D_i.
5. endfor

To use the bagged RBF and SVM models 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;

D. Heterogeneous Ensemble Classifiers

1) Weighted Majority Algorithm

In machine learning, Weighted Majority Algorithm (WMA) is a meta-learning algorithm used to construct a compound algorithm from a pool of prediction algorithms, which could be any type of learning algorithms, classifiers, or even real human experts. The algorithm assumes that we have no prior knowledge about the accuracy of the algorithms in the pool, but there are sufficient reasons to believe that one or more will perform well.

Assume that the problem is a binary decision problem. To construct the compound algorithm, a positive weight is given to each of the algorithms in the pool. The compound algorithm then collects weighted votes from all the algorithms in the pool, and gives the prediction that has a higher vote. If the compound algorithm makes a mistake, the algorithms in the pool that contributed to the wrong predicting will be discounted by a certain ratio β where $0<\beta<1$.

It can be shown that the upper bounds on the number of mistakes made in a given sequence of predictions from a pool of algorithms A is

$$O(log|A| + m) \quad (15)$$

if one algorithm in X_i makes at most $m_{mistakes}$. There are many variations of the Weighted Majority Algorithm to handle different situations, like shifting targets, infinite pools, or randomized predictions. The core mechanisms remain similar, with the final performances of the compound algorithm bounded by a function of the performance of the specialist (best performing algorithm) in the pool.

2) Stacking

Stacking (sometimes called *stacked generalization*) involves training a learning algorithm to combine the

predictions of several other learning algorithms. First, all of the other algorithms are trained using the available data, then a combiner algorithm is trained to make a final prediction using all the predictions of the other algorithms as additional inputs. If an arbitrary combiner algorithm is used, then stacking can theoretically represent any of the ensemble techniques described in this article, although in practice, a single-layer logistic regression model is often used as the combiner.

Stacking typically yields performance better than any single one of the trained models. It has been successfully used on both supervised learning tasks (regression) and unsupervised learning (density estimation). It has also been used to estimate bagging's error rate. It has been reported to out-perform Bayesian model-averaging. The two top-performers in the Netflix competition utilized *blending*, which may be considered to be a form of stacking.

3) Proposed RBF-SVM Hybrid System

Given a set D, of d tuples, arcing [34] works as follows; For iteration i (i =1, 2,…..k), a training set, D_i, of d tuples is sampled with replacement from the original set of tuples, D. some of the examples from the dataset *D* will occur more than once in the training dataset D_i. The examples that did not make it into the training dataset end up forming the test dataset. Then a classifier model, M_i, is learned for each training examples *d* from training dataset D_i. A classifier model, M_i, is learned for each training set, D_i. To classify an unknown tuple, X, each classifier, M_i, returns its class prediction, which counts as one vote. The hybrid classifier (RBF-SVM), M^*, counts the votes and assigns the class with the most votes to X.

Algorithm: Hybrid RBF-SVM using Arcing Classifier

Input:

- D, a set of d tuples.
- $k = 2$, the number of models in the ensemble.
- Base Classifiers (Radial Basis Function, Support Vector Machine)

Output: Hybrid RBF-SVM model, M^*.

Procedure:

1. For i = 1 to k do // Create k models
2. Create a new training dataset, D_i, by sampling D with replacement. Same example from given dataset *D* may occur more than once in the training dataset D_i.
3. Use D_i to derive a model, M_i
4. Classify each example din training data D_i and initialized the weight, $W_i$for the model, M_i, based on the accuracies of percentage of correctly classified example in training data D_i.
5. endfor

To use the hybrid 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;

The basic idea in Arcing is like bagging, but some of the original tuples of D may not be included in Di, whereas others may occur more than once.

IV. Performance Evaluation Measures

A. Cross Validation Technique

Cross-validation [35] sometimes called rotation estimation, is a technique for assessing how the results of a statistical analysis will generalize to an independent data set. It is mainly used in settings where the goal is prediction, and one wants to estimate how accurately a predictive model will perform in practice. 10-fold cross validation is commonly used. In stratified K-fold cross-validation, the folds are selected so that the mean response value is approximately equal in all the folds.

B. Criteria for Evaluation

The primary metric for evaluating classifier performance is classification Accuracy: the percentage of test samples that the ability of a given classifier to correctly predict the label of new or previously unseen data (i.e. tuples without class label information). Similarly, the accuracy of a predictor refers to how well a given predictor can guess the value of the predicted attribute for new or previously unseen data.

V. Experimental Results and Discussion

A. Acer07 Dataset Description

The Acer07 dataset, being released for the first time is a real world data set collected from one of the sensors in Acer eDC (Acer e-Enabling Data Center). The data used for evaluation is the inside packets from August 31, 2007 to September 7, 2007.

B. NSL-KDD Dataset Description

The data used in classification is NSL-KDD, which is a new dataset for the evaluation of researches in network intrusion detection system. NSL-KDD consists of selected records of the complete KDD'99 dataset [36].NSL-KDD dataset solve the issues of KDD'99 benchmark [37]. Each NSL-KDD connection record contains 41 features (e.g., protocol type, service, and flag, etc.) and is labeled as either normal or an attack, with one specific attack type.

C. Experiments and Analysis

In this section, new ensemble classification methods are proposed using classifiers in both homogeneous ensembles using bagging and heterogeneous ensembles using arcing classifier and their performances are analyzed in terms of accuracy. The performance of the proposed homogeneous and heterogeneous ensemble classifiers are compared to the performance of other standard homogeneous and heterogeneous ensemble methods.

1) Homogeneous Ensemble Classifiers

The NSL-KDD and Acer07 datasets are taken to evaluate the base classifiers and homogeneous ensemble classifiers.

TABLE I
THE PERFORMANCE OF BASE CLASSIFIERS AND HOMOGENEOUS ENSEMBLE CLASSIFIERS FOR NSL-KDD DATASET

Datasets	Classifiers	Classification Accuracy
NSL- KDD	RBF	84.74 %
	Proposed Bagged RBF	86.40 %
	ECOC RBF	86.07 %
	Dagged RBF	85.36 %
	SVM	91.81 %
	Proposed Bagged SVM	93.92 %
	ECOC SVM	91.93 %
	Dagged SVM	90.89 %

TABLE II
THE PERFORMANCE OF BASE CLASSIFIERS AND HOMOGENEOUS ENSEMBLE CLASSIFIERS FOR ACER07DATASET

Datasets	**Classifiers**	**Classification Accuracy**
Acer07	RBF	99.53%
	Proposed Bagged RBF	99.86 %
	ECOC RBF	99.40 %
	Dagged RBF	98.10 %
	SVM	99.80%
	Proposed Bagged SVM	99.93 %
	ECOC SVM	99.90 %
	Dagged SVM	97.90 %

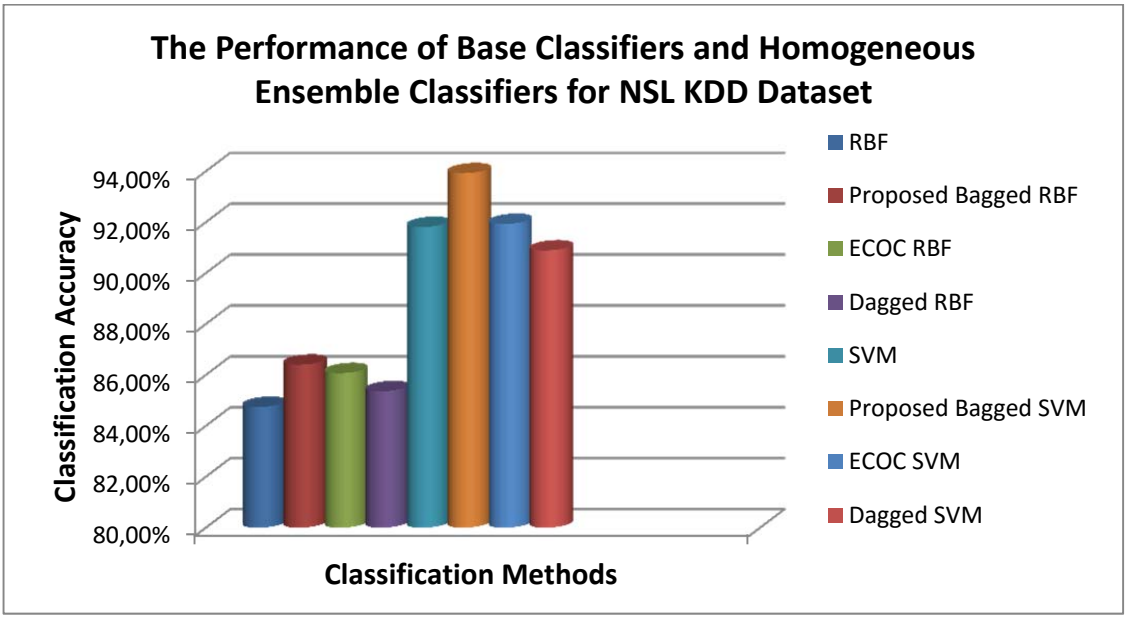

Fig.1 Classification Accuracy of base classifiers and Homogeneous Ensemble Classifiers using NSL- KDD dataset

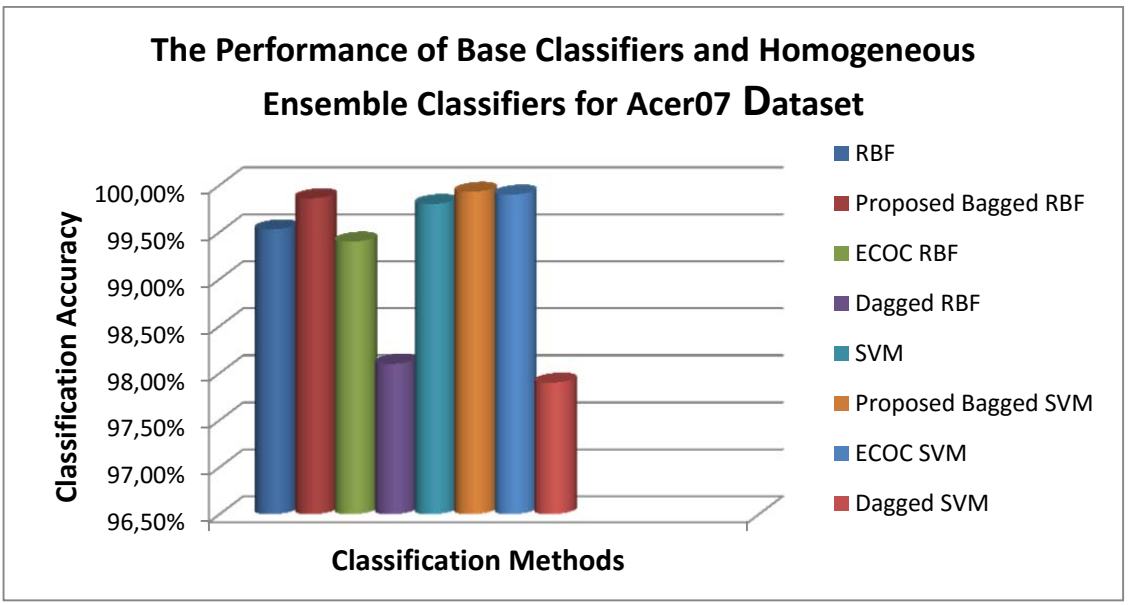

Fig. 2 Classification Accuracy of base classifiers and homogeneous ensemble classifiers using Acer07 dataset

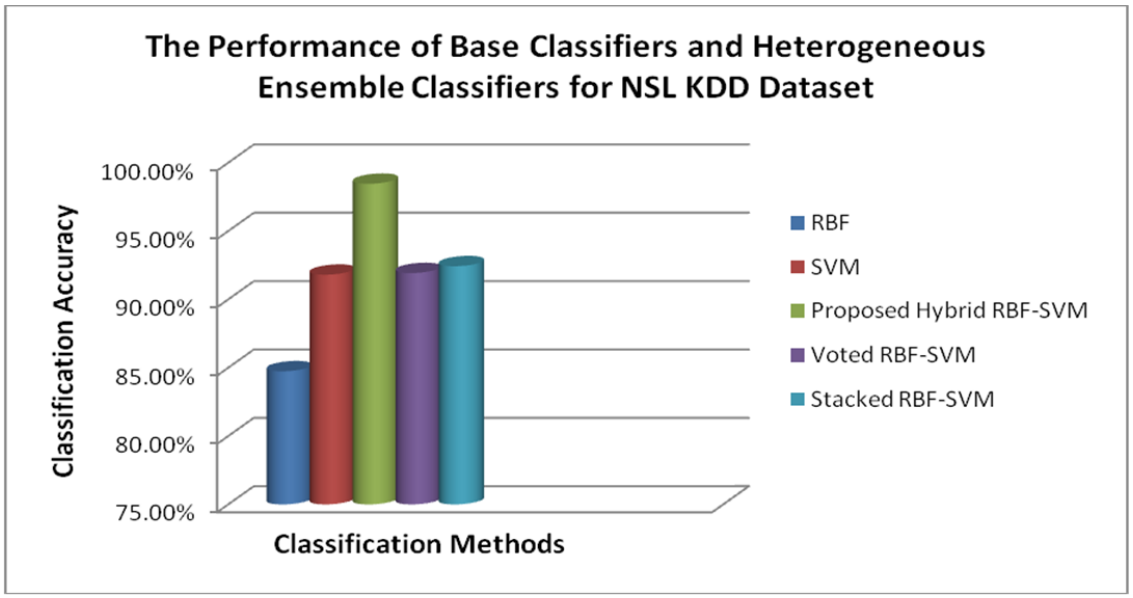

Fig.3 Classification Accuracy of Base and Heterogeneous Ensemble Classifiers using NSL-KDD dataset

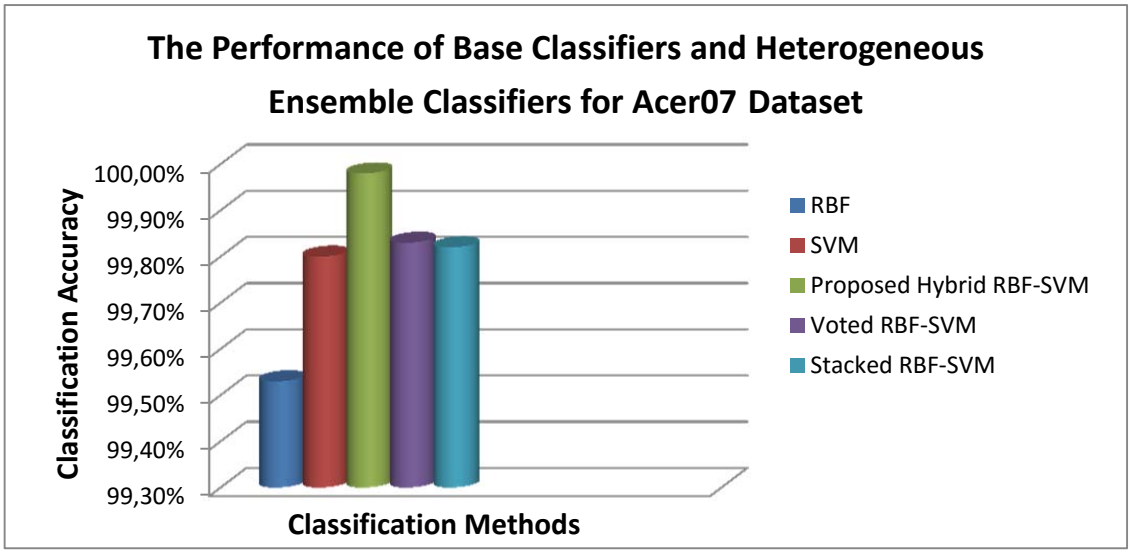

Fig. 4 Classification Accuracy of Base and Heterogeneous Ensemble Classifiers Using Acer07 dataset

2) Heterogeneous Ensemble Classifiers

The NSL-KDD and Acer07 datasets are taken to evaluate the base and heterogeneous ensemble classifiers.

TABLE III
THE PERFORMANCE OF BASE AND HETEROGENEOUS ENSEMBLE CLASSIFIERS FOR NSL- KDD DATASET

Dataset	Classifiers	Classification Accuracy
NSL- KDD	RBF	84.74 %
	SVM	91.81 %
	Proposed Hybrid RBF-SVM	98.46 %
	Voted RBF-SVM	91.93 %
	Stacked RBF-SVM	92.43 %

TABLE IV
THE PERFORMANCE OF BASE AND HETEROGENEOUS ENSEMBLE CLASSIFIERS FOR ACER07DATASET

Dataset	Classifiers	Classification Accuracy
Acer07	RBF	99.53 %
	SVM	99.80 %
	Proposed Hybrid RBF-SVM	99.98 %
	Voted RBF-SVM	99.83 %
	Stacked RBF-SVM	99.82 %

D.Experimental Comparison

In all experiments presented here, classification accuracy is estimated using 10-fold stratified cross validation [38]. Cross validation is repeated ten times using different random generator seeds resulting in ten different sets of folds. The same folds (random generator seeds) were used in all experiments. On the meta-level, the performances of four

algorithms for combining classifiers are compared. The four algorithms used for combining classifiers are ECOC, Dagging, majority voting and stacking.

The classification accuracy of the combining algorithms averaged over ten runs of ten-fold cross validation. Assessment of performance is based on the calculation of the χ^2 statistic for all the approaches and their critical values are found to be less than 0.455. Hence, their corresponding probability is $p < 0.5$. This is smaller than the conventionally accepted significance level of 0.05 or 5%. Thus examining a χ^2 significance table, it is found that this value is significant with a degree of freedom of 1. In general, the result of χ^2 statistic analysis shows that the proposed classifiers are significant at $p < 0.05$ than the existing classifiers.

1) Homogeneous Ensemble Classifiers

In this research work, new ensemble classification methods are proposed with homogeneous ensembles using bagging and their performances are analyzed in terms of accuracy. Here, the base classifiers are constructed using radial basis function and Support Vector Machine. Bagging is performed with radial basis function classifier and support vector machine to obtain a very good classification performance. Tables I and II show classification performance for standard datasets of intrusion detection using existing and proposed bagged radial basis function neural network and support vector machine. The analysis of results shows that the proposed bagged radial basis function and bagged support vector machine classifies are shown to be superior to individual approaches for standard datasets of intrusion detection problem in terms of classification accuracy. According to Figs.1 and 2 proposed combined models show significantly larger improvement of classification accuracy than the base classifiers and the results are found to be statistically significant. This means that the combined methods are more accurate than the individual methods in the field of intrusion detection.

Tables I and II compare the performance of proposed bagged RBF and SVM to the performance of ECOC and Dagging with RBF and SVM. The proposed bagged RBF and SVM performs significantly better than ECOC and Dagging on standard datasets of intrusion detection.

2) Heterogeneous Ensemble Classifiers

In this research work, new hybrid classification methods are proposed with heterogeneous ensembles using arcing classifier and their performances are analyzed in terms of accuracy. The data set described in Section V is being used to test the performance of base classifiers and hybrid classifier. In the proposed approach, first the base classifiers RBF and SVM are constructed individually to obtain a very good generalization performance. Secondly, the ensemble of RBF and SVM is designed. In the ensemble approach, the final output is decided as follows: base classifier's output is given a weight (0–1 scale) depending on the generalization performance as given in Tables III and IV. According to Figs.3 and 4, the proposed hybrid models show significantly larger improvement of classification accuracy than the base classifiers and the results are found to be statistically significant. The experimental results show that proposed hybrid RBF-SVM is superior to individual approaches for intrusion detection problem in terms of classification accuracy.

The performance comparison between proposed hybrid RBF-SVM and voting, stacking with RBF and SVM can be found in Tables III and IV. Both methods use the same base classifiers. The proposed hybrid RBF-SVM performs significantly better on standard datasets of intrusion detection. The overall relative improvement of accuracy is high.

VI. Conclusion

In this research work, new combined classification methods are proposed using classifiers in homogeneous ensembles using bagging and the performance comparisons have been demonstrated using standard datasets of intrusion detection in terms of accuracy. Here, the proposed bagged radial basis function and bagged support vector machine combines the complementary features of the base classifiers. Similarly, new hybrid RBF-SVM models are designed in heterogeneous ensembles involving RBF and SVM models as base classifiers and their performances are analyzed in terms of accuracy. The performance of the proposed homogeneous and heterogeneous ensemble classifiers are compared to the performance of other standard homogeneous and heterogeneous ensemble methods. The standard homogeneous ensemble methods include Error correcting output codes, dagging and heterogeneous ensemble methods include majority voting, stacking.

The experiment results lead to the following observations.

- SVM exhibits better performance than RBF in the important respects of accuracy.
- The proposed bagged methods are shown to be significantly higher improvement of classification accuracy than the base classifiers.
- The hybrid RBF-SVM shows higher percentage of classification accuracy than the base classifiers.
- The proposed ensemble methods provide significant improvement of accuracy compared to individual classifiers and the proposed bagged RBF and SVM performs significantly better than ECOC and Dagging and the proposed hybrid RBF-SVM performs significantly better than voting and stacking.
- The heterogeneous models exhibit better results than homogeneous models for standard datasets of intrusion detection.

The future research will be directed towards developing more accurate base classifiers particularly for the intrusion detection problem.

Acknowledgment

Author gratefully acknowledges the authorities of Annamalai University for the facilities offered and encouragement to carry out this work.

REFERENCES

[1] Summers RC, *Secure computing: threats and safeguards*, New York: McGraw-Hill, 1997.

[2] Heady R, Luger G, Maccabe A, Servilla M, "The architecture of a network level intrusion detection system", *Technical Report*, Department of Computer Science, University of New Mexico, 1990.

[3] Sundaram A, *An introduction to intrusion detection,* ACM Cross Roads, 2(4), 1996.

[4] Mukkamala S, Sung AH, Abraham A, "Intrusion detection using ensemble of soft computing paradigms", *proceedings of the third international conference on intelligent systems design and applications*, intelligent systems design and applications, advances in soft computing. Germany, Springer, 2003,pp. 239–48.

[5] Mukkamala S, Sung AH, Abraham A, "Modeling intrusion detection systems using linear genetic programming approach", *proceedings of the 17th international conference on industrial & engineering applications of artificial intelligence and expert systems*, innovations in applied artificial intelligence. In: Robert O., Chunsheng Y., Moonis A., editors. Lecture Notes in Computer Science, vol. 3029, Germany: Springer, 2004a, dpp. 633–42.

[6] Mukkamala S, Sung AH, Abraham A, Ramos V, "Intrusion detection systems using adaptive regression splines", Seruca I, Filipe J, Hammoudi S, Cordeiro J, editors. *Proceedings of the 6th international conference on enterprise information systems, ICEIS'04*, vol. 3, Portugal, 2004b, pp. 26–33.

[7] Shah K, Dave N, Chavan S, Mukherjee S, Abraham A, Sanyal S, "Adaptive neuro-fuzzy intrusion detection system", *IEEE International Conference on Information Technology: Coding and Computing (ITCC'04),* vol. 1. USA: IEEE Computer Society, 2004, pp. 70–74.

[8] Haykin, S, *Neural networks: a comprehensive foundation(second ed.),* New Jersey: Prentice Hall, 1999.

[9] T. Shon and J. Moon, "A hybrid machine learning approach to network anomaly detection", *Information Sciences*, vol.177, 2007, pp. 3799-3821.

[10] P. Anderson, "Computer security threat monitoring and surveillance", *Technical Report*, James P. Anderson Co., Fort Washington, PA, 1980.

[11] W. Stallings, *Cryptography and network security principles and practices*, USA: Prentice Hall, 2006.

[12] C. Tsai , Y. Hsu, C. Lin and W. Lin, "Intrusion detection by machine learning: A review", *Expert Systems with Applications*, vol. 36, 2009, pp.11994-12000.

[13] E. Biermann, E. Cloete and L.M. Venter, "A comparison of intrusion detection Systems", *Computer and Security*, vol. 20, 2001, pp. 676-683.

[14] T. Verwoerd and R. Hunt, "Intrusion detection techniques and approaches", *Computer Communications*, vol. 25, 2002, pp.1356-1365.

[15] K. Ilgun, R.A. Kemmerer and P.A. Porras, "State transition analysis:A rule-based intrusion detection approach" , *IEEE Trans. Software Eng.* vol. 21, 1995, pp. 181-199.

[16] D. Marchette, "A statistical method for profiling network traffic", proceedings *of the First USENIX Workshop on Intrusion Detection and Network Monitoring (Santa Clara),* CA, 1999, pp. 119-128.

[17] S. Mukkamala, G. Janoski and A.Sung, "Intrusion detection: support vector machines and neural networks" *proceedings of the IEEE International Joint Conference on Neural Networks (ANNIE)*, St. Louis, MO, 2002, pp. 1702-1707.

[18] E. Lundin and E. Jonsson, "Anomaly-based intrusion detection: privacy concerns and other problems", *Computer Networks,* vol. 34, 2002, pp. 623-640.

[19] S. Wu and W. Banzhaf, "The use of computational intelligence in intrusion detection systems: A review", *Applied Soft Computing*, vol.10, 2010, pp. 1-35.

[20] Ghosh AK, Schwartzbard A, "A study in using neural networks for anomaly and misuse detection", *proceeding on the 8th USENIX security symposium*, http://citeseer.ist.psu.edu/context/1170861/0, 1999. (accessed August 2006).

[21] W. H. Chen, S. H. Hsu, H.P Shen, "Application of SVM and ANN forintrusion detection", *ComputOperRes,* 32(10), 2005a, pp. 2617–2634.

[22] Chen Y, Abraham A, and Yang J, "Feature deduction and intrusion detection using flexible neural trees", *proceedings of Second IEEE International Symposium on Neural Networks*, 2005b,pp. 2617-2634.

[23] C. Katar, "Combining multiple techniques for intrusion detection", *Int J ComputSci Network Security*, 2006, pp. 208–218.

[24] Freund, Y. and Schapire, R, "A decision-theoretic generalization of on-line learning and an application to boosting", *proceedings of the Second European Conference on Computational Learning Theory*, 1995, pp. 23-37.

[25] Freund, Y. and Schapire, R, "Experiments with a new boosting algorithm", *Proceedings of the Thirteenth International Conference on Machine Learning*, 1996, pp.148-156 Bari, Italy.

[26] M.Govindarajan, RM.Chandrasekaran, "Intrusion Detection using an Ensemble of Classification Methods", *Proceedings of International Conference on Machine Learning and Data Analysis*, San Francisco, U.S.A, 2012, pp. 459-464.

[27] Oliver Buchtala, Manuel Klimek, and Bernhard Sick, Member, IEEE,"Evolutionary Optimization of Radial Basis Function Classifiers for Data Mining Applications", *IEEE Transactions on systems, man, and cybernetics—part b: cybernetics*, 35(5), 2005.

[28] Cherkassky, V. and Mulier, F, *Learning from Data - Concepts, Theory and Method*", John Wiley & Sons, New York, 1998.

[29] Burges, C. J. C, "A tutorial on support vector machines for pattern recognition", *Data Mining and Knowledge Discovery*, 2(2), 1998, pp.121-167.

[30] Vanajakshi, L. and Rilett, L.R, "A Comparison of the Performance of Artificial Neural Network and Support Vector Machines for the Prediction of Traffic Speed", *proceedings of the IEEE Intelligent Vehicles Symposium*, University of Parma, Parma, Italy, 2004, pp. 194-199.

[31] E. Allwein, R.E. Schapire and Y. Singer, Reducing multiclass to binary: A unifying approach for margin classifiers, *Journal of Machine Learning Research*, 1, 2000, pp.113–141.

[32] T.G. Dietterich and G. Bakiri, Solving multiclass learning problems via error-correcting output codes, *Journal of Artificial Intel Research*, 2 1995, pp.263–286.

[33] Breiman, L, Bagging predictors. *Machine Learning*, 24(2), 1996a, pp.123– 140.

[34] Breiman.L, "Bias, Variance, and Arcing Classifiers", *Technical Report 460*, Department of Statistics, University of California, Berkeley, CA, 1996.

[35] Jiawei Han, Micheline Kamber, *DataMining – Concepts and Techniques*, Elsevier Publications, 2003.

[36] Ira Cohen, Qi Tian, Xiang Sean Zhou and ThomsS.Huang, "Feature Selection Using Principal Feature Analysis", *Proceedings of the 15th international conference on Multimedia*, Augsburg, Germany, September, 2007, pp. 25-29.

[37] KDD'99 dataset, http://kdd.ics.uci.edu/databases, Irvine, CA, USA, 2010.

[38] Kohavi, R, "A study of cross-validation and bootstrap for accuracy estimation and model selection*", Proceedings of International Joint Conference on Artificial Intelligence*, 1995, pp. 1137–1143.

M. Govindarajan received the B.E and M.E and Ph.D Degree in Computer Science and Engineering from Annamalai University, Tamil Nadu, India in 2001 and 2005 and 2010 respectively. He did his post-doctoral research in the Department of Computing, Faculty of Engineering and Physical Sciences, University of Surrey, Guildford, Surrey, United Kingdom in 2011 and CSIR Centre for Mathematical Modelling and Computer Simulation, Bangalore in 2013. He is currently an Assistant Professor at the Department of Computer Science and Engineering, Annamalai University, Tamil Nadu, India. He has presented and published more than 85 papers at Conferences and Journals and also received best paper awards. He has delivered invited talks at various national and international conferences. His current Research Interests include Data Mining and its applications, Web Mining, Text Mining, and Sentiment Mining. He was the recipient of the Achievement Award for the field and to the Conference Bio-Engineering, Computer science, Knowledge Mining (2006), Prague, Czech Republic, Career Award for Young Teachers (2006), All India Council for Technical Education, New Delhi, India and Young Scientist International Travel Award (2012), Department of Science and Technology, Government of India New Delhi. He is Young Scientists awardee under Fast Track Scheme (2013), Department of Science and Technology, Government of India, New Delhi and also granted Young Scientist Fellowship (2013), Tamil Nadu State Council for Science and Technology, Government of Tamil Nadu, Chennai. He has visited countries like Czech Republic, Austria, Thailand, United Kingdom, Malaysia, U.S.A, and Singapore. He is an active Member of various professional bodies and Editorial Board Member of various conferences and journals.