ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 3, September 2014

Classification of large datasets using Random Forest Algorithm in various applications: Survey

Mohammed Zakariah
Researcher, King Saud University
College of Computer and Information Sciences, Riyadh, Kingdom of Saudi Arabia

***Abstract:** Random Forest is an ensemble of classification algorithm widely used in much application especially with larger datasets because of its outstanding features like Variable Importance measure, OOB error detection, Proximity among the feature and handling of imbalanceddatasets. This paper discusses many applications which use Random Forest to classify the dataset like Network intrusion detection, Email spam detection, gene classification, Credit card fraud detection, and Text classification. In this paper each application is briefly introduced and then the dataset used for implementation is discussed and finally the real implementation of Random Forest algorithm with steps wise procedure and also the results are discussed. Actual Random Forest Algorithm and its features are also discussed to highlight the main features of Random Forest Algorithm more clearly.*

Keywords: **Random Forest, Data Mining, Classification, Large datasets, Intrusion Detection, Email Spam Detection, Gene Selection, Credit Card Fraud Detection, Text Classification .**

I. INTRODUCTION

In this section the author would like to discuss about the introduction of each application and in section 2 the author discuss about the core Random Forest Algorithm **[1]** and its features followed by in section 3 all the datasets used in each application and in the section 4 implementation of Random Forest Algorithm inn each application with results and in section 5 the author concludes with a conclusion and discussion section.

A. Network Intrusion Detection

Network security is getting more importancebecause of the use of network based technologies and the sensitive information in the network. Many security technologies are developed like Intrusion prevention, information encryption and access control to protect the network based system but still they are not enough to detect many intrusions [2]. To detect the network attacks automatic monitoring of network activities play a vital rule in network security. The two well known intrusion detection techniques are misuse detection and anomaly detection [3]. Because of the significant deviations from the normal activities anomaly is detected [4]. Misuse although cannot detect novel attacks but has low false positive rates. Even though anomaly detection the unknown attacks but has high positive rate, Many hybrid approaches are developed with the intention to get the advantages of both techniques by combining both anomaly and misuse techniques **[4]**, **[5]**, **[6]**. Currently, many IDSs **[7]** Security experts define the rule which highly influence the performance in rule-based systems. Because of the huge amount of traffic the rule of encoding process is expensive and also slows, to impose new rules the security personals have to modify the rule manually using specific rule driven language. Data mining techniques are developed by IDS employees to overcome the limitations of rule based systems. Discovery of understandable patterns and models of large datasets is done by data mining **[8]**. Patterns of misuse and anomaly detections are extracted by data mining techniques which identify activities of normal network and the anomaly classifier to detect the anomaly attack. Flexibility and deploy ability is more in Data mining. It is the matter of only highlighting the label by security experts for auditing data to detect the intrusion instead of hand coding rules for intrusions. Among many data mining techniques Random Forest is the most effective ensemble classification and regression technique. Many different applications have extensively used Random Forest algorithm For instance, it has been applied to prediction **[9]**, **[10]** and probability estimation. In our proposed system, the misuse component uses the random forests algorithm for the classification in intrusion detection, while the anomaly component is based on the outlier detection mechanism of the algorithm.

B. Email Spam Detection

Phishing is one of the different types of frauds committed today. Fraud is and attempt made with the sole intention of personal gain or defaming an individual reputation. Fraud is an actof deceiving people by revealing their personal information for the purpose of financial and personal gain. Extraction of delicate and secret information electronically from users by creating a replica website of an organization. Phishing is done with an electronic device in the computer network and target the detection system from the end users [14, 15]. Email is the main source of phishes who communicate with the well composed messages to users and tempt them to reveal their secrets information’s like bank account and personal info. To gain access to the account. For example, a fraudulent email sent to a user might contain a malware (called man in the browser (MITB)), this malware could be in form of web browser ActiveX components, plugging, or email attachments; if this user ignorantly download this attachment to his pc, themalware will install itself on the user's pc and would in turn transfer money to the fraudster's bank account whenever the user (i.e., the legitimate owner of the bank account) tries to perform an online transaction [14]. Fraudulent activities is on the increase daily; individuals and companies who have been victims in the past now seek for ways to secure themselves from been attacked again. To achieve this, their defense mechanism has to be more secured to prevent them from falling prey again, which implies that the existing defense system (its designs and technology) needs to be greatly improved [16]. Behdad et al. [16] pointed out that improving the defense system is not enough to stop fraudsters as some of them could still penetrate; the system should also be able to identify fraudulent activities and prevent them from occurring. Several traditional approaches used by various email filters today are static in nature; they are not robust enough to handle new and emerging phishing patterns; they only have the ability to handle existing phishing patterns, thus leaving email users prone to new phishing attacks. This is a loop hole because fraudsters are not static in their activities; they change their mode of operation as often as possible to stay undetected.

C. Gene Classification

The study of gene classification is basically extraction of relevant genes for a common task like differentiating the genes based on patients with or without cancer.Biomedical researcher while working for gene selection problem focus on one of the following objectives:

***I). identifying the most relevant genes for Research*:** Extracting the set of genes which are of the interest for the research even if they are of the similar functionality or with correlation.

***II) Identifying the small set of genes*:** Extracting a set genes in smaller quantity and even giving maximum prediction for diagnostic purposes and clinical practices.

The problem of predicting a class in most of the gene expression selection process is done by combining the ranking of genes with a specific classifier. The most complicated task in gene expression classification is the selection of optimal number of genes, based on the simulation study **[22]** some of the preliminary guidelines are although available. Frequently an arbitrarydecision is to the number of genes to retain is made (e.g., keep the 50 best ranked genes and use them with a lineardiscriminate analysis as in **[19, 25]**; keep the best 150 genesas in **[26]**).The above approach is effective if the objective is only to classify few samples but it fails significantly if the objective is to extract the few most influencing genes set from the dataset while retaining the predictive accuracy and prediction performance. (e.g., **[27-29]**), another approach is to apply the same classifier to the smaller set of genes until satisfactory solutions is achieved. The problem of relevant and smaller set of gene selection gets more worst when it comes to multiclass classification solution, as evidence by recent papers in this area(e.g., **[2,8]**). Depending on the above discussion **variable importance** in Random Forest plays a major rule to resolve these issues since the classification system itself describes about the features it has and among them gives the importance of each feature as a variable which itself is required in selecting the most influential gene while giving most predictive accuracy.

D. Credit Card Fraud Detection

Application and behavioral frauds are the two basic types of credit card frauds [38]. Fraudsters access the credit cards from the issuing company by giving wrong information to the company or other peoples information this is called application fraud [39].Stolen/lost card, mail theft, "card holder not present", counterfeit are the methods used by fraudsters in the behavioral frauds. Mail theft usually occurs when the fraudsters intercept the credit card in mail before it reaches the actual card holder or gets the personal information from credit card statements and the bank. Stolen/lost card fraud occurs when the fraudsters get access to the card by theft or to the lost credit card. Because of the increase of online transactions these days there is raise on counterfeit card and "card holder not present" fraud. In the above frauds the credit card details are obtained without the knowledge of the card holder. The following are the ways by which the cardholder information is extracted.

1. Employees stealing information through unauthorized swipe's, phishes, scams, Intrusion into the company network.

2. Fraudulent transactions are done remotely by using carddetails in case of "card holder not present". Internet is the main source for online fraud which allows the fraudsters to commit fraud across the globe with anonymity and also speed.

In **[40]** the credit card fraud evolution over the years is chronicled. In 1970's the most prevalent type of credit card frauds were stolen and forgery type in which credit cards were stolen and used but later in 80's and 90's mail order and phone order became more popular and common.

Because of the use of internet nowa day's more frauds are happening through online transactions called online frauds which giveanonymity, speed and access to commit frauds across the globe. Organizations' and committees are also actively involving apart from the individual committing this type of fraud. Boltan and Hand **[41]** list the literature on credit card and fraud detection and difficult ideas to exchange the potential fraud detection and their innovations. Credit card fraud detection datasets are difficult to get especially for academicians and also fraud detection techniques are not much discussed in the public. **[4]** and **[42]** discusses the challenges in fraud detection with good details. The datasets for fraud detection is combination of categorical and numerical attributes. The amount of transaction is discussed in numerical attributes where as merchant code, name, date of transaction is done by categorical attributes. Depending on the type of datasets these attributes vary from hundreds to thousands of categories. This mix of few numerical and large categorical attributes have spawned the use of a variety of statistical, machine learning, and data mining tools **[4].**

E. Text Classification

Text date is available in larger quantities ever since the invention of internet, database, archives and categorizing these text and classify them into categories has become a challenge with such high dimensionality of text data, scarcity , multi class labels and unbalanced classes . Many classification approaches are developed for categorizing text documents such as Random Forest, supportvector machines (SVM), naïve Bayesian (NB), k-nearest neighbor (KNN), decision tree. Random forest has become a promising classifier technique for text data because of its unique features of classifying largedatasets and algorithmic simplicity and also performance. Text classification is collection of features which are not informative to a particular topic which is also referred as a class. To make the text classifier an informative one with a particular class then each individual tree should be empowered by enhancing the classification accuracy by weighting each feature proposed by Amaratunga**[30].** Depending on the correlation of features the weights are assigned to the class features. Feature weights could be a factor for selecting the features as a subspace. The above discussed principle is similar to Adaboost method **[31]** where the training samples are selected depending on the sample weights processed by the previous classifier and its results. The classification performance of individual trees is gradually getting increased by the above method since the information of each feature contains subspace of decision trees which are biased. In the previous work [54], proposed to use the out-of bag accuracy, this is a property of Random Forest which helps in selecting the most important tree from the forest.

II. RANDOM FOREST CLASSIFIER

A. Variable Importance

Random forests can be used to rank the importance of variables in a regression or classification problem in a natural way. The following technique was described in Breiman's original paper **[1]** and is implemented in the R package randomForest. The first step in measuring the variable importance in a data set $\mathcal{D}_n = \{(X_i, Y_i)\}_{i=1}^{n}$ is to fit a random forest to the data. During the fitting process the out-of-bag error for each data point is recorded and averaged over the forest (errors on an independent test set can be substituted if bagging is not used during training).To measure the importance of the j-th feature after training, the values of the j-th feature are permuted among the training data and the out-of-bag error is again computed on this perturbed data set. The importance score for the j-th feature is computed by averaging the difference in out-of-bag error before and after the permutation over all trees. The score is normalized by the standard deviation of these differences. Features which produce large values for this score are ranked as more important than features which produce small values.

B. Proximity

These are one of the most useful tools in random forests. The proximities originally formed aNxN matrix. After a tree is grown, put all of the data, both training and oob, down the tree. If cases k and n are in the same terminal node increase their proximity by one. At the end, normalize the proximities by dividing by the number of trees.Users noted that with large data sets, they could not fit an NxN matrix into fast memory. A modification reduced the required memory size to NxT where T is the number of trees in the forest. To speed up the computation-intensive scaling and iterative missing value replacement, the user is given the option of retaining only the nrnn largest proximities to each case.When a test set is present, the proximities of each case in the test set with each case in the training set can also be computed. The amount of additional computing is moderate.

C. OOB (Out of Bag Error)

In random forests, there is no need for cross-validation or a separate test set to get an unbiased estimate of the test set error. It is estimated internally, during the run, as follows: Each tree is constructed using a different bootstrap sample from the original data. About one-third of the cases are left out of the bootstrap sample and not used in the construction of the kth tree. Put each case left out in the construction of the kth tree down the kth tree to get a classification. In this way, a test set classification is obtained for each case in about one-third of the trees. At the end of the run, take j to be the class that got most of the votes every time case n was oob. The proportion of times that j is not equal to the true class of n averaged over all cases is the oob error estimate. This has proven to be unbiased in many tests.

D. Features of Random Forests

- ➢ It is unexcelled in accuracy among current algorithms.
- ➢ It runs efficiently on large data bases.
- ➢ It can handle thousands of input variables without variable deletion.
- ➢ It gives estimates of what variables are important in the classification.

- It generates an internal unbiased estimate of the generalization error as the forest building progresses.
- It has an effective method for estimating missing data and maintains accuracy when a large proportion of the data are missing.
- It has methods for balancing error in class population unbalanced data sets.
- Generated forests can be saved for future use on other data.
- Prototypes are computed that give information about the relation between the variables and the classification.
- It computes proximities between pairs of cases that can be used in clustering, locating outliers or (by scaling) give interesting views of the data.
- The capabilities of the above can be extended to unlabeled data, leading to unsupervised clustering, data views and outlier detection.
- It offers an experimental method for detecting variable interactions.

III. DATASETS

A. Network Intrusion Detection

Experiments were carried out with on the Knowledge Discovery and Data Mining 1999 (KDD'99) dataset. The limitations of the KDD'99 datasets are inherited from the Defense Advanced Research Projects Agency (DARPA) datasets **[11].** However, these are the most comprehensive and widely used datasets that can be employed to compare and contrast with other related IDSs. These datasets also do not require any further time-consuming preprocessing.

B. Email Spam Detection

For the implementation and testing of our machine learning algorithm, we used two publicly available datasets. We got our ham mails from the ham corpora provided by spam assassin project [32], and our phishing emails were gotten from the publicly available phishing corpus [33] provided by Nazario. All the emails coming from the ham corpora were labeled as ham emails and the emails coming from the phishing corpora waslabeled as phishing email.

C. Gene Classification

All simulations and analyses were carried out with R [43], using packages RandomForest for random forest, the microarray and simulated data sets are available from the supplementary material web page [44].

D. Credit Card Fraud Detection

In this study we use the dataset of [55], which was obtained from an international credit card operation. This dataset has 13 months, from January 2006 to January 2007, of about 50 million (49,858,600 transactions) credit card transactions on about one million (1,167,757 credit cards) credit cards from a single country.

E. Text Classification

Six real world text data sets were used. These text data sets are selected due to their diversities in number of features, data volume and number of classes. Their dimensionalities vary from 2000 to 8460, the numbers of documents vary from 918 to 18772, and the minority category rates vary from 0.32% to 6.43%.The Fbis, Re0, Re1, Oh5, and Wapdatasets are classical text document classification benchmark data which have been carefully selected a preprocessed by Han and Karypis[45]. The data set Fbisis from the Foreign Broadcast Information Service data of TREC-5 [46]. The data sets Re0 and Re1 are from Reuters-21578 text categorization test collection Distribution 1.0 [47]. The data set Oh5 is from OHSUMED-233445 collection [48]. The data set Wapis from the WebACE project (WAP) [49]. Newsgroups data set is a popular text corpus for experiments in text applications of machine learning techniques. It was obtained from 20 different Usenet newsgroups and contains 18772 documents divided into 20 different classes [50]. We preprocess this data by removing stop terms, and therefore, kept 5000 most informative terms as distinct features of the dataset.

IV. IMPLEMENTATION OF RANDOM FOREST

A. Network Intrusion Detection

We propose new systematic frameworks that apply a data mining algorithm called random forests in misuse, anomaly, and hybrid-network-based IDSs. In misuse detection, patterns of intrusions are built automatically by the random forests algorithm over training data. After that, intrusions are detected by matching network activities against the patterns. In anomaly detection, novel intrusions are detected by the **outlier detection** mechanism of the random forests algorithm. After building the patterns of network services by the random forests algorithm, outliers related to the patterns are determined by the outlier detection algorithm.KDD'99 dataset is used, which was preprocessed by extracting41 features from the tcpdump data in the 1998 DARPAdatasets **[11]**, **[13]**. It includes the full training set, the 10% trainingset, and the test set. The full training set has 4 898 431 connections the task of the KDD'99contest was to build a classifier capable of distinguishing betweenfour kinds of intrusions and normal traffic numbered asone of the five classes: normal, probe, DoS, U2R, and R2L.

i) ***Performance Comparison on Balanced and Imbalanced Dataset***: The original date set is imbalanced to make it balanced training set down sampling is done for Normal and DoS classes by randomly selecting 10% of connections belonging to normal and DoS from the original dataset. We also oversample U2R and R2L by replicating theirconnections. The balanced training set with 60 620 connectionsis much smaller than the original one.The first experiment is to compare the original training data used to build the pattern and the balances training set with sampling. The default values of the parameters are used to carry out the experiment on Random Forest Algorithm in WEKA **[12]**. 66% samples as training data, 34% samplesas test data, ten trees in the forest, and six random featuresto split the nodes. The comparison between the balanced dataset and the original dataset is the main purpose of this experiment to detect the performance. The results reveal that the sampling technique for balancing the dataset class significantly improves the performance and also reduces the time spent to build the pattern.

The default values of the parameters are used because of the convenience for both the datasets. Performance is improved by sampling technique especially for minority classes and also the reduction of time to build the pattern.

ii) Selection of Important Features**:** The second experimentis to select the most important features. Random Forest has a unique mechanism to calculate the importance of the features. RF calculates and selects the most relevant and influential features. Depending on the number of votes counted for the correct class using oob cases in every tree the estimation of variable importance is calculated.There are 41 features inthe KDD’99 dataset numbered from 1 to 41.We employ the featureselection algorithm supported by the random forests algorithmto calculate the value of variable importance.After randomly permuting the values of variable m the correct votes is again counted, the average between these two numbers is the raw importance score for the variable m. To get the z-score the raw score is divided by its standard error and the value ofvariable importance is the negative z-score for variable m. Therefore to build the pattern we select the remaining 38 features as the most important. Feature 3 (service type such as http, telnet, and ftp) is the most important feature to detect intrusions. Service types are most sensitive to intrusions. Feature 7(land) is used to check if from/to are connected to the same host.According to the domain knowledge; it is the most discriminatingfeature for land attacks. However, land attacks cause DoS,and they have much fewer connections than other types of DoS.

iii) Parameter Optimization**:** Optimizing the number of random features improves the detection rate. Balanced datasets are used to build the forest using different Mtry and then the time to built the patterns corresponding to different Mtry and the oob error are plotted. When Mtry is 15,25,30 the obb error comes to minimum and when the Mtry is increased the time to built the pattern also increases. Thus, we choose 15 asthe optimal value, which reaches the minimum of the oob errorrate and costs the least time among these three values.

B. Email Spam Detection

The features used for the email classification are described in this section. These features were identified from different literature; combination of these features together forms a feature set that effectively classified emails into phishing and no phishing. A group of 15 features frequently used by phishing attackers was identified from different literature and used in this paper. Although the features set are few (compared to some filters that used hundreds of features for detection), a high accuracy was still achieved. These features are described in the remaining part of this section.

- URLs Containing IP Address.
- Disparities between “href ” Attribute and LINK Text.
- Presence of “Link,” “Click,” and “Here” in Link Text of a Link.
- Number of Dots in Domain Name.
- HTML Email.
- Presence of Javascript.
- Number of Links.
- Number of Linked ToDomain.
- From Body MatchDomain Check.
- Word List Features.

In this work, we trained and tested our classifier using 10-fold cross validation. In 10-fold cross validation, the dataset is divided into 10 different parts; 9 of the 10 parts are used to train the classifier and the information gained from the training phase would be used to validate (or test) the 10^{th} part; this is done 10 times, such that, at the end of the training and testing phase, each of the parts would have been used as both training and testing data. This method (i.e., cross validation method) ensures that the training data is different from the test data. In machine learning, this method is known to provide a very good estimate of the generalization error of a classifier. Machine learning involves two major phases: the training phase and the testing phase. The predictive accuracy of the classifier solely depends on the information gained during the training process; if the information gained (IG) is low, the predictive accuracy is going to be low, but if the IG is high, then the classifier’s accuracy will also be high. As stated above, we used 10-fold cross validation. In our random forest classification, before the decision trees are constructed, the information gained for all the 15 features is calculated (using the IG method explained by Mitchell **[17]**) and the features with the best eight IG are selected and used for constructing the decision trees; the mode vote (from all the trees) is then calculated and used for the email prediction. Information gain is one of the feature ranking metric highly used in many text classification problems today. More details about our algorithm are described in the next section below.

```
Begin RF Algorithm
    Input:  $N$: number of nodes
    $M$: number of features
    $D$: number of trees to be constructed
    Output: $V$: the class with the highest vote
While stopping criteria is false do
```

```
    Randomly draw a bootstrap sample A from the training data $D$
    Use the steps below to construct tree $Ti$ from the drawn bootstrapped sample A:
        (I) randomly select $m$ features from$M$; where$m \ll M$
        (II) For node d, calculate the best split point among the $m$ features
        (III) Split the node into two daughter nodes using the best split
        (IV) Repeat I, II and III until $n$ number of nodes has been reached
    Build your forest by repeating steps I–IV for $D$ number of times
EndWhile
    Output all the constructed trees $\{Ti\}1D$
    Apply a new sample to each of the constructed trees starting from the root node
    Assign the sample to the class corresponding to the leaf node.
    Combine the decisions (or votes) of all the trees
    Output $V$, that is, the class with the highest vote.
End RF Algorithm
```

We tested our method using varied dataset sizes this was done to know the performanceof the algorithm on both small and large datasets. The algorithm performed best when tested on the dataset that has the largest size (having an overall accuracy of 99.7%, FN rate of 2.50%, and FP rate of 0.06%); this implies that ourmethodwill work effectively if applied to real world dataset, which is usually large in size. Our method also achieved a higher prediction accuracy (99.7%) compared to an accuracy of 97% achieved by Fette et al. **[18]**.

C. Gene Classification

Variable Importance is calculated by Random Forest with several measures. The most influencing measure is the decrease in classification accuracy when the values of a variable in the node of a tree is randomly permuted **[13, 36]** this measure is named as variable importance measure. Random Forests are iteratively fitted to select the genes and at each iteration a new forest is built by removing those variables with smaller importance measure. By doing the above process the ultimate and final selected genes would yield the minimum OOB error. Because of the iterative approach, the OOB error is biased down and cannot be used to access the overall error rate of the approach, for reasons analogous to those leading to "selection bias" **[34,37]**. To assess prediction error rates we will use the bootstrap, not OOB error (see above). (Using error rates affected by selection bias to select the optimal number of genes is not necessarily a bad procedure from the point of view of selecting the final number of genes; see **[38]**). In this **algorithm** the authors examine all forests that result from eliminating, iteratively, a fraction, fraction. Dropped, of the genes (the least important ones) used in the previous iteration. By default, fraction. Dropped= 0.2 which allows for relatively fast operation, is coherent with the idea of an "aggressive variable selection" approach, and increases the resolution as the number of genes considered becomes smaller. We do not recalculate variable importance's at each step as **[26]** mention severe over fitting resulting from recalculating variable importance's. After fitting all forests, we examine the OOB error rates from all the fitted random forests. We choose the solution with the smallest number of genes whose error rate is within u standard errors of the minimum error rate of all forests. Setting u = 0 is the same as selecting the set of genes that leads to the smallest error rate. Setting u = 1 is similar to the common "1 s.e. rule", used in the classification trees literature **[14,15]**; this strategy can lead to solutions with fewer genes than selecting the solution with the smallest error rate, while achieving an error rate that is not different, within sampling error, from the "best solution". and can achieve the objective of aggressively reducing the set of selected genes. Results for the real data sets ntree= {2000, 5000, 20000}, mtryFactor= {1, 13}, se = {0, 1}, fraction. dropped = {0.2, 0.5}).The number of genes selected varies by data set, but generally the variable selection procedure leads to small (< 50) sets of predictor genes, often much smaller than those from competing approaches. There are no relevant differences in error rate related to differences in mtry, ntreeor whether we use the "s.e. 1" or "s.e. 0" rules. The use of the "s.e. 1" rule, however, tends to result in smaller sets of selected genes.

D. Credit Card Fraud Detection

Training data is imbalanced and are sampled with two classes with reasonable proportions for fraud and non fraud cases. Random under sampling of majority class is better than the other sampling approaches. Random under sampling is used as training dataset with some proportion for fraud and non fraud cases. Many other algorithms are used to detect the performance with four training datasets having 15%, 10%, 5% and 2% fraudulent transactions. These are labeled DF1, DF2, DF3, and DF4 in the results. Test dataset with 0.5% fraudulent transaction is observed to detect the performance. Dataset A has 2420 observed fraudulent transactions. We divided dataset A into two subsets of1237 (51%) and 1183 (49%) transactions. The four modeling datasets (DF1, DF2, DF3, and DF4) are populated by using the first set of 1237 fraudulent transactions and similarly for populating the test dataset the second set of 1183 transactions are used. We sampled legitimate transactions from dataset C to create varying fraud rates in the modeling and test datasets. In other words, we kept the same number of fraudulent transactions in the four modeling datasets, but varied the number of legitimate transactions from dataset C to create varying fraud rates. Similarly, the actual fraud rates in the test dataset are 0.5%.Several measures of classification performance commonlynoted in the literature are used. Sensitivity and specificity measure the accuracy on the positive (fraud) and negative (non-fraud) cases. A tradeoff between these true positives and true negatives is typically sought. The F-measure giving the harmonic mean of precision andrecall, G-mean giving the geometric mean of fraud and non-fraud accuracies, and weighted-Accuracy provide summary performance indicators of such tradeoffs. The various performance measures are defined with respect to the confusion matrix below, where Positive corresponds to Fraud cases and Negative corresponds to non-fraud cases.

- Accuracy (TP+TN)/(TP+FP+TN+FN)
- Sensitivity (or recall) TP/(TP+FN) gives the accuracy on the fraudcases.
- Specificity TN/(FP+TN) gives the accuracy on the non-fraud cases.
- Precision TP/(TP+FP) gives the accuracy on cases predicted asfraud.
- F-measure 2 Precision Recall/(Precision+Recall).
- G-mean (Sensitivity Specificity)0.5.
- wtdAcc w Sensitivity+(1−w) Specificity; we use w=0.7 toindicate higher weights for accuracy on the fraud cases.

The above measures arising from the confusion matrix are based on a certain cutoff value for class-labeling, by default generally taken at 0.5. We also consider the AUC performance measure, which is often considered a better measure of overall performance **[53]**. Performance measures like AUC, however, give equal consideration to false positives and false negatives and thus do not provide a practical performance measure for fraud detection **[52]**. Where cost information is available, these can be incorporated into a cost function to helpassess performance of different models. We report on the multiple measures described above to help provide a broad perspective on performance and since the impact of sampling can vary by technique and across performance measures **[51]**.In evaluating credit card fraud detection, where non-fraud cases tend to dominate in the data, a high accuracy on the (minority class) fraud cases is typically sought. Accuracies on fraud and no fraud cases are shown through sensitivity and specificity, and these, together with precision can indicate desired performance characteristics. In implementation, a fraud detection model will be used to score transactions, with scores indicating a likelihood of fraud. Scored cases can be sorted indecreasing order, such that cases ranked towards the top have higher fraud likelihood. A well performing model here is one that ranks most fraud cases towards the top. With a predominance of non-fraud cases in the data, false positives are only to be expected. Model performance can be assessed by the prevalence of fraudulent cases among the cases ranked towards the top. The traditional measures described above do not directly address such performance concerns that are important in fraud management practice.

E. Text Classification

We present an improved random forest algorithm by simultaneously taking into account of a new feature weighting method and the tree selection method to categorize text documents. This algorithm can effectively reduce the upper bound of the generalization error and improve classification performance. From the results of two experiments on various text data sets, the random forest generated by this new method is superior to other text categorization methods.

Algorithm 1. Improved Random Forest Algorithm

```
Input:
  - D: the training data set,
  - A: the feature space {A1, A2,...,AM},
  - Y: the feature space {y1, y2,...,yq},
  - K: the number of trees,
  - m: the size of subspaces.
Output: A random forest μ
  Method:
  for i=1 to K do
    draw a bootstrap sample in-of-bag data subset IOBiand out-of-bag data subset OOBifrom the training data set D;
    hi(IOBi) = createTree(IOBi);
    use out-of-bag data subset OOBito calculate the out-ofbag accuracyOOBAcciof the tree classifier hi(IOBi) by Equation (4);
  end for
  sort all K trees classifiers in their OOBAccdescending order;
  select the top 70% trees with high OOBAccvalues and combine the 70% tree classifiers into an improved random forest μ;
  Function createTree()
  create a new node η;
  if stopping criteria is met then
    returnηas a leaf node;
  else
    for j=1 to j=M do
      compute the informativeness measure corr(Aj,Y) by Equation (1);
    end for
    compute feature weights {w1, w2,...,wM} by Equation (3);
    use the feature weighting method to randomly select m features;
    use these m feature as candidates to generate the best split for the node to be partitioned;
    call createTree() for each split;
  end if
  return η;
```

In this algorithm, input parameters are the training data set, the feature space, the class feature, the number of trees in the random forest and the size of subspaces. The output is a random forest model. Steps 1-5 are the loop for building K decision trees. In the loop, Step 2 samples the training data with the bootstrap method to generate an in-of-bag data subset for building a tree classifier, and generate an out-of-bag data subset for testing the tree classifier on out-of-bag accuracy. Step 3 calls the recursive function createTree() to build a tree classifier. Step 4 uses out-of-bag data subset to calculate the out-of bag accuracy of the tree classifier. After the loop, Step 6 sorts all built tree classifiers in their out-of-bag accuracies in descending order. Step 7 selects the top 70% trees with high out-of-bag accuracy values and combines the 70% tree classifiers into an improved random forest model. In practice, 70% is sufficiently enough to obtain good results. Function **createTree**first creates a new node. Then, it tests the stop criteria to decide whether to return to the upper node or to split this node. If splitting this node, it uses the feature weighting method to randomly select m features as a subspace for node splitting. These features are used as candidates to generate the best split to partition the node. For each subset of the partition, **create Tree**is called again to create a new node under the current node. If a leaf node is created, it returns to the parent node. This recursive process continues until a full tree is generated. Compared with Breiman's method, there are two changes for building a random forest model. The first change is the way to select the feature subspace at each node. Breiman uses simple random sampling method. For very high dimensional text data, the subspace must be set large in order to contain informative feature. This will increase computation burden. With the feature weighting method, we can still use Breiman's formula $2 \lfloor \log (M) +1 \rfloor$to specify the subspace size. The second change is that tree selection method is added. This method is further optimizing random forest model.

V. CONCLUSION AND DISCUSSIONS

Random Forest Algorithm is used to classify large datasets in various applications. In the above survey paper various applications are discussed and how Random Forest Algorithmis implemented on each application and the results are also discussed and also the datasets used in each application and its description is done. Random Forest has many properties which influence the classification results some of the features are Variable Importance detection and Outlier and OOB error and Proximity detection and working with imbalanced dataset, each property of Random Forest is used in different applications like for detection of intrusion outlier property is used, and for selecting the most relevant gene among in gene expression micro array datasets variable importance measure is used by iterating the Random Forest to remove the least significant gene until most significant gene is extracted and in text classification proximity is used to detect the correlation between the text in the dataset. Random Forest is of great use especially for large datasets as it is ensemble classification technique in data mining which makes many trees and each tree is developed with a set os samples selected in random and each node has a set of features to select the most important feature to break the node in to two depending on the information gain received. This paper is written with the intention to help the researchers get an idea about the importance of Random Forest algorithm and how it is used in various applications and how it could b further implemented on other application based on the above discussion. Random Forest in basically written in FORTRAN but later on R programming is also used to implement Random Forest Algorithm. This paper also gives references to the implementation steps followed by many researchers like what are the programming language and the platform to do the programming and the datasets used in the classification. The author of this paper hopes to assist the researchers in further implementing new applications using Random forest algorithm following the above applications.

REFERENCES

[1] Breiman, Leo (2001). "Random Forests". Machine Learning 45 (1): 5–32.Doi: 10.1023/A: 1010933404324.

[2] CSI/FBI Computer Crime and Security Survey. (2004). Computer Security Inst., San Francisco, CA. [Online]. Available: http://www.issa-sac.org/docs/FBI2004.pdf.

[3] D. Barbara and S. Jajodia, Applications of Data Mining in Computer Security. Norwell, MA: Kluwer, 2002.

[4] E. Eskin, A. Arnold, M. Prerau, L. Portnoy, and S. Stolfo, "A geometric framework for unsupervised anomaly detection: Detecting intrusions in unlabeled data," in Applications of Data Mining in Computer Security. Norwell, MA: Kluwer, 2002.

[5] D. Anderson, T. Frivold, and A. Valdes, "Next-generation intrusion detection expert system (NIDES)—A summary," SRI Int., Menlo Park, CA, Tech. Rep. SRI-CSL-95-07, May 1995.

[6] D. Barbara, J. Couto, S. Jajodia, L. Popyack, and N. Wu, "ADAM: Detecting intrusions by data mining," in Proc. 2nd Annu. IEEE Workshop Inf. Assur. Secur., New York, Jun. 2001, pp. 11–16.

[7] E. Tombini, H. Debar, L. Me, and M. Ducasse, "A serial combination of anomaly and misuse IDSes applied to HTTP traffic," in Proc. 20th Annu. Comput. Secur. Appl. Conf., Tucson, AZ, Dec. 2004, pp. 428–437.

[8] Snort Network Intrusion Detection System. (2006). [Online]. Available: http://www.snort.org.

[9] D. Hand, H. Mannila, and P. Smyth, Principles of Data Mining. Cambridge, MA: MIT Press, Aug. 2001.

[10] L. Guo, Y. Ma, B. Cukic, and H. Singh, "Robust prediction of fault proneness by random forests," in Proc. 15th Int. Symp. Softw. Rel. Eng.(ISSRE), Brittany, France, Nov. 2004, pp. 417–428.

[11] DARPA Intrusion Detection Evaluation. (2006). [Online]. Available: http://www.ll.mit.edu/IST/ideval/.

[12] WEKA software. (2006). [Online]. Available: http://www.cs.waikato. ac.nz/ml/weka/.

[13] C. Elkan, "Results of the KDD'99 classifier learning," SIGKDD Explorations, vol. 1, no. 2, pp. 63–64, 2000.

[14] M. Khonji, Y. Iraqi, and A. Jones, "Phishing detection: a literature survey," IEEE Communications & Surveys Tutorials, vol. 15, no. 4, pp. 2091–2121, 2013.

[15] S. Sheng, M. Holbrook, P. Kumaraguru, L. F. Cranor, and J. Downs, "Who falls for phish?: a demographic analysis of phishing susceptibility and effectiveness of interventions," in Proceedings of the 28th Annual SIGCHI Conference on Human Factors in Computing Systems (CHI '10), pp. 373–382, Atlanta, Ga, USA, April 2010.

[16] M. Behdad, L. Barone, M. Bennamoun, and T. French, "Nature inspired techniques in the context of fraud detection," IEEE Transactions on Systems, Man, and Cybernetics C: Applications and Reviews, vol. 42, no. 6, pp. 1273–1290, 2012.

[17] T. M.Mitchell, Machine Learning, McGraw-Hill, NewYork, NY, USA, 1997.

[18] I. Fette, N. Sadeh, and A. Tomasic, "Learning to detect phishing emails," in Proceedings of the 16th International World Wide Web Conference (WWW '07), pp. 649–656, Alberta, Canada, May 2007.

[19] Lee JW, Lee JB, Park M, Song SH: An extensive evaluation of recent classification tools applied to microarray data. Computation Statistics and Data Analysis 2005, 48:869-885.

[20] Yeung KY, Bumgarner RE, Raftery AE: Bayesian model averaging: development of an improved multi-class, gene selection and classification tool for microarray data. Bioinformatics 2005, 21:2394-2402.

[21] Jirapech-Umpai T, Aitken S: Feature selection and classification for microarray data analysis: Evolutionary methods for identifying predictive genes. BMC Bioinformatics 2005, 6:148.

[22] Hua J, Xiong Z, Lowey J, Suh E, Dougherty ER: Optimal number of features as a function of sample size for various classification rules. Bioinformatics 2005, 21:1509-1515.

[23] Li Y, Campbell C, Tipping M: Bayesian automatic relevance determination algorithms for classifying gene expression data. Bioinformatics 2002, 18:1332-1339.

[24] Díaz-Uriarte R: Supervised methods with genomic data: a review and cautionary view. In Data analysis and visualization in genomics and proteomics Edited by: Azuaje F, Dopazo J. New York: Wiley; 2005:193-214.

[25] Dudoit S, Fridlyand J, Speed TP: Comparison of discrimination methods for the classification of tumors suing gene expression data. J Am Stat Assoc2002, 97(457):77-87.

[26] Li T, Zhang C, Ogihara M: A comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression. Bioinformatics 2004, 20:2429-2437.

[27] van't Veer LJ, Dai H, van de Vijver MJ, He YD, Hart AAM, Mao M, Peterse HL, van der Kooy K, Marton MJ, Witteveen AT, Schreiber GJ, Kerkhoven RM, Roberts C, Linsley PS, Bernards R, Friend SH: Gene expression profiling predicts clinical outcome of breast cancer.Nature 2002, 415:530-536.

[28] Roepman P, Wessels LF, Kettelarij N, Kemmeren P, Miles AJ, Lijnzaad P, Tilanus MG, Koole R, Hordijk GJ, van der Vliet PC, Reinders MJ, Slootweg PJ, Holstege FC: An expression profile for diagnosis of lymph node metastases from primary head and neck squamous cell carcinomas. Nat Genet 2005, 37:182-186.

[29] Furlanello C, Serafini M, Merler S, Jurman G: An accelerated procedure for recursive feature ranking on microarray data. Neural Netw2003, 16:641-648.

[30] D. Amaratunga, J. Cabrera and Y.S. Lee, "Enriched Random Forests," Bioinformatics, vol. 24, no. 18, pp.2010-2014, 2008.

[31] Y. Freund and R.E. Schapire, "Experiments with a new boosting algorithm," in Proc. of the 13th InternationalConference on Machine Learning, pp.148-156, 1996.

[32] Apache Software Foundation, "Spam assassin homepage," 2006, http://spamassassin.apache.org/.

[33] J. Nazario, "Phishingcorpus homepage," 2006, http://monkey.org.

[34] Ambroise C, McLachlan GJ: Selection bias in gene extraction on the basis of microarray gene-expression data. ProcNatlAcadSci USA 2002, 99(10):6562-6566.

[35] Bureau A, Dupuis J, Hayward B, Falls K, Van Eerdewegh P: Mapping complex traits using Random Forests. BMC Genet 2003, 4(Suppl 1):S64.

[36] Simon R, Radmacher MD, Dobbin K, McShane LM: Pitfalls in the use of DNA microarray data for diagnostic and prognostic classification. Journal of the National Cancer Institute 2003, 95:14-18.

[37] Braga-Neto U, Hashimoto R, Dougherty ER, Nguyen DV, Carroll RJ: Is cross-validation better than resubstitution for ranking genes? Bioinformatics 2004, 20:253-258.

[38] R.J. Bolton, D.J. Hand, Unsupervised profiling methods for fraud detection, Conference on Credit Scoring and Credit Control, Edinburgh, 2001.

[39] CapitalOne Identity theft guide for victims, retrieved January 10, 2009, from http://www.capitalone.com/fraud/IDTheftPackageV012172004We.pdf?linkid=WWW_Z_Z_Z_FRD_D1_01_T_FIDTP.

[40] K. Williams, The Evolution of Credit Card Fraud: Staying Ahead of the Curve, eFunds Corporation, 2007.

[41] R.J. Bolton, D.J. Hand, Statistical fraud detection: a review, Statistical Science 17 (3) (2002) 235–249.

[42] F. Provost, Comment on Bolton and Hand, Statistical Science 17 (2002) 249–251.

[43] R Development Core Team: R: A language and environment for statistical computing. 2004 [http://www.R-project.org]. R Foundation for Statistical Computing, Vienna, Austria [ISBN 3-900051-00- 3].

[44] [http://ligarto.org/rdiaz/Papers/rfVS/randomForestVarSel.html].

[45] E. Han and G. Karypis, "Centroid-based document classification: Analysis & experimental results," in Proc. of the 4th European Conference on Principles of Data Mining and Knowledge, pp. 424-431, 2000.

[46] TREC. Text Retrieval conference. Available at: http://trec.nist.gov.

[47] D.D. Lewis, Reuters-21578 text categorization test collection distribution 1.0, Available at: http://www.research.att.com/~lewis.

[48] W. Hersh, C. Buckley, T.J. Leone and D. Hickam, "OHSUMED: An interactive retrieval evaluation and new large test collection for research," in Proc. of the 17th annual international ACM SIGIR conference on research and development in information retrieval, pp. 192-201, 1994.

[49] J. Moore, E. Han, D. Boley, M. Gini, R. Gross, K. Hastings, G. Karypis, V. Kumar and B. Mobasher, "Web page categorization and feature selection using association rule and principal component clustering," in Proc. of the 7th Workshop on Information Technologies and System, 1997.

[50] J. Rennie, Available at: http://people.csail.mit.edu/jrennie/20Newsgroups/20newsbydate-matlab.tgz.

[51] F. Provost, T. Fawcett, Analysis and visualization of classifier performance: comparison under imprecise class and cost distributions, Proceedings of the Third International Conference on Knowledge Discovery and Data Mining, 1997, pp. 43–48.

[52] C. Paasch, Credit Card Fraud Detection Using Artificial Neural Networks Tuned by Genetic Algorithms, Hong Kong University of Science and Technology (HKUST), Hong Kong, Doctoral Dissertation, 2007.

[53] C. Whitrow, D.J. Hand, P. Juszczak, D. Weston, N.M. Adams, Transaction aggregation as a strategy for credit card fraud detection, Data Mining and Knowledge Discovery 18 (1) (2009) 30–55.

[54] B.X. Xu, J.J. Li, Q. Wang and X.J. Chen, "A Tree Selection Model for Improved Random Forest," in Proc. Ofthe International Conference on Knowledge Discovery, pp.382-386, 2011.

[55] C. Paasch, Credit Card Fraud Detection Using Artificial Neural Networks Tuned by Genetic Algorithms, Hong Kong University of Science and Technology (HKUST), Hong Kong, Doctoral Dissertation, 2007.