PAPER Special Section on Information and Communication System Security

HFSTE: Hybrid Feature Selections and Tree-Based Classifiers Ensemble for Intrusion Detection System

Bayu Adhi TAMA[†,††a)], ***Nonmember and*** **Kyung-Hyune RHEE**[††b)], ***Member***

SUMMARY Anomaly detection is one approach in intrusion detection systems (IDSs) which aims at capturing any deviation from the profiles of normal network activities. However, it suffers from high false alarm rate since it has impediment to distinguish the boundaries between normal and attack profiles. In this paper, we propose an effective anomaly detection approach by hybridizing three techniques, i.e. particle swarm optimization (PSO), ant colony optimization (ACO), and genetic algorithm (GA) for feature selection and ensemble of four tree-based classifiers, i.e. random forest (RF), naive bayes tree (NBT), logistic model trees (LMT), and reduces error pruning tree (REPT) for classification. Proposed approach is implemented on NSL-KDD dataset and from the experimental result, it significantly outperforms the existing methods in terms of accuracy and false alarm rate.

key words: *classifier ensemble, intrusion detection systems, tree-based classifiers, hybrid feature selection*

1. Introduction

As number of Internet users has been mushrooming in the recent decades, a plethora of attacks have been proliferated over time. A large number of attacks have been discovered, but some of them are continuously rising. Intrusion detection systems (IDSs) are expected to reduce the escalation of such attacks before they cause a certain damage [1].

The objective of an IDS is to provide the promising protection system in computer networks. It deals with a security countermeasure that monitoring, detecting, and repelling any malicious activities over computer networks. It also can be used to evade the network from being targeted by an attacker such as probe attack that breach the availability, confidentiality, and integrity of invaluable information sources [2].

Based on the use of information analysis, IDSs are commonly grouped into two categories, called signatured-based and anomaly-based intrusion detection system. Signature-based system generates alarms when a known attacks occurs. It is able to detect known attacks instantly with a lower false alarm rate. Apart from these advantages, signature-based system possesses difficulty to detect novel attacks. In a different manner, anomaly-based system detects the objects that behave significantly different from the normal profile, thus it is able to detect new types of attack. Nevertheless, anomaly-based system is obstructed by high false alarm rate and even in a hazardous case, some attackers can use anomaly profile as normal network pattern to train an IDS, so that it will misidentify malicious profile as normal.

Since anomaly-based IDS can detect novel and unfamiliar attacks, it has remained a profoundly research topic in the realm of IDS in the recent decades [3]. Anomaly-based IDS relies on how well the model is trained to predict new future attack patterns. In addition, anomaly-based IDS is also a binary classification problem in which it attempts to classify network traffic either as normal or malicious with resulting higher predictive accuracy while maintaining lower false alarm rate. Specifically, supervised learning algorithms use labeled instances to create a model and the future unknown instances can be labeled using the model.

However, with a large number of features, getting a superior classification accuracy calls for sophisticated computing resources. In the context of modern intrusion detection and prevention, fast detection capability with high accuracy and low false alarm rate are much indispensable. Hence, fast detection approach could be achieved using appropriate feature selection technique and high detection accuracy could be obtained using ensemble of lightweight classifier combination approach which requires a restricted computational resource.

Classifier ensemble or multiple classifier system (MCS) has been widely employed for IDSs since they have better performance in comparison with single classifier [4]. It is deployed by incorporating several base classifiers to predict final class output. In this paper we focus on the performance evaluation of tree-based classifiers ensemble, i.e. random forest (RF), naive bayes tree (NBT), logistic model trees (LMT), and reduces error pruning tree (REPT) using voting combination approach. Classifier significant test is carried out to measure how much the classifier ensemble is significant by comparing with a single classifier using the statistical significant test.

The rest of the paper is organized as follows. Section 2 covers a brief review of anomaly-based IDS in the existing literature, whilst proposed intrusion detection model is highlighted in Sect. 3. Experimental design is presented in Sect. 4 and the discussion of experimental result is detailed

Manuscript received September 8, 2016.
Manuscript revised December 19, 2016.
Manuscript publicized May 18, 2017.

[†]The author is with Faculty of Computer Science, University of Sriwijaya, Indonesia.
[††]The authors are with Laboratory of Information Security and Internet Applications (LISIA), Dept. of IT Convergence and Application Engineering, Pukyong National University, Busan, South Korea.
a) E-mail: bayuat@pukyong.ac.kr
b) E-mail: khrhee@pknu.ac.kr (Corresponding author)

DOI: 10.1587/transinf.2016ICP0018

in Sect. 5. Finally, some concluding remarks are drawn in Sect. 6.

2. Related Work

Many previous researchers have utilized classifier ensemble for IDSs. The details contribution of each research are presented in this section. We merely consider to include the implementation of classifier ensemble for anomaly-based intrusion detection which is on our current interest. Earlier work of classifier ensemble for anomaly detection is proposed by [5]. Three base classifiers, i.e. neural network, support vector machine, and multivariate regression splines are combined to predict a final class using majority voting. The performance of the proposed approach was evaluated on the KDDCup 99 dataset with an accuracy as a performance metric. The authors also applied feature selection to reduce the computational overhead while training dataset with many features.

Ensemble of decision tree and support vector machine using weighted ensemble approach is suggested by [6]. Similar to the previous work, accuracy is used as performance evaluation and the proposed approach is implemented on the full features set of KDDCup 99 dataset. A classifier ensemble, called Adaboost is used to improve the performance of decision stump [7]. Two performance metrics, i.e. precision and false alarm rate are used to evaluate the proposed method on the reduced-features of KDDCup 99 dataset. A product rule combination is proposed by [8]. It is utilized as the combiner to predict final class prediction in which area under ROC curve (AUC) is employed as a performance evaluation metric. This proposed scheme then is applied on the KDDCup 99 dataset which no feature selection is performed.

Three different classification combination approach, i.e. minimum probability, maximum probability, and product rule is suggested by [9] to improve the performance of four base classifiers, i.e. k-means and υ-support vector classification. Performances of classifiers are evaluated using standard KDDCup 99 dataset with reduced number of features, whilst precision and the false alarm rate are considered as evaluation metric. Classifier fusion using Bagging strategy is suggested by [10]. It is exploited to incorporate the output of two neural network algorithms, i.e. multilayer perceptron and radial basis function as base classifiers. In order to estimate the performance implementation of the proposed approach, accuracy is considered as a performance metric and it is applied on the private dataset which feature selection is also done.

Voting combiner is adopted in [11] to fuse two base classifiers, i.e. neural network and decision tree. The experiment is carried out on the full features set of KDDCup 99 dataset with several performance metrics, including true positive rate, false positive rate, precision, recall, and F_1 measure. The recent work of anomaly-based IDS using classifier ensemble is proposed by [12]. Two tree-based classifiers, i.e. NBTree and random tree were merged to obtain a better final prediction using sum rule probability. This work is claimed as the highest result achieved so far using the complete features of NSL-KDD dataset.

To distinguish between our approach and the existing studies, we defined some viewpoints of them as follows.

- Most studies use old version of KDDCup 99 dataset for anomaly detection where NSL-KDD dataset is still underexplored.
- Most studies use one feature selection technique so it is indispensable to choose the proper feature selection method by hybridizing several combination approaches.
- Most studies do not examine the performance difference between classifier ensemble and single classifier in the ensemble.
- Most studies do not undertake a statistical significant test to prove of significance of the results.

Our proposed model is a combination of multiple feature selection techniques and ensemble of four base classifiers for anomaly-based intrusion detection systems. For each feature selection algorithm, the performance is measured in term of accuracy metric of support vector machine (SVM) [13] classifier. SVM is chosen since it is one of the prevalent techniques used in the literature. For the experiment, an improved version of KDDCup 99, called NSL-KDD [14] is used. A hybrid feature selection comprises three algorithms, i.e. particle swarm optimization (PSO) [15], ant colony optimization (ACO) [16], and genetic algorithm (GA) [17] are employed in order to get the most suitable subset of features. In addition, four classification algorithms, i.e. random forest (RF) [18], Naive-bayes tree (NBT) [19], logistic model trees (LMT) [20], and Reduces error pruning tree (REPT) [21] are combined using voting rule [22] fusion scheme. The significant results of each classifier are then assessed using Friedman test [23] and Nemenyi post hoc test [24].

The major pillar of contribution of this paper lies in several axes:

- Hybrid use of feature selection and classifier ensemble simultaneously.
- Comparing the performance of classifier ensemble with base classifier with respect to classification problem in anomaly-based IDSs.
- We show that a voting rule combination approach is the best choice for anomaly-based IDSs since it gives us a better result compared to the existing ones.
- Considering a thoroughly iterative process in the experiment to choose the best parameter setting for feature selection.
- Providing two statistical significant tests to prove that the differences among classifiers are significant.

3. Proposed Approach

In this section, we describe the background of feature selec-

tion algorithms, base classifiers, classifiers ensemble, and the proposed model.

3.1 Feature Selection Algorithms

The feature selection (FS) is the problem of selecting a subset of attributes from a feature set in order to obtain a precise, compact, and fast classifier performance. For attribute evaluator, we adopt correlation-based feature selection (CFS) which is one of the leading feature subset selection method in machine learning and pattern recognition [25]. The worth of a subset of attributes is evaluated using entropy and information gain theory. The lack of computation using information gain is symmetrical uncertainty and biased of feature with more values. Hence, CFS takes a coefficient to compensate information gain's bias toward attribute with more values and to normalize its value to the range [0, 1].

Three different search methods for the attribute selection are describe as follows.

- *Particle swarm optimization (PSO).* It is used to search the set of all possible features so that the best set of features can be obtained [4]. PSO is firstly introduced by Kennedy and Eberhart [15], is one of the computation technique which is inspired by behavior of flying birds and their means of information exchange to solve the problems. Each particle in the swarm represents possible solution. A number of particle is located in the hyperspace, which has random position φ_i and velocity υ_i. The basic update rule for the position and the speed is depicted in Eqs. (1) and (2), respectively.

$$\varphi_i(t+1) = \varphi_i + \upsilon_i(t+1) \tag{1}$$

$$\upsilon_i(t+1) = \omega\upsilon_i(t) + c_1 r_1(p_i - x_i) + c_2 r_2(g - x_i) \tag{2}$$

  Where ω denotes inertia weight constant, c_1 and c_2 denotes cognitive and social learning constant, respectively, r_1 and r_2 represent random numbers, respectively, p_i is personal best position of particle i, and finally, g is a global best position among all particles in the swarm.
- *Ant Colony Optimization (ACO).* It is represented as a graph, which nodes represents features, with the edges between them denoting the choice of the next feature. The search of the final feature subset is an ant traversal through the graph where a minimum number of nodes is visited that satisfying the traversal stopping criterion [16], [26]. A probabilistic transition rule is used to give an indication on which features are more informative on the currently selected features. It denotes the probability of an ant at feature i choosing to travel to feature j at time t:

$$p_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{l \in J_i^k} [\tau_{il}(t)]^\alpha \cdot [\eta_{il}]^\beta} \tag{3}$$

  Where k is the number of ants, J_i^k is the set of ant k's unvisited features, η_{ij} is the heuristic desirability of choosing feature j when currently at feature i and $\tau_{ij}(t)$ is the amount of virtual pheromone on edge (i, j). The choice of α and β is determined experimentally.
- *Genetic Algorithm (GA).* It is depicted by one chromosome which is a set of the features. Gene is a feature that has binary value 1 or 0, which means that there is or is not a particular feature in the set, respectively. Goldberg strategy is commonly used to discover an ideal set of features. The subset evaluator function with k-cross validation is applied to evaluate the input features. We consider to set the value of the initial population, maximum number of generations, mutation, crossover probability, k, and random seed number are 30, 30, 0.01, 0.9, 10 and 1, respectively.

3.2 Base Classifiers

As it has been mentioned previously, we consider four tree-based classifiers as base classifiers in the ensemble. Random forest (RF) [18], Naive-bayes tree (NBT) [19], logistic model trees (LMT) [20], and Reduces error pruning tree (REPT) [21] are selected since they require less computational resource and have shown better predictive accuracy in many applications [27]. We set the same parameters, either as a member of ensemble or as a single classifier. We briefly discuss the aforementioned base classifiers as follows.

- *RF*. This generates a number of trees. Random trees are grown without pre- or post-pruning, which contributes to their diversity. At each node, the feature to split upon is chosen from a randomized split of the original feature. Classification accuracy is positively gained due to the diversity of the trees. There are only two parameters in RF, i.e. number of trees and the number of variables to try at each split. We consider large number of trees is 1000 and set the number of variables to the square root of the total number of predictors.
- *NBT*. It is a hybrid approach that incorporate the advantages of decision tree and Naive-Bayes. The final decision tree is built with univariate splits at each node, but with Naive-Bayes classifiers at the leaves. The decision-tree segments the data and each segment of the data, represented by a leaf, is described through a Naive-Bayes classifier. No parameter setting is required for this algorithm.
- *LMT*. It is similar to NBT, but logistic regression function is used at the leaves of the tree. We consider the use of logitboost algorithm as the regression function, the number on boosting iteration is cross-validated, and the minimum number of instances at which a node is considered for splitting is 15.
- *REPT*. It is a fast decision tree learning algorithm which tree is built using the information gain with entropy. It takes reduce error pruning in order to minimize the error from the variance. We set the parameter of the algorithm as follows. The minimum total weight

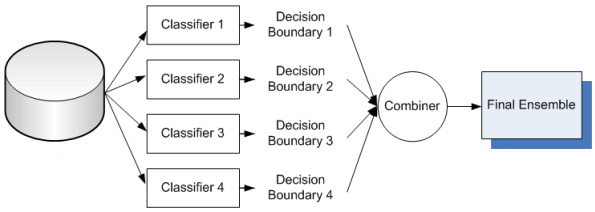

Fig. 1 Illustration of classifiers ensemble

of the instances in a leaf is 2, the amount of data used for pruning (folds) is 3, and tree pruning is applied.

3.3 Classifiers Ensemble

As illustrated in Fig. 1, the ensemble combines different parameters of all base classifiers using combination rules. Let T individual classifiers $\{h_1, \ldots, h_T\}$ be given and we want to combine h_i's to predict the class label from a set of l possible class label $\{c_1, \ldots, c_l\}$. It is assumed that for an instance x, the final outputs of the classifier h_i are given as an l-dimensional label vector $(h_i^1(x), \ldots, h_i^l(x))^T$ which $h_i^j(x)$ is the output of h_i for the class label c_j. Hence, $h_i^j(x) \in \{0, 1\}$ which takes value one if h_i predicts c_j as the class label and zero otherwise.

In majority voting, every classifier votes for one class label, and the final output class label is the one that receives more than half of the votes, otherwise a rejection option is given. Hence, the output class label of majority voting is expressed as:

$$H(x) = \begin{cases} c_j \quad if \sum_{i=1}^{T} h_i^j(x) > \frac{1}{2} \sum_{k=1}^{l} \sum_{i=1}^{T} h_i^k(x) \\ rejection \end{cases} \tag{4}$$

3.4 The Proposed Model

In this section, a hybrid feature selection and classifier ensemble for anomaly detection is briefly presented. As shown in Fig. 2, the proposed model comprises two stages such as feature selection and classification (modeling). In the first stage, three feature selection techniques are gathered in order to obtain the most representative features subset for enhancing the performance of the classification in the classification (modeling) stage. The three feature selection techniques involved in this stage are PSO, ACO, and GA. Parameters tuning of all feature selection techniques are performed and the selected feature subset are then applied for SVM classification. The optimal parameters in this stage are determined by the SVM classification accuracy.

In order to obtain the SVM classification accuracy, a hold-out evaluation method is adopted in which dataset are divided into two parts, e.g. 70% and 30% are used for training and testing, respectively. In addition to the best selected features, the output of the first stage is the most appropriate feature selection technique. In the second stage, four base classifiers, i.e. RF, NBT, LMT, and REPT as well as

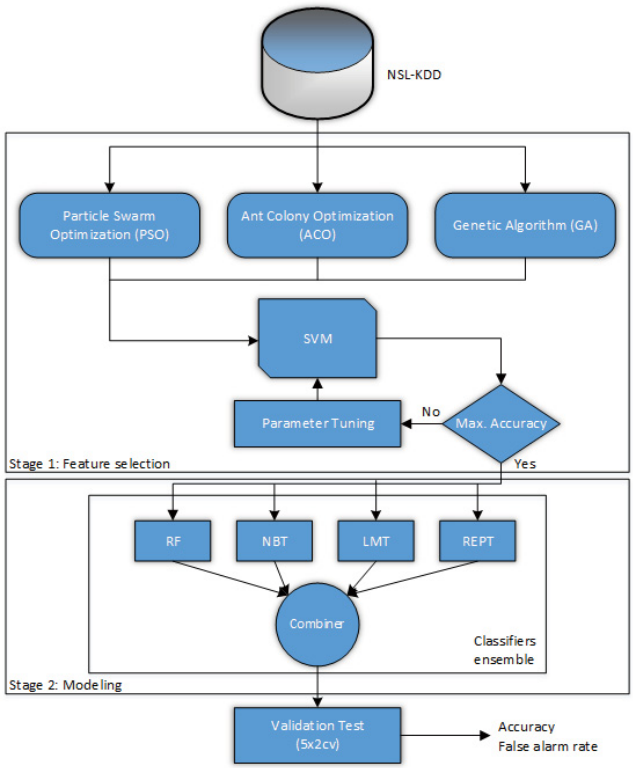

Fig. 2 Proposed model for anomaly detection

ensemble of these base classifiers are used for classification (modeling). The performance of base classifiers as single classifier and classifiers ensemble are validated using five times of 2-cross validation ($5 \times 2cv$) [28] in terms of two metrics, i.e. accuracy and false alarm rate.

4. Experimental Design

4.1 Experimental Setup

The overall performance of classifiers are evaluated in *R* environment using *RWeka* library [29]. The experiment is conducted on a machine with Windows 7, 16GB RAM, and Intel® CPU 3.5GHz.

4.2 Dataset Description

KDD Cup 99 dataset has been widely used for intrusion detection [14]. It is considerably accepted as a standard dataset for benchmarking. However, the dataset has inherent problems due to the synthetic characteristic of the data. For this reason, we considered to use NSL-KDD dataset since it does not include redundant instances which lead the classifiers to produce biased result. The dataset possesses 41 attributes and one class label attribute. The 20% of NSL-KDD training set contains 25192 instances, which is composed of two classes, e.g. anomaly class (13499 instances) and normal class (11743 instances).

4.3 Performance Metrics

All classifiers are evaluated using performance metrics, i.e. average accuracy and false alarm rate (FAR). We considered to employ these performance metrics since they have been taken into account in the previous related studies (see Sect. 2). These evaluation metrics are briefly calculated as follows.

$$Average\ Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{5}$$

$$FAR = \frac{FP}{FP + TN} \tag{6}$$

where True Positive (TP) is the number of instances correctly identified as belonging to the normal class, False Positive (FP) or Type I error is the number of instances incorrectly identified as belonging to the normal class, True Negative (TN) is the number of instances correctly identified as belonging to the anomaly class, and False Negative (FN) or Type II error is the number of instances incorrectly identified as belonging to the anomaly class.

4.4 Statistical Significant Test

To provide a detailed comparative study among classifier ensemble schemes, statistical test is employed to prove that the differences among classifiers are significant [30]. The Friedman test [23] is used to test whether the differences among the classifiers in term of evaluation metric are significant [31]. It is a non-parametric test which is equivalent to the repeated-measures ANOVA [31]. In addition, it ranks the classifiers, with the best algorithm receiving rank 1, and the worst classifier receiving rank equal to the number of classifiers. Friedman test is defined as follows.

$$\chi_F^2 = \frac{12N}{k(k+1)}\left[\sum_j R_j^2 - \frac{k(k+1)^2}{4}\right] \tag{7}$$

where N is the number of elements, k is the number of classifiers, and R_j is the average rank of the jth of k classifiers. The average rank is defined as $R_j = \frac{1}{N}\sum_i^N r_i^j$, where r_i^j is the rank of the jth of k classifiers on the ith of N elements.

When the Friedman test is rejected, we carry out post-hoc test using Nemenyi test [24] to determine which classifiers are significantly different. Two classifiers are significantly different if the corresponding average ranks differ by at least the critical difference (CD), which is defined as:

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6N}} \tag{8}$$

where the critical values q_α are computed using the Studentized range statistic divided by $\sqrt{2}$, N is the number of elements and k is the number of classifiers to be compared [31].

5. Experimental Result and Discussion

This section shows the experimental result of the proposed model. As presented in Sect. 3.4, the three different FS techniques are applied and their parameters are tuned with respect to the SVM classification accuracy. The parameters for each FS technique and the accuracy of SVM are presented in the following section.

Table 1 Parameter setting for PSO

Model	Particles (n)	Selected features	Accuracy (%)
1	2	37	**97.47**
2	5	12	92.88
3	10	19	96.40
4	20	5	83.67
5	50	6	89.20
6	100	6	87.40
7	200	7	91.81
8	500	7	91.31
9	1000	8	91.52
10	2000	8	91.52

5.1 PSO Parameter Setting

In particle swarm optimization FS, parameter n (number of particle) is changed. We set parameter c_1 and c_2 are equal to 2, whilst the maximum number of generations is 30. In literature, these values have been proposed as a generally acceptable setting for most of problems [32]. The output of FS is used for SVM classification model as shown in Table 1.

The outcomes show that model 1 (particle size of 2) has higher classification accuracy than others. It can be seen that the classification accuracy of the model 1 is 97.47%. The thirty-seven features have been successfully obtained by PSO, such as duration, protocol_type, service, flag, src_bytes, dst_bytes, land, wrong_fragment, urgent, hot, num_failed_logins, logged_in, num_compromised, root_shell, su_attempted, num_file_creations, num_shells, num_outbound_cmds, is_host_login, is_guest_login, count, srv_count, serror_rate, srv_serror_rate, rerror_rate, srv_rerror_rate, same_srv_rate, srv_diff_host_rate, dst_host_count, dst_host_same_srv_rate, dst_host_diff_srv_rate, dst_host_same_src_port_rate, dst_host_srv_diff_host_rate, dst_host_serror_rate, dst_host_srv_serror_rate, dst_host_rerror_rate, and dst_host_srv_rerror_rate.

5.2 ACO Parameter Setting

Similar to feature selection using PSO, parameter of k (number of ants) is changed in ACO feature selection. β is a parameter which determines the relative importance of pheromone versus heuristic. With regard to this, we set $\beta = 1$, which gives equal importance to cost minimization while selecting the features. As suggested by [33], local pheromone update strength parameter (α) is set to 0.8. The outcomes of each parameter setting for ACO feature selection and the SVM classification accuracy are presented in Table 2.

It can be seen in Table 2 that model 9 and 10 receives higher accuracy (91.52%) in the SVM classification. Therefore, the selected features of model 9 and 10

Table 2 Parameter setting for ACO

Model	Number of ants (k)	Selected features	Accuracy (%)
1	2	7	90.29
2	5	6	89.01
3	10	8	91.28
4	20	6	89.01
5	50	6	89.18
6	100	6	89.18
7	200	7	90.69
8	500	7	91.31
9	1000	8	**91.52**
10	2000	8	**91.52**

Table 3 Parameter setting for GA

Model	Population size	Selected features	Accuracy (%)
1	2	25	**94.39**
2	5	25	**94.39**
3	10	14	92.31
4	20	10	91.32
5	50	11	89.88
6	100	7	87.76
7	200	7	91.31
8	500	9	91.89
9	1000	8	91.36
10	2000	6	89.20

can be used for building classification model. After conducting feature selection using ACO, 8 features are obtained such as flag, src_bytes, dst_bytes, logged_in, srv_serror_rate, same_srv_rate, diff_srv_rate, and dst_host_srv_diff_host_rate.

5.3 GA Parameter Setting

As it is mentioned previously, feature selection using GA also requires parameters setting. These parameters such as the value of the initial population, maximum number of generations, mutation, crossover probability, k, and random seed number are set to 30, 30, 0.01, 0.9, 10 and 1, respectively. Population size parameter is changed with the same interval number of the previous experiment using PSO and ACO. The results of SVM accuracy and selected features are shown in Table 3.

As depicted in Table 3, model 1 and 2 give the best classification accuracy in SVM classification. They share the same number of selected features (25 features) as well as performance accuracy (94.39%). Hence, selected features obtained by model 1 and 2 can be used for building classification model in the second stage. Twenty-five features have been generated by using GA feature selection, e.g. duration, flag, src_bytes, dst_bytes, land, wrong_fragment, urgent, hot, logged_in, root_shell, su_attempted, num_shells, num_outbound_cmds, is_host_login, count, srv_count, serror_rate, srv_serror_rate, srv_rerror_rate, same_srv_rate, dst_host_diff_srv_rate, dst_host_same_src_port_rate, dst_host_serror_rate, dst_host_rerror_rate, and dst_host_srv_rerror_rate.

5.4 Classifiers Performance Result

After performing feature selection and tuning parameter setting, an appropriate subset features have been obtained as indicated in Table 1–3. The next step is the implementation of all classifiers, i.e. RF, NBT, LMT, and REPT and voting ensemble of these base classifiers. Figure 3 denotes the performance result of all classifiers for each FS technique in terms of accuracy and FAR value. The performance of all classifiers are evaluated using $5 \times 2cv$ [28]. This method divides the dataset randomly into two equal parts. One part is used for training and the other part to test the algorithm, and vice versa. This procedure is then repeated five times. With regard to this, the results presented in this paper are the average value of accuracy and FAR.

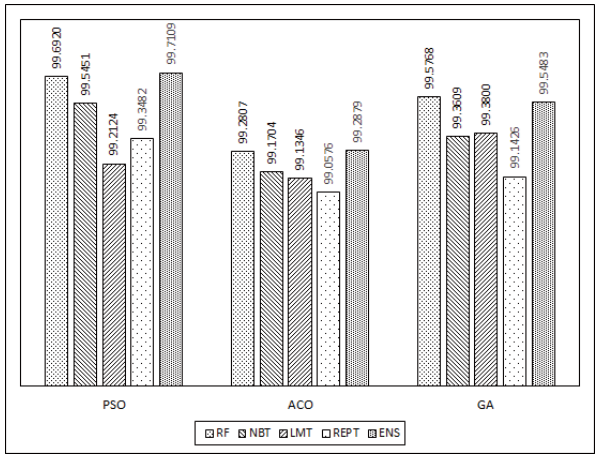

Fig. 3 Average accuracy for each feature selection technique in all classifiers

As depicted in Fig. 3, it is obvious that voting ensemble (ENS) resulted from the PSO feature selection is the best performer in comparison with other FS techniques. Figure 3 confirms that our proposed classifier ensemble also significantly outperforms base classifiers as well as SVM classifier in term of accuracy metric. For instance by using PSO feature selection, ENS gains 99.7109%, whilst RF, NBT, LMT, REPT, and SVM gain 99.6920%, 99.5451%, 99.2124%, and 99.3482%, respectively.

Figure 4 presents the classifier performance of all classifiers in term of FAR metric for each feature selection techniques. It is clear that ENS resulted from the PSO feature selection is the best performer in comparison with other FS techniques. It significantly outperforms other classifiers, i.e. RF, NBT, LMT, and REPT with the lowest false alarm rate. For instance by using PSO feature selection, ENS gains 0.0053, whilst RF, NBT, LMT, and REPT gain 0.0049, 0.0064, 0.0110, and 0.0081, respectively.

Furthermore, in order to ensure that the validation test does not happen by chance, we tested the significance of these result by using the Friedman test. We are only interested to assess the significant differences of all classifiers' accuracy resulted from the PSO feature selection since this result is the best one. The null hypothesis is considered as there is no significant differences of accuracy among three classifiers, and alternative hypothesis is considered as there

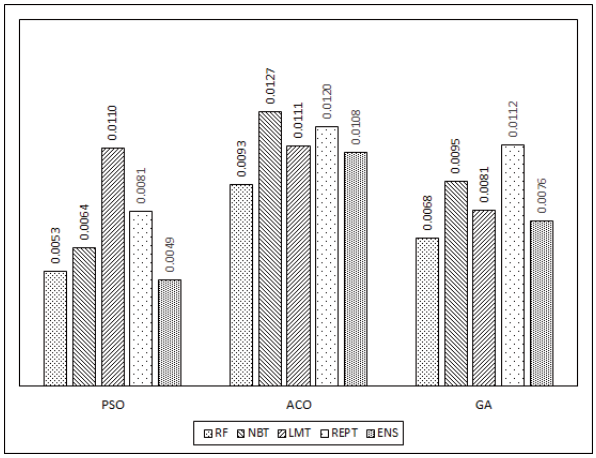

Fig. 4 Average FAR for each feature selection technique in all classifiers

Table 4 The results of classifier significance using Friedman test

χ_F^2	df	p-value
32.72	4	1.363E-06

is significant differences of accuracy among three classifiers. As indicated in Eq. (7), N is the number of elements (10 in our case) and k is the number of classifiers (5 in our case). We fix the level significant level $\alpha = 0.05$ which refers to a confidence level of 95%. The results of classifier significance test are summarized in Table 4.

The result above indicates that there are significant differences among classifiers. However, this result is very conservative so we apply more powerful post hoc test, i.e. Nemenyi test for comparing all classifiers to each other. The critical difference (CD), which represents the rank difference among classifiers, is computed using Eq. (8). The q_α corresponds to the critical values from the Tukey test by dividing it by $\sqrt{2}$ (see Table A.8 in [34]). The two classifiers are significantly different in which their average rank of each classifiers are larger or equal to the CD. For $\alpha = 0.05$ and degree of freedom $(df) = (n-1)(k-1) = 9 \times 4 = 36$, we get $q_\alpha = 4.04$ for the Tukey test. It yields $q_\alpha = 2.86$ for the Nemenyi test. Recall from Eq. (8), we compute CD as follows.

$$CD = 2.86\sqrt{\frac{5(5+1)}{6 \times 10}} = 2.02 \quad (9)$$

To determine which classifiers are significantly different, it is required to calculate the average rankings of the accuracy and then compare which differences are greater than 2.02. Another method is we can plot the critical difference for each classifier as shown in Fig. 5. First of all, there is no performance difference between ENS and RF. The performance of ENS differs highly significant to LMT and REPT ($p < 0.01$) whilst other comparisons are not significant ($p > 0.05$).

Subsequently, in order to demonstrate that our proposed approach is comparable to other methods, we compare our result with the existing approaches where 20% of

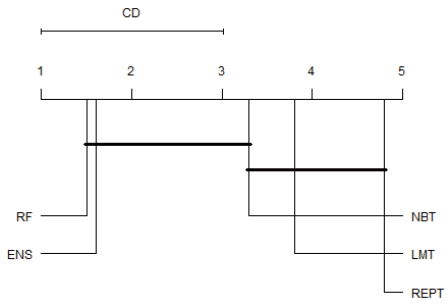

Fig. 5 Critical difference of all classifiers in term of accuracy metric

Table 5 Comparison of the proposed approach for $10f - cv$

Study	Feature selection	Average accuracy (%)	Significant Test
NBTree [14]	No	99.67	No
Discriminative Multinomial Naive-Bayes [35]	N2B	96.5	No
Adaboost+GA [36]	No	99.57	No
RT+NBT [12]	No	99.53	No
Proposed Approach	**PSO**	**99.77**	**Yes**

NSL-KDD dataset is trained and tested using 10-folds cross validation ($10f - cv$). Table 5 depicts the comparison result for the experiment using 10-folds cross validation. It is obvious that our proposed approach considerably outperforms other methods found in the literature.

6. Conclusion

This paper proposes the hybrid approach of feature selections and tree-based classifiers ensemble for intrusion detection systems. Three feature selection techniques, i.e. PSO, ACO, and GA are involved in order to obtain the best subset of features. Moreover, four tree-based classifier algorithms, i.e. RF, NBT, LMT, and REPT are combined for classification analysis. Based on our experimental result, it can be revealed that the proposed scheme yields detection accuracy 99.77%, significantly outperforms the existing methods applied on the NSL-KDD dataset. We also conclude that classifiers ensemble performs better than single classifier in the pool. Our work contributes to the existing literature by providing a comprehensive statistical significant test, including post-hoc test in the evaluation of classifier algorithms for intrusion detection systems.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIP) No. NRF-2014R1A2A1A11052981.

References

[1] B.A. Tama and K.H. Rhee, "Performance analysis of multiple classifier system in DoS attack detection," International Workshop on Information Security Applications, pp.339–347, 2015.

[2] B.A. Tama and K.H. Rhee, "Data mining techniques in DoS/DDoS attack detection: A literature review," Information (Japan), vol.18, no.8, p.3739, 2015.
[3] X.-S. Gan, J.-S. Duanmu, J.-F. Wang, and W. Cong, "Anomaly intrusion detection based on PLS feature extraction and core vector machine," Knowledge-Based Systems, vol.40, pp.1–6, 2013.
[4] B.A. Tama and K.H. Rhee, "A combination of PSO-based feature selection and tree-based classifiers ensemble for intrusion detection systems," in Advances in Computer Science and Ubiquitous Computing, vol.373, pp.489–495, Springer, 2015.
[5] S. Mukkamala, A.H. Sung, and A. Abraham, "Intrusion detection using an ensemble of intelligent paradigms," J. Netw. Comput. Appl., vol.28, no.2, pp.167–182, 2005.
[6] S. Peddabachigari, A. Abraham, C. Grosan, and J. Thomas, "Modeling intrusion detection system using hybrid intelligent systems," J. Netw. Comput. Appl., vol.30, no.1, pp.114–132, 2007.
[7] W. Hu, W. Hu, and S. Maybank, "Adaboost-based algorithm for network intrusion detection," IEEE Trans. Syst., Man, Cybern. B, Cybernetics, vol.38, no.2, pp.577–583, 2008.
[8] J.B.D. Cabrera, C. Gutiérrez, and R.K. Mehra, "Ensemble methods for anomaly detection and distributed intrusion detection in mobile ad-hoc networks," Information Fusion, vol.9, no.1, pp.96–119, 2008.
[9] G. Giacinto, R. Perdisci, M.D. Rio, and F. Roli, "Intrusion detection in computer networks by a modular ensemble of one-class classifiers," Information Fusion, vol.9, no.1, pp.69–82, 2008.
[10] M. Govindarajan and R. Chandrasekaran, "Intrusion detection using neural based hybrid classification methods," Computer Networks, vol.55, no.8, pp.1662–1671, 2011.
[11] S.S.S. Sindhu, S. Geetha, and A. Kannan, "Decision tree based light weight intrusion detection using a wrapper approach," Expert. Syst. Appl., vol.39, no.1, pp.129–141, 2012.
[12] J. Kevric, S. Jukic, and A. Subasi, "An effective combining classifier approach using tree algorithms for network intrusion detection," Neural Computing and Applications, pp.1–8, 2016.
[13] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," ACM Transactions on Intelligent Systems and Technology (TIST), vol.2, no.3, pp.1–27, 2011.
[14] M. Tavallaee, E. Bagheri, W. Lu, and A.A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," The Second IEEE Symposium on Computational Intelligence for Security and Defence Applications 2009, pp.1–6, 2009.
[15] J. Kennedy and R.C. Eberhart, "A discrete binary version of the particle swarm algorithm," IEEE International Conference on Systems, Man, and Cybernetics – Computational Cybernetics and Simulation, pp.4104–4108, IEEE, 1997.
[16] E. Bonabeau, M. Dorigo, and G. Theraulaz, Swarm intelligence: from natural to artificial systems, Oxford University Press, 1999.
[17] M. Mitchell, An introduction to genetic algorithms, MIT Press, 1998.
[18] L. Breiman, "Random forests," Machine learning, vol.45, no.1, pp.5–32, 2001.
[19] R. Kohavi, "Scaling up the accuracy of Naive-Bayes classifiers: A decision-tree hybrid," KDD, pp.202–207, Citeseer, 1996.
[20] N. Landwehr, M. Hall, and E. Frank, "Logistic model trees," Machine Learning, vol.59, no.1-2, pp.161–205, 2005.
[21] J.R. Quinlan, "Simplifying decision trees," International Journal of Human-Computer Studies, vol.51, no.2, pp.497–510, 1999.
[22] L. Kuncheva, Combining pattern classifiers: methods and algorithms 2nd edition, John Wiley & Sons, 2014.
[23] M. Friedman, "A comparison of alternative tests of significance for the problem of m rankings," The Annals of Mathematical Statistics, vol.11, no.1, pp.86–92, 1940.
[24] P. Nemenyi, "Distribution-free multiple comparisons," Biometrics, p.263, 1962.
[25] M.A. Hall, Correlation-based feature selection for machine learning, Ph.D. thesis, The University of Waikato, 1999.
[26] R. Jensen and Q. Shen, "Fuzzy-rough data reduction with ant colony optimization," Fuzzy sets and systems, vol.149, no.1, pp.5–20, 2005.
[27] B.A. Tama, "Learning to prevent inactive student of Indonesia open university," Journal of Information Processing Systems, vol.11, no.2, pp.165–172, 2015.
[28] T.G. Dietterich, "Approximate statistical tests for comparing supervised classification learning algorithms," Neural computation, vol.10, no.7, pp.1895–1923, 1998.
[29] K. Hornik, C. Buchta, and A. Zeileis, "Open-source machine learning: R meets weka," Computational Statistics, vol.24, no.2, pp.225–232, 2009.
[30] S. García, A. Fernández, J. Luengo, and F. Herrera, "Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power," Information Sciences, vol.180, no.10, pp.2044–2064, 2010.
[31] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," Journal of Machine learning research, vol.7, no.Jan, pp.1–30, 2006.
[32] E. Ozcan and C.K. Mohan, "Particle swarm optimization: surfing the waves," Proc. 1999 Congress on Evolutionary Computation, 1999, CEC 99, IEEE, 1999.
[33] R.K. Sivagaminathan and S. Ramakrishnan, "A hybrid approach for feature subset selection using neural networks and ant colony optimization," Expert. Syst. Appl., vol.33, no.1, pp.49–60, 2007.
[34] N. Japkowicz and M. Shah, Evaluating learning algorithms: A classification perspective, Cambridge University Press, 2011.
[35] M. Panda, A. Abraham, and M.R. Patra, "Discriminative multinomial naive bayes for network intrusion detection," 2010 Sixth International Conference on Information Assurance and Security (IAS), pp.5–10, IEEE, 2010.
[36] H.M. Harb and A.S. Desuky, "Adaboost ensemble with genetic algorithm post optimization for intrusion detection," International Journal of Computer Science Issues, vol.8, no.5, 1, 2011.

Bayu Adhi Tama received his bachelor degree in electrical engineering from Universitas Sriwijaya and master degree in information technology from Universitas Indonesia in 2004 and 2008, respectively. Currently, he is working toward a Ph.D. degree at the Laboratory of Information Security and Internet Applications (LISIA), Pukyong National University, Republic of Korea. He is an awardee of the Korean Government Scholarship Program in 2013–2018. His research interests include data mining and machine learning techniques for cyber security.

Kyung-Hyune Rhee received his M.S. and Ph.D. degrees from Korea Advanced Institute of Science and Technology (KAIST), Republic of Korea in 1985 and 1992, respectively. He worked as a senior researcher in Electronic and Telecommunications Research Institute (ETRI), Republic of Korea from 1985 to 1993. He also worked as a visiting scholar in the University of Adelaide, the University of Tokyo, and the University of California, Irvine. He has served as a Chairman of Division of Information and Communication Technology, Colombo Plan Staff College for Technician Education in Manila, the Philippines. He is currently a professor in the Department of IT Convergence and Application Engineering, Pukyong National University, Republic of Korea. His research interests center on key management and its applications, mobile communication security and security evaluation of cryptographic algorithms.