Correlation-Based Feature Grouping with Decision Tree for Classifying High-Dimentional Imbalanced Data

Md. Abu Darda
ID: 011122005
Md. Arbab Khan Panni
ID: 011131024
Md. Bakhtiar Islam
ID: 011132162
Md. Abdullah Al Mamun
ID: 011132060
Department of Computer Science and Engineering
United International University

A thesis submitted for the degree of
BSc in Computer Science & Engineering

August 2018

Abstract

Classifying high-dimensional imbalanced data is a big challenge in mining real-world big data. Existing algorithms are classifying the majority class instances and get the maximum classification accuracy and minority class instance is overpowered by getting misclassified. In real life applications minority class instances are more significant than the majority class. For classifying imbalanced data sets few techniques based on sampling (Under-sampling / over-sampling), cost sensitive learning methods and ensemble learning are used. In our research, A new technique has been introduced, "correlation-based feature grouping with decision tree for classifying high-dimensional imbalanced data".We have assessed the dispatch of the the proposed algorithm on few of the high dimensional imbalanced data sets with different imbalance correspondences. The results are tremendously better to work with high imbalanced data sets.

We are devoting this thesis to our parents.

Acknowledgements

We have finished our thesis work under the supervision of one of the finest faculty of United International University (UIU). He is our beloved Dr. Dewan Md. Farid, Associate Professor, who was our academic supervisor. His positive efforts towards us and pushing us to our limits were the only reason that we have made it till here. We are thankful to Prof.Dr. Hasan Sarwar, Prof. Dr. Salekul Islam (Head of the Department,CSE), Prof. Dr. Khondoker Abdullah Al Mamun, Dr. Swakkhar Shatabda, Associate Professor and Undergraduate Program Coordinator, and some other teachers and faculty members who are effortlessly working to enlightening our Computer Science and Engineering department. We would also want to thank our loving United International University (UIU). We are really lucky that we had chance to be a part of this United International University (UIU) family. We have been provided the best environment and all the other necessary supports to finish our journey.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

A tremendous change in science of modern times are leading us to the all kind of data sets like world wide web healthcare and similar scientific sectors. These kinds of large data opens the opportunity for discovering the knowledge and it is an important part in a huge range of applications from our everyday activities to all kind of industrial decisions making applications. It was always a big help on data mining sector to make it a fast growing area in our age. To identify useful trends of data for processing structured/ unstructured data mining and machine learning methods are strongly capable. Useful patterns can be discovered by supervised learning or unsupervised learning techniques.

A very recent problem that came to attention in data mining application is class imbalance problem. In real world, class imbalance data sets like software prediction, oil spill detection, fraudulent transaction detection finding a rare disease, the minority class instances are overlooked [1–3]. But these minority class instances are representing a significant interest than the majority class instances [4].

There are 3 ways to solve class imbalance problem:

1. sampling method
2. ensemble method
3. cost-sensitive learning method

In sampling techniques (under sampling/ oversampling), we can remove the majority class instances from the imbalanced datasets or add the minority classes instances into imbalanced dataset to get a better and balanced dataset [5]. In Ensemble technique,

bagging and boosting are used for classifying imbalanced datasets. In ensemble method, sampling technique is used in each iteration. Cost-sensitive learning is used for solving the imbalance problems. Different costs are assigns based on the misclassification error of classes [6]. Usually for minority class, high cost is allocated. But the classification consequence are not substantial in cost sensitive learning methods.
Handling imbalanced classification problems can be explain in two categories:

(i) External methods.
(ii) Internal methods.

External method is referred to balancing methods and it is processes the imbalance datasets to get a balanced data [7]. Existing learning algorithm are modified by internal methods which reduce the sensitiveness to the class imbalance while culturing it self from imbalanced data.

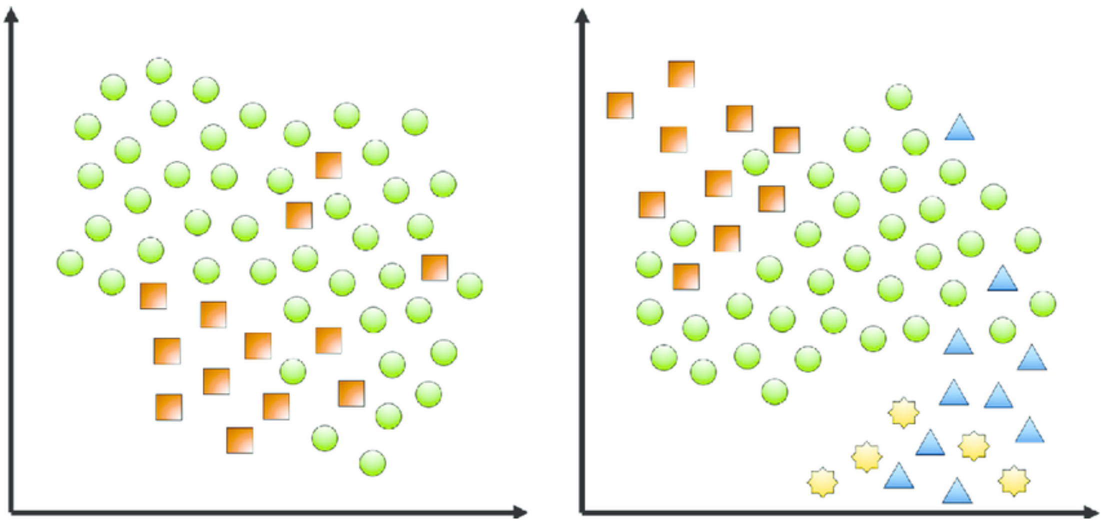

Figure 1.1: Class Imbalanced data.

1.1 Machine Learning

Artheur samuel came with the term machine learning at first. Various analysis and creation of algorithm which ease the creation of data operate conclusion from hitherto furnished data is dealt by machine learning [8]. The provided datas are also known as training data.

1.1.1 Supervised Learning

The process for identifying new or unrevealed instances by training a group of samples with known class values is supervised learning. Each example of the training data has series of attributes in the form of a vector and labeled as belonging to a certain class. Predictive model is the creation from training data. Supervised algorithm is divided into two categories:

Classification algorithm: When the model is trained to predict class labels.
Examples:

a. Decision trees [9]
b. Random forests [10]
c. Support vector machines (svm) [11]
d. Neural Networks [12]
e. k-Nearest-Neighbors (KNN) [13]
f. Naïve Bayes [14]

Regression algorithms: When the model is trained to predict new values for data where the data is continuous.
Example:

a. Linear regression [15]
b. Multivariate regression [16]
c. Regression trees [17]
d. Lasso regression [18]
e. Logistic regression [19]

.

1.1.2 Unsupervised Learning

Unsupervised learning consists of machine learning algorithm where various techniques are used to input the data to get new meaningful patterns and create groups of data instances.

It is divided in two subcategories.

- Clustering algorithms: Unlabeled data are used to create groups or cluster of different classes. Dividing data is based on mean, medoids, hierarchies and others.
  Example:

  a. k-means
  b. k-medoids
  c. Hierarchical clustering
  d. Fuzzy c-mean
  e. Gaussain clustering
  f. Density based clustering

- Association rule learning algorithms: These use features of the given data to mine rules and patterns from the datasets and it explain the relationships between different attributes.
  Example:

  a. Apriori algorithm
  b. FP-Growth algorithm
  c. Eclat algorithm

1.2 Class imbalance problem

It is well aware, that nearly all of the real world applications are based on imbalanced datasets. Nearly all crucial information are clasp by minority classes and it guides to huge trouble by getting misclassified.
Class imbalance occurs in a manner that 1 in 100 instances is a minority class instances. So while classifying the minority class instances it will still have an accuracy of 99%. This is the reason the classifier is not always sufficient to choose the performance of a classifier [20].
Two important problems assumptions based on traditional classification:

1. Maximize the precision or minimize error rate is the goal.

2. Test dataset and class distribution of training is same.

Machine learning algorithms get influenced towards the majority class because its goal is to maximize accuracy and to get minority class dominated.

1.3 Thesis Contributions

The goals we have achieved:

- Different sampling techniques on imbalanced datasets are studied.
- Different ensemble learning procedure for class imbalance at both data level and algorithm level is learned
- Tried to mitigate the problems which have been caused by high dimensionality.
- We have made proposal about a superlative ensemble classifier for classifying high dimensional imbalanced datasets.

So basically we have tried to present a new correlation-based feature grouping approach combined with under-sampling and bagging. We have firstly generated the correlation matrix among the features [21]. After that we cluster the features into several clusters using equal size K-means clustering algorithm. Based on that matrix feature, which were selected from some cluster to form a sub data set to use as model training. To group the feature clustering, it helps us in such way that features the same cluster more related to each other. So instead of randomly forming features groups we have used clustering technique to form these groups [22–25]. The datasets we have used to run our algorithm showed us some tremendous results.

1.4 Organization of the Thesis

The thesis is organized as follows:

Chapter 2 Presents different work,which is related to class imbalance classification.

Chapter 3 : Presents proposed method in details.

Chapter 4 Presents data sets and experimental results.

Chapter 5 Presents conclusion and future work.

Chapter 2

Imbalanced Data Classification

In recent years, the class imbalance problem has gathered huge concern in the research community. Several methods have been proposed which can often be used as extensions to traditional machine learning approaches. In the previous decade, ensemble methods based on bagging, boosting and sampling methods have been among the most popular methods used to handle imbalanced binary classification problems. In this chapter, we have discussed some different methods which are co-related to our work and very effectively helpful to understand our proposed method.

2.1 Sampling Method

Oversampling and undersampling, in data analysis these artistries are used to create the class distribution of a set. Oversampling and under-sampling are reverse and approximately same techniques [26]. Both of them are using prejudice to select more samples from one class than another. The current reason for oversampling is to right for a prejudice to the archetypal dataset. The plot where it is necessary is when training a classifier using labeled training data from a prejudice origin, although labeled training data is expensive but usuaally comes from heteroclite origin.

2.1.1 Over Sampling Methods

When minority instances are increased and gets closer to majority instances.

SMOTE

A dataset can be oversampled by various mechanism. Smote is one of the most ordinary methods among them. Smote means Synthetic this method works on some training data which has s samples and f features of the feature space of the data. For intelligibility these features are continous. For example, imagine a dataset of birds for clarification. We want to oversample could be bill estent, wingspan and weight where the feature space for the minority class. A sample can be taken from the dateset and consider it is k nearest neighbors when could be oversample. To choose the vector between one of those k neighbors and the contemporary data point considering that create a synthetic data point. A random number x multiplied this vector which lies into 0 and 1. A contemporary synthetic data point gets created for adding this to the current data point.

2.1.2 Under Sampling Methods

When majority instances are reduced and gets closer to minority instances.

Cluster centroid

To say it in a simple way cluster centroid is the equidistant of a cluster. A vector which compromise each variable for one number. For observing the cluster which is known as centroid each number is the mean of a variable. It can be taught as the multifaceted average of the cluster [27]. Common measure of cluser location is used as cluster centroid and it helps us to elucidate each cluster. Each centroid is seen as constituting the "average observation" within a cluster covering all the variables in the analysis [28, 29].

2.2 Ensemble Learning

Special computational intellgence issue multiple models can be expounded, like classifiers experts, are strategically created in ensemble learning. In ensemble learning it amplifies the precision of a model, or minimizes the similarity of an unlucky selection of penurious one [30]. Supplemental implementations of ensemble learning involve imposing credit to the resolution assembled by the model, picking superior features, data blend, progressive learning, non-stationary teaching and error fixing [31]. The focal point of this artical is classifying the applications of enesemble learning, however all major intentions narrated beneath can be induced easily to corollary approximation or prognosis type difficulties as well.

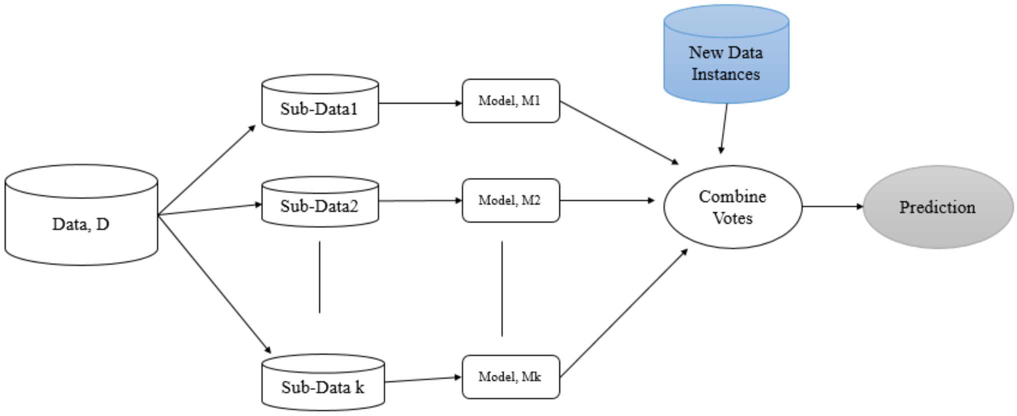

Figure 2.1: Ensemble model to advance classification accuracy.

2.2.1 Random Forest

Random forest is a group of decision trees where similar expend random vectors and all tree toss a vote for most accepted class of input instance. We can say that it's an improved version of Bagging. In case of splitting nodes, bagging works with all features set where Random Forest considers a subset of features which helps to reduce correlation among further several other trees. Decision trees might suffer from overfitting but Random Forest avoids overfitting most of the time, by generating random subsets of the features and constructing smaller trees using these subsets. While developing the trees, Random Forest adds additional randomness to the model. It pursuits for the best feature among a random subset of features alternately searching for very significant feature while splitting a node. This consequences in a wide diversity that generally results in a better model.

2.2.2 Bagging

One of the most elementary and strong method is Bagging [32]. It takes random instances with replacement when we make the sub data sets. So, it reduces the correlation among trees.When multiple machine learning algorithm attached together one by one and create different and better machine learning model, it is called ensemble method. Bootstrap Aggregation is a common process that can be used to minimize the variance for those algorithms that have high variance. An algorithm that has high variance are decision trees, like classification and regression trees. Trained decision trees get impressible to fixed data. Outcome of the decision tree can be rather dissimilar by changing the training data and it turn the enumerations can be different. It takes advantage of ensemble learning when various feeble learner outclass a single strong learner. Thus it lessens variance and helps us avoid over fitting. The bagging algorithm is shown below:

Algorithm 1 Bagging algorithm

input: Training Data D, number of iterations, k, and a learning scheme
Output: Ensemble Model, M*
Method:
for *i=1 to k* **do**
    Create bootsrap sample D_i, by sampling D with replacement;
    Use D_i, and learning scheme to derive a model, M_i;
end
To use M^* to classify a new instance, Xnew:
Each $M_i \epsilon$M* classify Xnew and return the majority vote;

2.2.3 Boosting

Boosting is a machine learning algorithm which helps to reduce prejudice, and more discrepancy in supervised learning, and a genealogy of machine learning algorithms that converts weak learners and make them strong. Boosting is established on the query invented by Kearns and Valiant [33], Can asset of weak learners create a distinct strong learner? The classifier which is only somewhat harmonized with true classification can be defined as a weak learner. On the other hand, a classifier that is randomly well-harmonized with the literal classification can be defined as a strong learner. The common factor between bagging and boosting is it merges multiple root learners for getting a result based on majority voting. However, its variances allocates weight to instances in which way it is not easy to classify.

Algorithm 2 AdaBoosting algorithm

input: Training Data D, number of iterations, k, and a learning scheme
Output: Ensemble Model, M*
Method:
initialize weight, $x_i \epsilon$ D to $\frac{1}{d}$;
for *i=1 to k* **do**
  sampling D with replacement according to instance weight to obtain D_i;
  Use D_i, and learning scheme to derive a model,M_i;
  Compute Error (M_i);
  if error$(M_i) \geq 0.5$ **then**
  go back to step 3 and try again;
  end if
  for *each correctly classified instanced* $x_i \epsilon D$ **do**
    multiply weight of x_i by $\frac{error(M_i)}{1-error(M_i)}$;
  end
  normalize the weight of instances;
end
initialize weight of each class to zero;
for *i=1 to n* **do**
  $w_i = \log\left(\frac{1-error(M_i)}{error(M_i)}\right)$; //weight of the classifier's vote
  c = M_i(X$_{\text{new}}$); // class prediction by M_i
  add w_i to weight for class c;
end
return class with largest weight;

2.3 Summary

In this chapter we have shared some important knowledge about some existing methodology. Researchers have been doing a lot research about ensemble methods, sampling technique to classify the class imbalanced data. We can see that Random Forest, Bagging, Boosting are some ensemble methods that can help to advance classification accuracy, in addition to handle imbalanced data, two types sampling method like over-sampling and under-sampling are highly operative

Chapter 3

Proposed Method

We have proposed a new algorithm entitled "Advance Enesemble of Trees (Ad.EoT)" from our analysis. Ad.EoT is established on mixture of random under sampling, feature grouping in addition bagging algorithm. It is alike to Bagging and RF with a critical unlike occurring in the feature bagging method. Our suggested Ad.EoT uses cluster-based feature grouping performed in the correlation space and trains each of the base models using one of these groups. In contrast for training the base model, complete feature set is used by Bagging while RF picks from random feature at each division of node [34]. In order to lessen class-imbalance problem, Ad.EoT parts the majority and minority class instances from the novel dataset and executes under-sampling on the minority class instances then and splits the features into m size cluster using same size K-means method. The pseudo code of modified K means method is labeled in algorithm 4. We have selected our parameter by using hyper-parameter regulation. Afterwards, at the time of training each of the base models, one of the groups is picked arbitrarily [35, 36]. The anticipation behind this formula is to use features that are linked to each other and work in assistance to form each tree. We have tried to preserve this collaboration by grouping features based on correlation matrix.

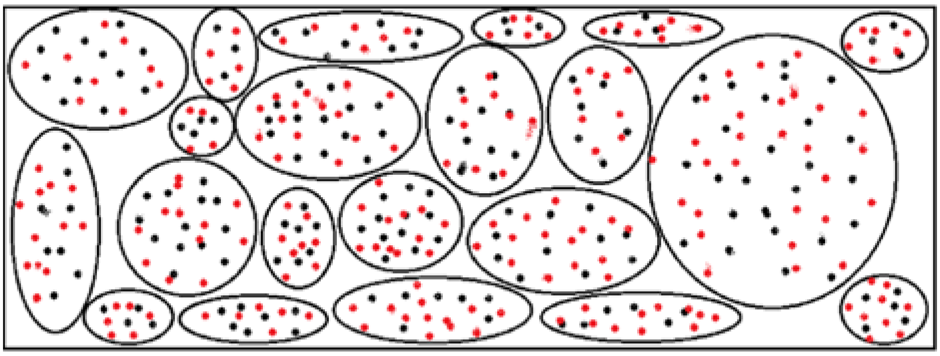

Figure 3.1: Feature Clustering using Modified K means.

Still, features in the groups expressed by our planned grouping method may be linked with each other; they will denote the same perception in such a situation as well. We have presented a probability value which is used inside the dividing technique of each of decision trees by taking into such potentials [37].
The algorithm for the proposed Ad.EoT method given below:

Algorithm 3 Ad EoT algorithm

input: iteration number i, Training Data D, and A learning scheme
Output: Ensemble Model, M*
Method:
1. remove all features with variance less then zero;
2. find the correlation matrix for attributes of D;
3. on correlation space divide attributes into m groups using modified size K means;
for *j=1 to estimator_number* **do**
    take features from a Cluster;
    Balance the dataset and generate D_j;
    train base model M_i using D_j with parameter q to model;
end
To use M^* to classify a new instance, I_{new};
Each $M_i \epsilon$M* classify I_{new} and return the majority vote;

Algorithm 4 Modified K-means algorithm

```
while ending_flag true do
    estimate centroid for each cluster;
    for every instance do
        estimate the distance to the cluster centroids;
    end
    sort instance upon improvement of the best possible substitute cluster over the
    current cluster;
    for each instances extracted from a max heap do
        for every cluster: do
                a) If any instance is waiting to leave the cluster and this swap gener-
            ates any improvement then swap instances.
        end
            b) If instances are swapped without breaching size limit, swap instances.
        if not moved:
                        a.add instances to list for handover;
    end
    if handover==0 ||iteration == max
    termination_flag == true;
    end if
end
```

By using C4.5 algorithm to classify new instances Ad.EoT merges each vote of an ensemble of decision tree. Entire instances are stuffed with indistinguishable weights and the weights stay so until the training procedure is over (Ad.EoT is not cost sensitive for dealing with class imbalance). Though, the outcome of class imbalance is lessened through under sampling inside every repetition.

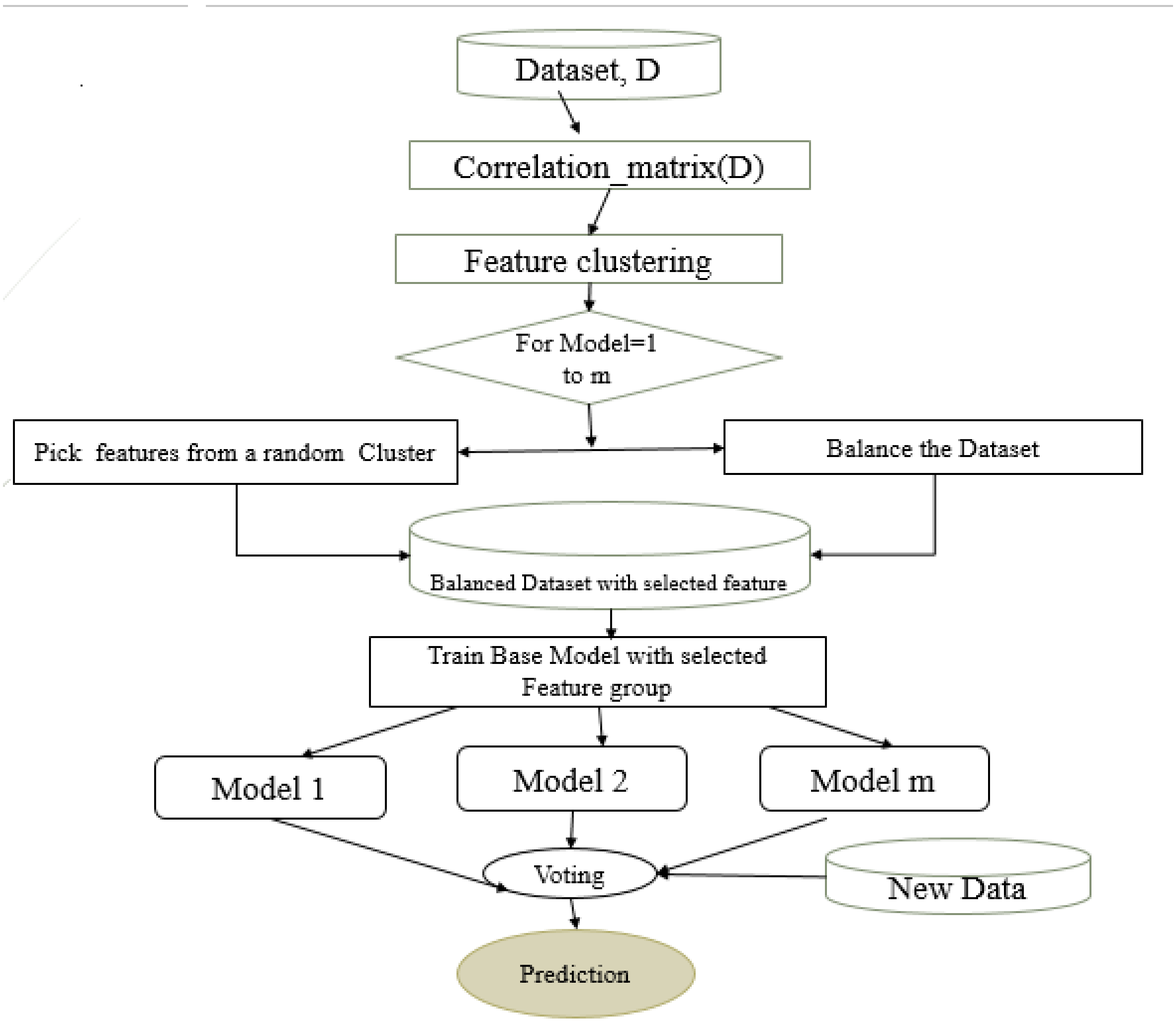

Figure 3.2: Flowchart of our proposed Method.

3.1 Comparison with Bagging

Benefits of Ad.EoT over Bagging

- Unlike Bagging, Ad.EoT disregard the full feature set for each division which guarantees multiplicity of the base model which is necessary for acceptable performance of ensemble [1].
- Above-mentioned multiplicity allows our projected method to lessen the variance of the final bagged model.

Drawbacks of Ad.EoT compare to Bagging

- Uncertainty there may occur a situation where only a few features comprise maximum number of the info, using those features can be conclusive in case of accuracy of the base models which is contributory to triumph of ensemble. But those features may not be picked for each of the base model in Ad.EoT.

3.2 Comparison with RandomForest

Benefits of Ad.EoT over RandomForest

- RandomForest pick feature group arbitrarily for each division among which the best feature is elected. Though, the selected feature may not have the level of connection which is mandatory for the accuracy of base models and that is ultimately affecting the accuracy of the final base model in such scenario where the number of features is massive [38, 39].

- Even in case of tremendously high dimensionality, Ad.EoT is unaffected from above-mentioned matter due to the approach it guarantees that each tree is trained with closely connected features [40].

Drawbacks of Ad.EoT compare to RandomForest

- The amount of multiplicity achieved by Ad.EoT may be less than RandomForest due to the approach RandomForest picks splitting feature from the complete dataset [41, 42].

Chapter 4

Experimental Analysis

4.1 Performance Evaluation

In this experiment, we have applied the recommended formula in Python and applied scikit learn integrated development environment (IDE) 0.19.1(http://scikit-learn.org/stable/). Random forest, Bagging and Boosting code has taken from Ensemble Of Tress [43]. We have used AUROC(are under receiving operating characteristic) and AUPR(area under precision and recall) to test the performance of ensemble classifier [44].

4.1.1 AUROC & AUPR

Receiving operator characteristic or ROC is a visual way of inspecting the performance of binary classification algorithm [45]. In particular, its comparing rate at which classifier make correct prediction (True Positive) and the rate at which classifier making wrong prediction (False Positive).

$$\text{TPR} = \frac{TP}{TP+FP} \tag{1}$$

$$\text{FPR} = \frac{FP}{FP+TP} \tag{2}$$

On the other hand, AUPR measure the performance with precision and recall. Precision is proportion of correct positive classes from the item we found and the total number of item we predicted as positive [46]. Recall is proportion of correct positive classification relevant item we found and the item that actually positive [47].

$$\text{Precision} = \frac{TP}{TP+FP} \tag{3}$$

$$\text{Recall} = \frac{TP}{TP+FN} \tag{4}$$

4.1.2 Confusion Matrix

Confusion Matrix is a table that shows the occurrences of correct and incorrect predictions made by classifier compared to the actual results in the data [4, 6]. There are two main classes have been considered. One is an actual class and other one predicted class. Based on this two we get four outcomes represented as following:

True Positive It is means that when a class is predicted positive the actual class is also positive indeed.

True Negative It means that when a class is predicted negative the actual class is also negative indeed.

False Positive It means that when a class is predicted positive but in actually it is negative. This error made by classifier is called Type 1 error.

False Negative It means that when a class is predicted negative but in actually it is positive. This error made by classifier is called Type 2 error.

Using these four evaluation metrics, we can now measure the accuracy and error rate. These are common metrics to find the performance of a classifier.

Table 4.1: Confusion Matrix

		Predicted	
		Positive	Negative
Actual	Positive	$TruePositive(TP)$	$FalseNegetive(FN)$
	Negative	$FalsePositive(FP)$	$TrueNegative(TN)$

4.1.3 Drawbacks of accuracy as performance metrics

Although we use accuracy as performance metrics but it is not always good to use accuracy as performance evaluation for imbalance datasets. When there is an imbalance class, it usually occurs in a way that 1 in 100 instances is a minority class

instances. Therefore, even if a classifier is unable to classify the minority class instances it will still have an accuracy of 99%. For, that reason, accuracy of a classifier is not always sufficient to determine the performance of a classifier.
Let us take an example of fraud detection banking account. For fraud account in banking, let's suppose that:

Yes = Fraud Account

No = Non Fraud Account

Here, False Positive (FP) represents that it was predicted that the account is fraud but after using practical evaluation, the account was not actually fraud. Similarly, False Negetive (FN) means that it was predicted the account was not fraud but after performance evaluation, the account was actually fraud. So, we clearly understand that the second type of error False Negetive (FN) is more costly than the first False Positive (FP). Detecting a not fraud account as fraud account did not pose a serious threat to banking security as proper measure was taken. However, not detecting an actual fraud would result in the entire banking security system in trouble. This shows type 2 error is more costly than type 1 error.

4.2 Datasets Details

We have used 10 imbalanced datasets from keel dataset repository. With imbalance ratio: ranging from 2.9 to 58.4

Table 4.2: Dataset Description.

Datasets	**Attribute**	**Instances**	**Imbalance ratio**
vowel0	13	988	9.98
page-blocks0	10	5472	8.7
winequalityred4	11	1599	29.17
poker9_vs_7	10	244	29.5
poker89_vs_6	10	1485	58.4
pageblocks13_vs_4	10	472	15.86
segment0	19	2308	6.02
vehicle1	18	846	2.9
cleveland0_vs_4	13	177	12.62
dermatology6	34	358	16.9

4.3 Results

Our proposed method along with other ensemble classifier have tested on 10 imbalance data sets. The performance of Random Forest was best among them but our method sometimes even surpasses it. Random forest got the maximum TPR (true positive rate) for most of the datasets. For AUROC, Random forest got the maximum TPR for Random forest is 100% for dermatology dataset and the maximum rate of our proposed method 99% for poker-89_vs_6 dataset. For AUPR, Random forest got the maximum precision recall rate for Random forest is 100% for dermatology dataset and the maximum rate of our proposed method 98% for pageblocks13_vs_4 dataset. The table 2 and 3 shows AUROC and AUPR comparison respectively for each of the following dataset.

Table 4.3: Average AUROC comparison

Datasets	**Bagging**	**Random Forest**	**Proposed Method**
segment0	0.964	0.979	**0.986**
vehicle1	0.990	**0.995**	0.993
cleveland0_vs_4	0.900	**0.968**	0.926
dermatology6	0.998	**1.0**	0.999
vowel0	0.985	**0.988**	0.988
page-blocks0	0.986	**0.990**	0.989
winequality-red-4	0.762	**0.795**	0.739
poker-9_vs_7	0.881	0.958	**0.990**
poker-8-9_vs_6	0.719	0.965	**0.999**
page-blocks-1-3_vs_4	0.998	0.999	**0.999**

Table 4.4: Average AUPR comparison

Datasets	**Bagging**	**Random Forest**	**Proposed Method**
segment0	0.988	**0.994**	0.987
vehicle1	0.966	**0.982**	0.977
cleveland0_vs_4	0.548	**0.784**	0.6406
dermatology6	0.96	**1.0**	0.999
vowel0	0.957	**0.994**	0.989
page-blocks0	0.905	**0.918**	0.916
winequality-red-4	0.142	**0.157**	0.137
poker-9_vs_7	0.458	0.839	**0.893**
poker-8-9_vs_6	0.183	0.631	**0.983**
page-blocks-1-3_vs_4	0.978	**0.988**	**0.988**

We also demonstrate the AUPR and AUROC comparison with graphical representation. It also shows us that Random forest performance is best and our proposed method works best for specific data sets.

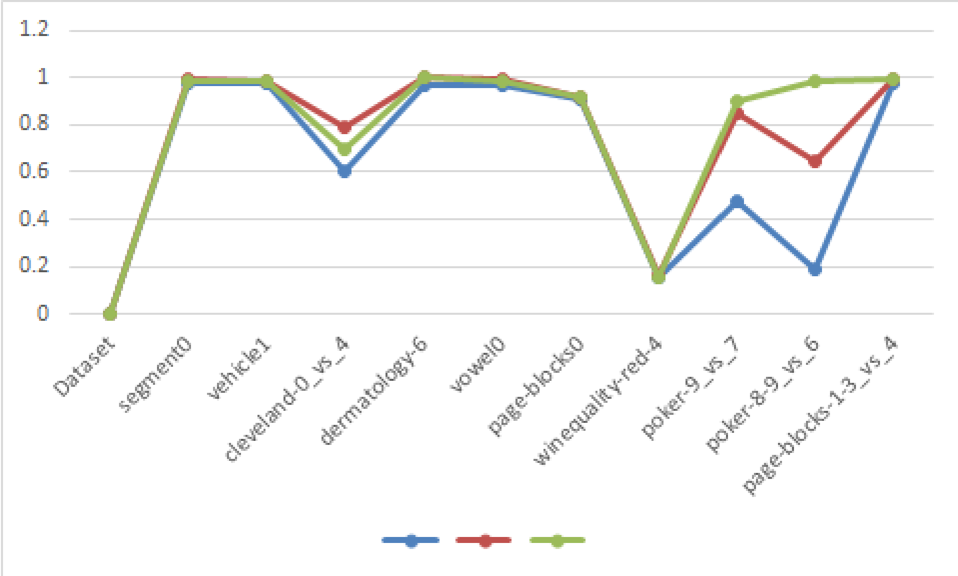

Figure 4.1: AUROC Comparison.

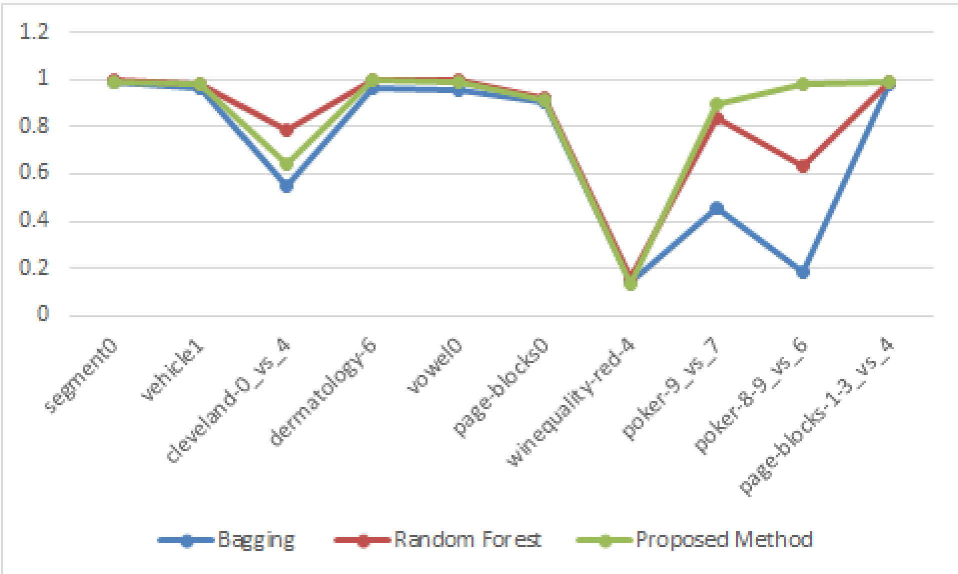

Figure 4.2: AUPR Comparison.

4.4 Summary

In this chapter, we have built new method that can perform much better than both Bagging and Random forest. But, unfortunately Random forest got better performance than our proposed method. But it won against bagging completely. If we can research further we might be able to beat random forest in future.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this thesis, we mostly involved in investigating imbalance classification problem and the receptivity of ensembles of classifiers. So far all the existing classification algorithms are focused on majority class instance while ignoring the minority class instance and gets additionally strenuous for the classifier to extract useful patterns (without over fitting on the majority class) in case of datasets which has class imbalance because of large number of feature. It becomes a huge challenge to correctly classify the instances a high dimensional imbalanced dataset by constructing an effective classifier. Now a day's artificial intelligence researchers have demonstrated a lot of hybrid techniques by mixing sampling with ensemble classifiers with different feature selection and feature grouping methods to deal with data which are high dimensional class imbalance. We are trying to instigate a new algorithm where we are grouping features based on correlation with the help of decision tree for classifying high dimensional imbalanced data. It carries the the potential of outperforming Bagging due to the decrease in correlation among the base models(feature grouping). It has the potential of outperforming Random Forest due to the way the base models over-fit particular regions of the feature space (Informed feature grouping instead of Random groups).

5.2 Future Work

In upcoming future we would like to perform experiments with other bases models. We would like to employ hierarchical clustering for grouping features. Also, we will apply these imbalanced data classification methods in real-life high-dimensional imbalanced big data and apply sampling technique with modified adaboost algorithm. We will try to discover the instructional trial instances from the trial data which will assist us to ratify the ensemble classifiers for mining imbalanced data.

Bibliography

[1] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 4, pp. 463–484, 2012. 1, 15

[2] D. M. Farid, A. Nowe, and B. Manderick, "Ensemble of trees for classifying high-dimensional imbalanced genomic data," in *Proceedings of SAI Intelligent Systems Conference*. Springer, 2016, pp. 172–187.

[3] Z. J. Lu, "The elements of statistical learning: data mining, inference, and prediction," *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, vol. 173, no. 3, pp. 693–694, 2010. 1

[4] M. A. Aly and A. F. Atiya, "Novel methods for the feature subset ensembles approach," *International Journal of Artificial Intelligence and Machine Learning*, vol. 6, no. 4, pp. 1–7, 2006. 1, 18

[5] R. Bryll, R. Gutierrez-Osuna, and F. Quek, "Attribute bagging: improving accuracy of classifier ensembles by using random feature subsets," *Pattern recognition*, vol. 36, no. 6, pp. 1291–1302, 2003. 1

[6] F. Y. Tani, D. M. Farid, and R. F. M. Zahidur, "Ensemble of decision tree classifiers for mining web data streams," *International Journal of Applied Information Systems*, vol. 1, no. 2, pp. 30–36, 2012. 2, 18

[7] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, no. 2, pp. 123–140, 1996. 2

[8] D. M. Farid and C. M. Rahman, "Assigning weights to training instances increases classification accuracy," *International Journal of Data Mining & Knowledge Management Process*, vol. 3, no. 1, pp. 13–25, 2013. 2

[9] G. M. Weiss, "Mining with rarity: a unifying framework," *ACM Sigkdd Explorations Newsletter*, vol. 6, no. 1, pp. 7–19, 2004. 3

[10] D. M. Farid, M. Z. Rahman, and C. M. Rahman, "An ensemble approach to classifier construction based on bootstrap aggregation," *International Journal of Computer Applications*, vol. 25, no. 5, pp. 30–34, 2011. 3

[11] D. M. Farid, A. Nowé, and B. Manderick, "A new data balancing method for classifying multi-class imbalanced genomic data," in *25th Belgian-Dutch Conference on Machine Learning (Benelearn)*, 2016, pp. 1–2. 3

[12] D. M. Farid, L. Zhang, A. Hossain, C. M. Rahman, R. Strachan, G. Sexton, and K. Dahal, "An adaptive ensemble classifier for mining concept drifting data streams," *Expert Systems with Applications*, vol. 40, no. 15, pp. 5895–5906, 2013. 3

[13] A. A. Afza, D. M. Farid, and C. M. Rahman, "A hybrid classifier using boosting, clustering, and naïve bayesian classifier," *World of Computer Science and Information Technology Journal (WCSIT) ISSN: 2221-0741 Vol*, vol. 1, pp. 105–109, 2011. 3

[14] N. F. Haq, A. R. Onik, M. A. K. Hridoy, Rafni, Musharrat, F. M. Shah, and D. M. Farid, "Application of machine learning approaches in intrusion detection system: a survey," *IJARAI-International Journal of Advanced Research in Artificial Intelligence*, vol. 4, no. 3, pp. 9–18, 2015. 3

[15] Y. M. Bishop and S. R. Freedman, "Classification of metadata." in *Scientific and Statistical Database Management Conference(SSDBM)*, 1983, pp. 230–234. 3

[16] A. Fujii, M. Utiyama, M. Yamamoto, and S. Shimohata, "Overview of the patent translation task at the ntcir-8 workshop," *Proc.NII Testbeds and Community for Information access Research (NTCIR-8)*, 2010. 3

[17] R. L. Lawrence and A. Wright, "Rule-based classification systems using classification and regression tree (cart) analysis," *Photogrammetric engineering and remote sensing*, vol. 67, no. 10, pp. 1137–1142, 2001. 3

[18] N. Damera-Venkata, T. D. Kite, W. S. Geisler, B. L. Evans, and A. C. Bovik, "Image quality assessment based on a degradation model," *IEEE transactions on image processing*, vol. 9, no. 4, pp. 636–650, 2000. 3

[19] S. Greenland, "Modeling and variable selection in epidemiologic analysis,," *American journal of public health*, vol. 79, no. 3, pp. 340–349, 1989. 3

[20] J. R. Quinlan, *C4. 5: programs for machine learning.* Elsevier, 2014. 4

[21] A. Liaw, M. Wiener *et al.*, "Classification and regression by randomforest," *R news*, vol. 2, no. 3, pp. 18–22, 2002. 5

[22] J. A. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural processing letters*, vol. 9, no. 3, pp. 293–300, 1999. 5

[23] S. Haykin and N. Network, "A comprehensive foundation," *Neural networks*, vol. 2, no. 2004, p. 41, 2004.

[24] L. E. Peterson, "K-nearest neighbor," *Scholarpedia*, vol. 4, no. 2, p. 1883, 2009.

[25] R. Barandelaa, J. Sanchezb, and V. Garcia, "Strategies for learning in class imbalance problems," 2003. 5

[26] F. Provost, "Machine learning from imbalanced data sets 101," in *Proceedings of the AAAI'2000 workshop on imbalanced data sets*, 2000, pp. 1–3. 7

[27] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge & Data Engineering*, no. 9, pp. 1263–1284, 2008. 8

[28] Y. Sun, A. K. Wong, and M. S. Kamel, "Classification of imbalanced data: A review," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 23, no. 04, pp. 687–719, 2009. 8

[29] H. He, Y. He, E. A. Garcia, and S. Li, "Adasyn: Adaptive synthetic sampling approach for imbalanced learning," in *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on.* IEEE, 2008, pp. 1322–1328. 8

[30] D. M. Farid, L. Zhang, C. M. Rahman, M. A. Hossain, and R. Strachan, "Hybrid decision tree and naïve bayes classifiers for multi-class classification tasks," *Expert Systems with Applications*, vol. 41, no. 4, pp. 1937–1946, 2014. 8

[31] P. Meer, D. Mintz, A. Rosenfeld, and D. Y. Kim, "Robust regression methods for computer vision: A review," *International journal of computer vision*, vol. 6, no. 1, pp. 59–70, 1991. 8

[32] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979. 9

[33] M. Kearns and L. Valiant, "Cryptographic limitations on learning boolean formulae and finite automata," *Journal of the Association for Computing Machinery(JACM)*, vol. 41, no. 1, pp. 67–95, 1994. 10

[34] K. M. Ting, "An instance-weighting method to induce cost-sensitive trees," *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 3, pp. 659–665, 2002. 12

[35] Z.-H. Zhou and X.-Y. Liu, "Training cost-sensitive neural networks with methods addressing the class imbalance problem," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 1, pp. 63–77, 2006. 12

[36] Y. Tang, Y.-Q. Zhang, N. V. Chawla, and S. Krasser, "Svms modeling for highly imbalanced classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 1, pp. 281–288, 2009. 12

[37] S. Hoque, M. Y. Arafat, and D. M. Farid, "Machine learning for mining imbalanced data," *International Conference on Emerging Technology in Data Mining and Information Security(IEMIS)*, pp. 1–10, 2018. 13

[38] R. E. Schapire and Y. Singer, "Improved boosting algorithms using confidence-rated predictions," *Machine learning*, vol. 37, no. 3, pp. 297–336, 1999. 16

[39] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of computer and system sciences*, vol. 55, no. 1, pp. 119–139, 1997. 16

[40] N. V. Chawla, A. Lazarevic, L. O. Hall, and K. W. Bowyer, "Smoteboost: Improving prediction of the minority class in boosting," in *European conference on principles of data mining and knowledge discovery.* Springer, 2003, pp. 107–119. 16

[41] H. Guo and H. L. Viktor, "Learning from imbalanced data sets with boosting and data generation: the databoost-im approach," *ACM Sigkdd Explorations Newsletter*, vol. 6, no. 1, pp. 30–39, 2004. 16

[42] S. Hu, Y. Liang, L. Ma, and Y. He, "Msmote: improving classification performance when training data is imbalanced," in *Computer Science and Engineering, 2009. WCSE'09. Second International Workshop on*, vol. 2. IEEE, 2009, pp. 13–17. 16

[43] C. Seiffert, T. M. Khoshgoftaar, J. V. Hulse, and A. Napolitano, "Rusboost: A hybrid approach to alleviating class imbalance," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 40, no. 1, pp. 185–197, 2010. 17

[44] M. Galar, A. Fernández, E. Barrenechea, and F. Herrera, "Eusboost: Enhancing ensembles for highly imbalanced data-sets by evolutionary undersampling," *Pattern Recognition*, vol. 46, no. 12, pp. 3460–3471, 2013. 17

[45] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, S. Y. Philip *et al.*, "Top 10 algorithms in data mining," *Knowledge and information systems*, vol. 14, no. 1, pp. 1–37, 2008. 17

[46] Y. Lu, Y. ming Cheung, and Y. Y. Tang, "Hybrid sampling with bagging for class imbalance learning," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining.* Springer, 2016, pp. 14–26. 17

[47] R. Barandela, R. M. Valdovinos, and J. S. Sánchez, "New applications of ensembles of classifiers," *Pattern Analysis & Applications*, vol. 6, no. 3, pp. 245–256, 2003. 17