BRCORP
TECHNOLOGY | INNOVATION | INDUSTRY

5th World Conference on
Applied Sciences, Engineering & Technology
02-04 June 2016, HCMUT, Vietnam

Design and Development of an Enhanced Random Forest Method to Reduce the Attributes

A. MALATHI[1], A. GANDHIMATHI[2]
[1]*Department of Computer Science, Government Arts College, Coimbatore, India*
[2]*Department of Civil Engineering, Kumara Guru College of Technology, Coimbatore, India*
Email: malathi.arunachalam@yahoo.com, gandhimathi.a.ce@kct.ac.in

Abstract: This paper explains the enhancement of Random forest method to reduce the attributes and to make the efficient usage of IDS. The Random Forest is a new ensemble algorithm for classification. This Random Forest uses ensemble unpruned classification or a regression algorithm and which generated many decision trees. Random Forest uses both bagging and boosting as a successful approach for tree building. It is a collection of tree predictors with the same distribution for all trees in the forests. It builds many trees which minimizes the classification errors based on a bootstrap sample of training data from the original dataset using a tree classification algorithm. Random Forest is a most successful method for pre-processing the dataset. The dataset contains many unwanted and irrelevant attributes. This enhanced Random Forest is used to classify the intrusions detection. Pre-process is the process of keeping the dataset ready for the process with necessary attributes for its process.

Keywords: *Random Forest, Pre-processing and Attribute Reduction*

Introduction:

There are many preprocessing algorithms available to preprocess the dataset. Preprocessing is the process of keeping the dataset ready for the process [1]. Improvements in classification accuracy have given the output from growing an ensemble of trees and letting them vote for the most general class. To grow these bands, often random vectors are produced that govern the growth of each tree in the ensemble [3].
Random Forest model has risen significantly in its efficiency and for some really good reasons. Random forest method can be implemented easily and quickly to any data science problems to get best set of benchmark results. They are incredibly powerful and can be implemented firstly. Random forest can be applied in predictive models.

Random Forest Algorithm:

Random Forest is classification algorithm. Random Forest is a most successful method for pre-processing the given and large dataset. Decision tree is a prevalent method for various machine learning tasks. Random Forest is a traditional algorithm used for intrusion detection. This algorithm can handle very large and imbalanced dataset. The machine learning is closest to meet the requirement to serve some procedure for data mining. Random forest is invariant under scaling and various other alterations of feature values, is robust to inclusion of unrelated features, and produces inspect able models. However, they are seldom accurate
In particular, decision trees that are grown very deep tend to learn highly unequal patterns. Decision trees over fit their training sets, because they have low bias, but very high alteration. Random forests are a way of be in the region of multiple deep decision trees, trained on different parts of the same training data set, with the goal of reducing the alteration. This comes at the expense of a small increase in the bias and some loss of interpretability, but generally greatly boosts the performance of the final model.
The training algorithm for random forests applies the general technique of bootstrap aggregating, or bagging, to tree learners. Given a training set $X = x_1$, ..., x_n with responses $Y = y_1$, ..., y_n, bagging repeatedly (B times) selects a random sample with replacement of the training set and fits trees to these samples:
For $b = 1$, ..., B:

1. Sample, with replacement, n training examples from X, Y; call these X_b, Y_b.
2. Train a decision or regression tree f_b on X_b, Y_b.

After training, predictions for unseen samples x' can be made by averaging the predictions from all the individual regression trees on x':

$$\hat{f} = \frac{1}{B}\sum_{b=1}^{B} \hat{f}_b(x')$$

or by taking the majority vote in the case of decision trees.
This bootstrapping technique leads to better performance because it decreases the variance of the model, without increasing the bias. When single tree is predicted, the highly sensitive to noise in its training set, the average of many trees is not, as long as the trees are not correlated. Simply training many trees on a single training set would give strongly correlated trees (or even the same tree many times, if the training algorithm is deterministic); bootstrap sampling is a way of de-correlating the trees by showing them different training sets.
The number of samples/trees, B, is a free parameter. Typically, a few hundred to several thousand trees are used, depending on the size and nature of the

training set. An optimal number of trees *B* can be found using cross-validation, or by observing the *out-of-bag error*: the mean prediction error on each training sample x_i, using only the trees that did not have x_i in their bootstrap sample. The training and test error tend to level off after some number of trees have been fit.

Data Set:
Random Forest algorithm is best suited algorithm to pre-process the dataset. This handles the very large data set and imbalanced dataset.

Imbalance Data set
The imbalanced data set contains un related data and more number. It is one of the most famous and learning algorithms available. For many datasets, it produces a highly accurate classifier. A data set is class-imbalanced if one class contains significantly more samples than the other. In other words, non-events have very large number of records than events in dependent variable.

In such cases, it is inspiring to create an appropriate testing and training data sets, given that most classifiers are built with the assumption that the test data is drawn from the same distribution as the training data.

Presenting imbalanced data to a classifier will produce unwanted results such as a much lower performance on the testing than on the training data. To deal with this problem, you can do under sample of non-events.

Under sampling
Under sampling means down-sizing the non-events by removing clarifications at random until the dataset is balanced.

Random forest is affected by multicollinearity but not by outlier problem.

Impute missing values within random forest as closeness matrix as a measure

Enhanced Random Forest Algorithm:
The RF classifier is a pre-processing algorithm for classification developed by Leo Breiman **[1]** that uses ensemble unpruned classification algorithm. This generates many decision trees. Each decision tree is built using a bootstrap sample of the data, and at each split the candidate set of variables is a random subset of the variables. Thus, RF uses both bagging and boosting as a successful approach for tree building. Random Forest is a combination of tree predictors such that each tree depends on the values of a random vector sampled independently, with the same distribution for all trees in the forests [4]. When new records arrive as input, RF generates a classification and choose the class that is classified by the most trees. It builds many trees, which minimizes the classification errors based on a bootstrap sample of training data from the original dataset using a tree classification algorithm **[2]**. After the forest is created, a replacement object must be classified within its tree. Every tree provides a vote concerning the category of the object. The algorithm chooses the class with the maximum number of votes based on a threshold value selected by the user. By injecting randomness at each node of the grown tree, it has improved accuracy. A new sample is pushed down the tree based on the training sets in the terminal node it ends up at. This procedure is iterated over all the trees in the ensemble as an RF prediction. It is appli in various research fields for modelling. Nowadays, such predictions are used to detect intrusions. **[7]** implemented an RF algorithm to detect the hybrid IDS. The main advantage of using RF is that it can easily handle large and imbalanced datasets more effectively and it does not over-fit. It is excellent for pattern recognition tasks and for detecting attacks.

The additional features of Random Forest:

- It is one of the most accurate learning algorithms available. For many datasets, it produces a highly accurate classifier
- It runs efficiently on large databases
- It can handle thousands of input variables without their deletion
- It generates an internal unbiased estimate of the generalization error as the forest building progresses
- It is an effective method for estimating missing data nd it maintains accuracy when a large proportion of the data are missing.

There are several pre-processing algorithms that are not adaptive when the dataset size is large, which may result in false endorsements. In this research work, a new proposed enhanced Random Forest Algorithm is used – the traditional algorithm is mainly used to find a global optimum solution for a non-linear function. This approach is used to solve the classification problem of the RF classifier that splits the dataset based on a tree classification algorithm [5] and identifies the most important attribute by injecting randomness at each node of the grown tree, reducing the dimensionality of the dataset. Upon pre-processing, the proposed method chooses the most suitable attribute for identifying whether the record is normal or else an attack [6]. The evolutionary analysis consists of four steps for implementation, as follows:

- Representation of individuals
- Initialization of the particles
- Fitness evaluation
- Evolution method

Representation of individuals
The set of all 61 features is represented by a vector of 61 bits, with each bit value represented as either 1 (selected attribute) or 0 (non-selected attribute) from the current set of all attributes.

Initialization of the particles
The particles are initialized by random binary variables. The value of each element in the vector is represented by a random binary variable between 0 and 1. For the experiment, the maximum number of particles be 50 and the maximum iteration be 100. Each particle consists of 41 attributes that are initialized with random binary values based on the presence or absence of the features.

Table 1: Sample attributes initialization.

Attribute Number	1		2	..	..	...	..	61
Binary Value	1		0	0	1	1	0	1

Fitness Evaluation
The fitness value of each attribute is determined using accuracy results from the RF classification algorithm. A feature with a bit value of 1 is selected for calculating the degree of accuracy. If suppose, the bit values to be 111001, this means that Feature 1, Feature 2, Feature 3 and Feature 6 are selected. Based on 1s and 0s, the global best particle will know about the best attribute selected by the RF classifier. In the next generation, the global best particle may be changed based on the position of its 1s, which may reduce the number of attributes as compared with the input set.

Evolution Method:
The proposed Enhanced Random Forest algorithm for reducing the attributes is as follows.

Step 1: Let the population size be p, the maximum generation be MG, the maximum fitness function be MF, and the three predetermined constants be C_w, c_p and c_g, be initialized.

Step 2: Generate a random number R=0 to 1 for d dimension data.

Step 3: Perform the comparison strategy where:
if $(0 \leq R < C_w)$, then $\{x_{nd} = x_{nd}\}$;
Else if $(C_w \leq R < C_p)$, then $\{x_{nd} = p_{nd}\}$;
Else if $(C_p \leq R < C_g)$, then $\{x_{nd} = g_{nd}\}$;
Else if $(C_g \leq R \leq 1)$, then $\{x_{nd} = \text{new } (x_{nd})\}$;

Step4: RF (n, d) =1/exp (x (n, d))
Update (x_{nd});

Step5: Update pbest

Step6: Update gbest

Step7: The process will repeat until the terminated condition is satisfied or the maximum IDR is reached.

Results and Discussion:
The experimental environment is split according to two main steps. The First step, the proposed enhanced algorithm is implemented to reduce the feature set for the IDS. Within its loop, the attributes are validated using the RF classifier. The evolutionary algorithm is repeated over **n** times, where n is initialized as 10 for both the training and testing datasets. The experimental analysis was evaluated for all 50 particles and the iterations were repeated for 100 generations or until the maximum IDR reached. By reducing the number of attributes specified, this may lead to a higher Detection Rate and the reduce alarm rate in relation to anomaly detection. The Enhanced RF reduces the number of attributes more effectively in the dataset, which makes the detection process more secure.

Table 2: Performance comparison between the two methods

Methods	Number of Reduced Attribute (Out of 61 Attributes)
Random Forest	26
Enhanced Random Forest	17

The result of the enhanced Random Forest algorithm is compared with other existing methods as presented in the Table.

Table 3: Sample Reduced features using Enhanced Random Forest Algorithm

S. No	**Attribute**	**Types**
1.	Service	Discrete
2.	Protocol Type	Discrete
3.	Flag Status	Discrete
4.	Size of source in bytes (SRC Bytes)	Continuous
5.	Size of destination in bytes (DST Bytes)	Continuous
6.	Accessed Filed Count (Num_Access_Files)	Continuous
7.	Connection Status (logged in)	Discrete
8.	Host Login Status (is_host_login)	Discrete

The features are consistently detected by a certain algorithm based on three runs out of five, as follows. The features that are detected by traditional methods are 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 16, 17, 18, 19, 20 and 21. Finally, for Enhanced RF, the dominant features are 2, 3, 4, 5, 6, 9, 12, 19, 20, 21 and 22. Consequently, Enhanced RF is considered to be the best evolution algorithm for feature selection compared with Traditional methods.

The result shows the amount of feature convergence based on two methods with respect to the number of iterations. The figure shows that the proposed method is the best algorithm in reducing the number of features compared to existing method.

The Table 3 shows the reduced features of the proposed method. It was observed that the proposed work filters the attributes and reduces the dimensionality of the live dataset. These attributes are further used in the detection engine to analyse intrusive behaviour. It produces near optimal results compared to other intelligent swarm techniques. The main advantage of Enhanced Random Forest is compared to other techniques is that there is no mutation and that crossover is required for processing individual particles, which may reduce its speed.

Present the measurements made in the experiment, compare them with preliminary work or previously published results. In the discussion section you have to relate the results to initial hypotheses.

Conclusion:

The conclusion of this paper is to reduce the number attributes in order to show the efficiency of the intrusion detection analysis framework. An important element of intrusion detection is the intrusion recognition.

This paper has focused on reducing the number of attributes based on Enhanced Random Forest algorithm. This experiment has proved that the proposed Enhanced algorithm reduces the number of attributes more effectively than other existing algorithms and that the results of the proposed algorithm are more consistent and clear.

References:

[1] Breiman L, (2001). "Random forests," *Machine Learning*, Volume 45, pp. 5–32.

[2] Liaw A & Wiener M, (2002). "Classification and regression by random forest", R News 2, pp. 18-22.

[3] Revathi S & Malathi A, (2014). "Network intrusion detection based on fuzzy logic", *International Journal of Computer Applications*, Volume 1, Issue 4, February 2014, pp. 143-149. Available online on http://www.rspublication.com/ijca/ijca_index.htm.

[4] Shai K, & Abbass H.A, (2007). "Biologically inspired complex adaptive systems approaches to network intrusion detection", *Information Security Technical Report*, vol. 12 no. 4, pp. 209–217

[5] Yeh W.C, Chang W.W & Chung Y. Y, (2009). "A new hybrid approach for mining breast cancer patterns using discrete particle swarm optimization and statistical methods", *Expert Systems with Applications*, May, Volume 36, Issue 4, pp. 8204–8211.

[6] Yuk Ying Chung & Noorhaniza Wahid, (2012). "A hybrid network intrusion detection system using simplified swarm optimization (SSO)", *Applied Soft Computing*, Volume 12, pp. 3014–3022.

[7] Zhang J & Zulkernine M, (2005). "Network intrusion detection using random forests", Proceedings of the 3rd Annual Conference on Privacy, Security and Trust (PST), St. Andrews, NB, Canada, pp. 53–61.