Ensemble Learning For Imbalanced Data Classification Problem

Pasapitch Chujai*, Kittipong Chomboon, Pongsakorn Teerarassamee, Nittaya Kerdprasop, Kittisak Kerdprasop

School of Computer Engineering, Institute of Engineering, Suranaree University of Technology,
Nakhorn Ratchasima 3000, Thailand

*Corresponding Author: pasapitchchujai@gmail.com

Abstract

Imbalanced data is a kind of information that occurs in real life, such as medical diagnosis in which records of seriously ill patients outnumber by records of healthy ones. These imbalanced data affect the learning performance of algorithms in data mining. The boundary of decision in out of balance data chosen by most standard algorithms of machine learning tends to bias toward the majority class and hence misclassifies the minority class. Therefore, we present an approach for dealing with imbalanced data classification problem by applying the decision tree ensemble learning using both bagging and boosting techniques to build models that compensate the misclassification with cost sensitive learning. In this research, we build the model templates from different characteristics of synthetic data. We have chosen an appropriate model template for the real data with different imbalanced rating and overlapping ratio. The results showed that the chosen model template can solve the imbalanced data classification problem efficiently. But there are some model templates that cannot classify correctly when imbalanced rate increases.

Keywords: Ensemble Learning, Imbalanced Datasets, Decision Tree Classification, Cost-Sensitive Learning, Visualization.

1. Introduction

Data mining⁽¹⁾ is a method that has been extensively used to retrieve the hidden knowledge from a large information repository. Data mining task has many categories depending on the purpose of application. The one of those categories is data classification that aims to learn patterns to make prediction about the class of some unknown data. Most standard algorithms for data

classification can be applied very efficiently in terms of overall classification accuracy if data in each class are in equal proportion. However, these algorithms show poor learning performance when classifying the imbalanced data that have amount of instances in the group of interest less than those in the other groups⁽²⁾.

For example, we can demonstrate a comparison between classifying balanced data and unbalanced data with 300 instances and two classes. For the balanced dataset, the amounts of data in the two classes are equal; that is, 150 instances in each class. For the unbalanced dataset, there are 285 instances in the *class a*, whereas there are only 15 instances in the *class b*. Take both datasets to be classified by decision tree induction. The results are shown in Fig. 1. Both datasets show good performance in terms of overall classification accuracy. When considering accuracy in each class, we found that the performance of classifying *class b* in the balanced data is more accurate than classifying *class b* in the imbalanced data. This classification accuracy drops drastically from 0.947 to 0.467.

Fig. 1. Comparisons of classification between the two datasets of balanced data and unbalanced data.

This example indicates that using imbalanced data in classification will affect the learning performance of algorithms that tend to bias toward the group of majority and cause high misclassification rate over a group of minority.

The problem of classifying imbalanced data mentioned above has drawn attention from many researchers to propose various methods to solve this problem. The proposed methods focus on a more accurate classification over a minority group. Some important work that proposed the methods mentioned above are as follows:

Brown and Mues⁽³⁾ proposed the undersampling technique to deal five credit scoring imbalanced datasets.

Cateni et al.⁽⁴⁾ proposed the oversampling and undersampling techniques to deal the benchmarks imbalanced datasets from the KEEL Repository.

Lopez et al.⁽⁵⁾ studied the performance of classification with the hybrid techniques of SMOTE+ENN and SMOTE. They solved the problem by combining the techniques at data level approach and algorithm level approach and focused on the cost-sensitive learning. They compared these techniques with their proposed hybrid techniques and performed experiments with 66 datasets that were taken from the KEEL Repository.

Krawczyk et al.⁽⁶⁾ proposed the decision tree ensemble data classification and cost-sensitive learning to deal with six binary benchmarks imbalanced datasets from the KEEL Repository. They proposed a technique to prune the decision tree using the novel algorithm and optimal C_{minority} derived from the ROC analysis. They compared the proposed method with the other six methods. The result showed that their proposed method was efficient in some datasets.

Liao et al.⁽⁷⁾ proposed the ensemble learning for binary classification. They used Support Vector Machine (SVM) for rebalancing data in the stage of preprocessing and then selected the features for ensemble learning with Back-Propagation Neural Network (BPNN). The outputs from ensemble learning were taken to build new knowledge by using the rough set theory. They performed experiments with the listed electronics companies from 2005 to 2011: 63 financial crisis corporations and 2680 non-financial crisis corporations. The result showed that their proposed method was more efficient than other methods.

We recognize the importance of solving imbalanced data classification problem with an effective method. Therefore, we present an approach for dealing with imbalanced datasets by applying the decision tree ensemble

learning using both bagging and boosting techniques to build models and combine the compensation technique to handle misclassification with cost sensitive learning.

The rest of this research is organized as follows: Section 2 gives details of the background and relevant techniques. Section 3 presents details of our proposed method. The experimental results and analysis will be presented in Section 4. Finally, the research is concluded in Section 5.

2. Background

2.1 Imbalanced Data

Imbalanced data is the data that have the amount of instances in the group of interest much smaller than those in other classes⁽²⁾. The group of data that has a larger number of instances is called the majority class or negative class, whereas the group of data that has a smallest number of instances is called the minority class or positive class⁽⁸⁾. When classifying imbalanced data, the boundary of decision acceptable by standard algorithms tends to bias toward the majority class and misclassify in minority class as illustrated in Fig. 2.

Characteristics of imbalanced data that can influence the classification algorithms⁽⁹⁾ are divided into three cases as follows:

(a) Imbalanced Ratio

Imbalanced data can be verified by the degree of imbalance, which represents the ratio between the number of data instances in majority class (n_{majority}) and the number of data instances in minority class (n_{minority})⁽¹⁰⁾. The imbalanced ratio can be defined by Eq. (1)

$$\text{Imbalanced Ratio (IR)} = \frac{n_{\text{majority}}}{n_{\text{minority}}} \quad (1)$$

(b) Lack of data

This problem occurs when the size of samples in the minority class is too small⁽⁹⁾. Because of small sample size will cause difficulty in finding the patterns.

(c) Overlapping ratio between classes

Overlap occurs when the data of each class has shared area. Overlap that occurs in conjunction with imbalanced data would result in the more complex situation for classification⁽¹¹⁾. Maximum Fisher's Discriminant Ratio (F1) is one method that can be used for measurement the overlapping ratio. The F1 is defined by Eq. (2)

Fig. 2. Linear classification of imbalanced data which bias towards majority class.

$$f_i = \frac{(\mu_{minority} - \mu_{majority})_i^2}{(\sigma_{minority}^2 - \sigma_{majority}^2)_i} \quad (2)$$

where f_i is Fisher's Discriminant Ratio of feature i .

$\mu_{minority}$, $\mu_{majority}$ are mean of minority class and majority class, respectively.

$\sigma_{minority}^2$, $\sigma_{majority}^2$ are variance of minority class and majority class, respectively.

The methods for solving imbalanced classification problem^(5,12) can be divided into three approaches as follows:

(a) Data Level Approaches

This approach solves the problem in a pre-processing stage by rebalancing the class distribution using the sampling techniques: oversampling, undersampling, and a hybrid technique.

(b) Algorithm Level Approaches

This approach attempt to adapt existing algorithms by adjusting the parameters.

(c) Cost Sensitive Approaches

This approach uses both data level approaches by adding special cost to misclassification and algorithm level approaches by modification the algorithms to the possible classification that leads to less errors.

2.2 Ensemble Learning

The functions of single model have high classification performance but have a problem in terms of a fixed a set of parameters, which causes the bias. Reduction of such bias can be obtained through the ensemble learning.

The performance of ensemble learning depends on the precision of classifiers. In ensemble classification learning, multiple classifiers are used to learn the original dataset together. The results from learning will be combined and then used to classify the unknown data. The process of ensemble learning is given in Fig. 3⁽¹³⁾.

Fig. 3. The fundamental step of ensemble learning.

Ensemble learning can be divided into three approaches as follows:

(a) Boosting method

Boosting method⁽¹⁴⁾ is an ensemble classification such that each classifier has a weight which is derived from the precision of learning. The results models are used to predict the unknown data by the majority vote. The popular technique is AdaBoost.

(b) Bagging method

Bagging method⁽¹⁵⁾ builds the models from the same learning algorithm but each algorithm learns from different instances. This method also uses majority vote for prediction of unknown data. The popular technique is Bag.

(c) Random subspace method

Random subspace method or Attribute Bagging⁽¹⁶⁾ learns from the same dataset and then performs sampling without replacement over the features. The method also uses majority vote for the prediction of unknown data.

2.3 Cost Sensitive Learning

Cost-sensitive learning takes into account the cost of misclassification. Penalties of misclassification will be built as a cost matrix as shown in Table 1.

From Table 1, let $C(i,j)$ be the cost of predicting the sample in class i as class j . $C(0,0)$ and $C(1,1)$ are the cost of correct classification which is set to be equal to 0. $C(0,1)$ is the cost of misclassifying of majority class to be minority, and the cost is set to 1. $C(1,0)$ is the cost of misclassifying of minority class to be majority. The cost is $C_{minority}$, which can be adjusted according to the specific algorithm.

The most important issue for solving the imbalanced data classification problem is recognizing correctly the positive instance (minority class) rather than the negative instance (majority class). Therefore, the cost of misclassifying of minority class must higher than the cost of misclassifying of majority class ($C(1,0) > C(0,1)$).

Table 1. Cost matrix C for binary classification.

Actual Class	Predicted Class	
	Majority Class (0)	Minority Class (1)
Majority Class (0)	$C(0,0) = 0$	$C(0,1) = C_{\text{majority}} = 1$
Minority Class (1)	$C(1,0) = C_{\text{minority}}$	$C(1,1) = 0$

3. Methodology

In this research, our main objective is to find the classification model efficiently for solving the imbalanced data classification problem with high accuracy and efficiency. Our concern is to improve the process of imbalanced classification at different imbalanced ratio and overlapping ratio between classes. We apply the decision tree ensemble learning using both bagging and boosting techniques to build models and compensate the misclassification with cost sensitive learning by building cost matrix and then take the values from cost matrix to adjust the parameters of ensemble learning. We also find the optimal number of trees by visualization. The framework and the steps are shown in Fig. 4, we can explain in detail of each stage as follows:

(a) Building the Model Templates

This stage is for building the model templates with the following two steps:

1. Generate the numeric synthetic data, 1000 instances with normal distribution, slightly overlapped with imbalance ratios of 10%.
2. Build model templates from the synthetic data using five algorithms: AdaBoostM1, Bag, TotalBoost, LogitBoost and RUSBoost. Name of the model templates are M1, M2, M3, M4 and M5, respectively.

(b) Building the System Model

This stage is for building the system model with the following six steps:

1. Normalization the features to zero mean and set standard deviation equal to 1.
2. Sampling data by using stratified sampling to draw samples from the imbalanced data with several of imbalance ratios. We call the sampled data DB1.
3. Find optimal cost value for minority class by generate cost matrix for misclassification cost. We are setting IR of DB1 to be the C_{minority} . Other values in the cost matrix will be determined by a constant: C_{majority} equals 1, $C(0,0)$ and $C(1,1)$ equals 0.
4. Model matching by analyzes the characteristics of DB1 and then choose the appropriate model from the model templates.

Fig. 4. The framework of steps for building the imbalanced classification models.

5. Building the imbalanced classification model using the model templates and the best value of C_{minority} . We initialize the number of decision trees to be equal to 200 and test the model by k-fold cross validation, $k=5$.

6. Find optimal number of decision trees, we employ visualization to reduce the number of decision trees obtained from ensembles learning to achieve a number of suitable decision trees. The visualization will show the test of classification error for each decision tree and we will choose the top-10 decision trees with the minimum error.

(c) Test Performance of the Ensemble Model

The optimal ensemble model will be tested to evaluate its performance by the evaluation measure.

4. Experimental Results

4.1 Datasets

For our experiment, the proposed ensemble models have been developed and applied for binary classification on the following datasets:

(a) Synthetic datasets

The synthetic datasets have been created using Matlab. We created five datasets which have a slight overlap with an initial imbalanced ratio of 10%, two classes and three features. Details of the synthetic datasets and the optimal model templates are given in Table 2 and one of the synthetic dataset is shown in Fig. 5.

(b) KEEL datasets

For our experiments, we have taken six binary imbalanced datasets from the KEEL Repository⁽¹⁷⁾. Details of the employed datasets are given in Table 3.

Table 2. Details of the synthetic datasets and optimal model templates.

Data Set	Characteristics	Model Template
D1	M=[0 0 0; 2 2 2] S1=[0.2 0.0 0.0; 0.0 0.1 0.0; 0.0 0.0 0.0] S2=[1.1 0.3 0.2; 0.3 1.1 0.7; 0.2 0.7 1.2]	AdaBoostM1
D2	M=[0.65 -0.09 -2.59; -0.05 0.01 0.19] S1=[1.09,0.03,-0.13; 0.03,0.73, 0; -0.13,0, 0.05] S2=[0.96,-0.02,0.57; -0.02,1.02,0; 0.57,0, 0.55]	Bag
D3	M=[0.18 0.45 1.07; -0.02 -0.05 -0.12] S1=[1.87,1.46,2.42; 1.46,8.58,-0.81; 2.42,-0.81,6.33] S2=[0.90,0.26,0.31; 0.26, 0.11,0.06; 0.31,0.06,0.25]	TotalBoost
D4	M=[3 1 3; 0 0 0] S1=[0.77,-1.17,-0.98;-1.17,5.82,5.17;-0.98,5.17,4.6] S2=[0.23,-0.21,-0.09;-0.21,0.49,0.2; -0.09,0.2,0.14]	LogitBoost
D5	M=[0 0 0; 1 1 1] S1=[0.8,-0.2,-0.2;-0.2,1.8,-0.1;-0.2,-0.1,0.3] S2=[0.8,0.1,-0.01; 0.1, 0.8,0.2; -0.01,0.2, 0.8]	RUSBoost
<i>M is mean of minority class and majority class.</i> <i>S1 and S2 are variance of minority class and majority class, respectively.</i>

Fig. 5. An overview of the synthetic dataset: D5.

Table 3. Details of the datasets used in the experiments.

Name	Attribute (R/I/N)*	Instance	Majority Class	Minority Class	IR
page-blocks	10 (4/6/0)	5472	4913	559	8.79
pima	8 (8/0/0)	768	500	268	1.87
segment	18 (18/0/0)	2308	1979	329	6.02
shuttle	9 (0/9/0)	1829	1706	123	13.87
vehicle	18 (0/18/0)	846	628	218	2.88
yeast	8 (8/0/0)	1484	1321	163	8.10
(R/I/N)* : Real/Integer/Nominal Valued					

Table 4. Confusion Matrix for binary classification.

	Positive Prediction	Negative Prediction
Actual Positive Class	True Positive (TP)	False Negative (FN)
Actual Negative Class	False Positive (FP)	True Negative (TN)

4.2 Evaluation Metrics

In order to evaluate the effectiveness of the proposed method we used confusion matrix to show the accuracy of the classification and the reliability of the model. Detail of confusion matrix is given in Table 4.

From Table 4, row of the matrix shows the number of actual instances of each class and column shows the number of predicted of each class. It is divided into the following four cases:

TP: number of instances that are correctly classified as positive class.

TN: number of instances that are correctly classified as negative class.

FP: number of instances that are negative class incorrectly classified as positive class.

FN: number of instances that are positive class incorrectly classified as negative class.

(a) Sensitivity or True Positive Rate (TPRate) or Recall

This measure shows the ability of the model to classify positive class, which is defined as the ratio between the number of correctly classified positive class and the total number of the actual positive class. The sensitivity is defined by Eq. (3)

$$Recall = Sensitivity = TPrate = \frac{TP}{(TP + FN)} \quad (3)$$

(b) Specificity or True Negative Rate (TNRate)

This measure shows the precision of the classification model in classifying negative class, which is defined as the ratio between the number of correctly classified negative class and the total number of the actual negative class. The specificity is defined by Eq. (4)

$$Specificity = TNrate = \frac{TN}{(TN + FP)} \quad (4)$$

(c) Accuracy

This measure shows evaluation of the overall performance of all classes in the classification of a model. The accuracy is defined by Eq. (5)

$$Accuracy = \left(\frac{TP + TN}{TP + FN + TP + FP} \right) * 100 \quad (5)$$

4.3 Results and Analyses

In this section, we present the results from the evaluation of our proposed model using six binary benchmarks imbalanced datasets. We perform stratified sampling to draw samples from imbalanced datasets at different imbalanced ratios and then analyze the characteristics of data to find the suitable model from the model templates. The results of the suitable model templates are given in Table 5.

We derive the C_{minority} from imbalanced ratio (IR) and then use it in the step of building the ensemble model. The initial number of decision trees is 200 and then reduce this number to get the optimal number of decision trees using visualization.

Table 5. Optimal model templates for benchmark datasets.

Dataset	Model Template
page-blocks	TotalBoost
pima	RUSBoost
segment	AdaBoostM1, LogitBoost and TotalBoost
shuttle	Bag and LogitBoost
vehicle	LogitBoost
yeast	RUSBoost

Fig. 6 shown the decrease in the number of decision trees of the yeast dataset adjusted with the smallest error rate. The optimal number of decision trees equal 30.

From the obtained optimal model templates shown in Table 5, we performed the experiments over the benchmark datasets with these models. We also performed the experiments with the rest of model templates as a base line for comparisons. We experiments with the three imbalanced ratios: 1:10, 1:25 and 1:50. Each imbalanced ratio is equal C_{minority} . The performance of such models in terms of sensitivity (SE), specificity (SP) and accuracy (Acc) are given in Table 6, Table 7 and Table 8, respectively.

Fig. 6. Shows the error of classification of yeast dataset with varying number of decision trees.

Table 6. Classification results for benchmark datasets with imbalanced ratios 1:10, the best value of C_{minority} is 10.

Model	AdaBoostM1			Bag			TotalBoost			LogitBoost			RUSBoost		
Dataset	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc
page-blocks	66.67	99.39	96.30	40.83	99.83	94.25	82.50	98.00	96.54	81.67	99.04	97.40	79.17	95.74	94.17
pima	20.00	95.33	88.48	6.67	100.00	91.52	33.33	94.00	88.48	26.67	98.67	92.12	80.00	76.00	76.36
segment	97.50	99.50	99.32	57.50	100.00	96.14	95.00	99.25	98.86	95.00	99.50	99.09	85.00	99.50	98.18
shuttle	0.00	100.00	90.24	100.00	100.00	100.00	0.00	100.00	90.24	100.00	100.00	100.00	80.00	100.00	98.05
vehicle	70.00	99.00	96.36	0.00	100.00	90.91	70.00	98.00	95.45	75.00	99.00	96.82	85.00	90.00	89.55
yeast	66.67	98.95	95.87	3.33	100.00	90.79	86.67	94.39	93.65	76.67	98.60	96.51	93.33	93.68	93.65

Table 7. Classification results for benchmark datasets with imbalanced ratios 1:25, the best value of C_{minority} is 25.

Model	AdaBoostM1			Bag			TotalBoost			LogitBoost			RUSBoost		
Dataset	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc
page-blocks	54.17	99.93	98.17	17.50	99.93	96.76	80.00	99.27	98.53	77.50	99.53	98.69	79.17	96.77	96.09
pima	13.33	99.47	96.15	0.00	100.00	96.15	40.00	96.00	93.85	26.67	97.33	94.62	80.00	80.80	80.77
segment	97.50	100.00	99.90	15.00	100.00	96.73	97.50	99.90	99.81	97.50	100.00	99.90	75.00	99.80	98.85
shuttle	0.00	100.00	96.15	75.00	100.00	99.04	0.00	100.00	96.15	100.00	100.00	100.00	100.00	99.70	99.71
vehicle	50.00	99.80	97.88	5.00	100.00	96.35	65.00	88.40	87.50	65.00	99.60	98.27	85.00	90.60	90.38
yeast	36.67	99.47	97.05	3.33	100.00	96.28	80.00	93.47	92.95	66.67	99.60	98.33	86.67	94.27	93.97

Table 8. Classification results for benchmark datasets with imbalanced ratios 1:50, the best value of C_{minority} is 50.

Model	AdaBoostM1			Bag			TotalBoost			LogitBoost			RUSBoost		
Dataset	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc
page-blocks	35.83	99.98	98.73	12.50	100.00	98.28	73.33	99.62	99.10	70.83	99.82	99.25	73.33	97.20	96.73
pima	0.00	100.00	98.04	0.00	100.00	98.04	26.67	99.07	97.65	13.33	100.00	98.30	73.33	73.73	73.73
segment	92.50	100.00	99.85	15.00	99.85	98.19	92.50	99.95	99.80	92.50	100.00	99.85	82.50	99.55	99.22
shuttle	0.00	100.00	98.04	35.00	100.00	98.73	0.00	100.00	98.04	100.00	100.00	100.00	100.00	99.85	99.85
vehicle	40.00	99.90	98.73	5.00	100.00	98.14	60.00	99.50	98.73	50.00	99.90	98.92	90.00	89.70	89.71
yeast	3.33	99.93	98.04	0.00	100.00	98.04	80.00	86.07	85.95	50.00	99.60	98.63	83.33	95.93	95.69

Table 9. Comparison between the best results from our proposed and the method with proposed by Krawczyk et al.⁽⁶⁾

IR	1:10				1:25				1:50			
Dataset	Krawczyk et al. ⁽⁶⁾		Proposed		Krawczyk et al. ⁽⁶⁾		Proposed		Krawczyk et al. ⁽⁶⁾		Proposed	
	SE	SP	SE	SP	SE	SP	SE	SP	SE	SP	SE	SP
page-blocks	82.95	80.23	82.50	98.00	70.76	81.23	80.00	99.27	88.04	87.64	73.33	99.62
pima	85.23	97.10	80.00	76.00	77.13	94.50	80.00	80.80	71.43	92.67	73.33	73.73
segment	75.24	82.94	97.50	99.50	72.89	90.11	97.50	100.00	70.02	93.43	92.50	100.00
shuttle	92.31	89.23	100.00	100.00	86.28	91.05	100.00	100.00	75.98	92.98	100.00	100.00
vehicle	88.23	89.23	85.00	90.00	77.67	93.45	85.00	90.60	67.37	90.98	90.00	89.70
yeast	70.25	97.23	93.33	93.68	67.78	98.11	86.67	94.27	60.34	96.22	83.33	95.93

The results of the experiments in Table 6, Table 7 and Table 8, show that there are chosen model templates could classify efficiently, such as RUSBoost model for yeast and pima datasets and TotalBoost model for page-blocks dataset. For vehicle dataset, we choose LogitBoost model templates which are inappropriate, RUSBoost model could classify efficiently. The rest of datasets demonstrate that part of model template can classify efficiently and some model templates without chosen; RUSBoost model for shuttle and page-blocks datasets; are able to classify efficiently.

The model template is disabling to classify efficiently when increasing the imbalanced ratio. The examples are Bag model for segment dataset can classify correctly at imbalanced ratio at 1:10, and AdaBoostM1, TotalBoost and LogitBoost model for segment dataset can classify correctly at imbalanced ratio at 1:25. For increasing imbalanced ratios of shuttle dataset, our proposed method could improve the classification of imbalanced data significantly with chosen model template: LogitBoost model.

As for comparison with the results proposed by Krawczyk et al.⁽⁶⁾ is shown in Table 9. The proposed method shows the performance is quite satisfactory; especially the imbalanced datasets are imbalance ratio of 1:25 and 1:50. For an imbalance ratio of 1:10, the proposed method was statistically better on three of imbalanced

datasets, while the competition method was statistically better on the rest of imbalanced datasets. For an imbalance ratio of 1:25, the proposed method was statistically better on all of six imbalanced datasets. For an imbalance ratio of 1:50, the proposed method was statistically better on five of imbalanced datasets, while the competition method was statistically better on the rest of imbalanced datasets.

5. Conclusions

Imbalanced data classification is a significant challenge for standard algorithms of machine learning. In this research, we propose the method to deal with imbalanced data classification problems with the main focus to improve recognition of the minority class. We combine the cost-sensitive learning with ensemble decision tree classification using bagging and boosting techniques. The numbers of decision trees are also decreased to an optimal number of decision trees by visualization. We create normally distributed synthetic data with binary classes, three features and 1000 instances. Then, we build the model templates from five algorithms: AdaBoostM1, Bag, TotalBoost, LogitBoost and RUSBoost. We analyze the standard datasets and selected the best model template by considering the overlapping between classes, mean and

standard variation of minority class and majority class with closed to model templates. The experiments show that the overlapping ratio between classes has an effect to the performance of proposed model. If datasets has overlapping between classes, the model can classify correctly at imbalanced ratio not over 25. The appropriate ensemble technique is boosting such as RUSBoost, LogitBoost, TotalBoost, and AdaBoostM1. For Bag model, it classifies correctly at imbalanced ratio not over 10. The best model is RUSBoost that can classify an imbalanced data which have overlapping between classes and high imbalanced ratio.

Acknowledgment

Financial support for Pasapitch Chujai has been provided by a scholarship from Office of the Higher Education Commission, Thailand.

References

- (1) Jiawei Han, and Micheline Kamber: "Data mining: concepts and techniques" Amsterdam; Boston: Elsevier, 2006
- (2) Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer: "SMOTE: synthetic minority over-sampling technique", *Journal of Artificial Intelligence Research*, Vol. 16, pp. 321–357, 2002
- (3) Iain Brown, and Christophe Mues: "An experimental comparison of classification algorithms for imbalanced credit scoring data sets", *Expert Systems with Applications*, Vol. 39, No. 3, pp. 3446-3453, 2012
- (4) Silvia Cateni, Valentina Colla, and Marco Vannucci: "A method for resampling imbalanced datasets in binary classification tasks for real-world problems", *Neurocomputing*, Vol. 135, pp.32-41, 2014
- (5) Victoria López, Alberto Fernández, Jose G. Moreno-Torres, and Francisco Herrera: "Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification. Open problems on intrinsic data characteristics", *Expert Systems with Applications*, Vol. 39, No. 7, pp. 6585-6608, 2012
- (6) Bartosz Krawczyk, Michal Wozniak, and Gerald Schaefer: "Cost-sensitive decision tree ensembles for effective imbalanced classification", *Applied Soft Computing*, Vol. 14, pp. 554-562, 2014
- (7) Jui-Jung Liao, Ching-Hui Shih, Tai-Feng Chen, and Ming-Fu Hsu: "An ensemble-based model for two-class imbalanced financial problem", *Economic Modeling*, Vol. 37, pp. 175-183, 2014
- (8) M.A.H. Farquad, and Indranil Bose: "Preprocessing unbalanced data using support vector machine", *Decision Support Systems*, Vol. 53, No. 1, pp. 226-233, 2012
- (9) Son Lam Phung, Abdesselam Bouzerdoun, and Giang Hoang Nguyen: "Learning pattern classification tasks with imbalanced data sets", In P. Yin (Eds.), *Pattern recognition*, Vukovar, Croatia: In-Teh, pp. 193-208, 2009
- (10) Albert Orriols-Puig, and Ester Bernadó-Mansilla: "Evolutionary rule-based systems for imbalanced data sets", *Soft Computing*, Vol. 13, No. 3, pp. 213-225, 2009
- (11) Julian Luengo, Alberto Fernández, and Francisco Herrera: "Addressing data-complexity for imbalanced data-sets: A preliminary study on the use of preprocessing for c4.5", In *Intelligent Systems Design and Applications*, 2009. ISDA'09. Ninth International Conference on IEEE, pp. 523-528, 2009
- (12) Mikel Galar, Alberto Fernandez, Edurne Barrenechea, Humberto Bustince, and Francisco 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, 2012
- (13) Meenakshi A. Thalor, and S. T. Patil: "Review of Ensemble Based Classification Algorithms for Nonstationary and Imbalanced Data", *IOSR Journal of Computer Engineering (IOSR-JCE)*, Vol. 16, No. 6, pp. 103-107, 2014
- (14) Yoav Freund, and Robert E. Schapire: "Experiments with a new boosting algorithm", *Proceedings 13th International Conference on Machine Learning*, Vol. 96, pp. 148-156, 1996
- (15) Leo Breiman: "Bagging predictors", *Machine learning*, Vol. 24, No. 2, pp. 123-140, 1996
- (16) Robert Bryll, Ricardo Gutierrez-Osuna, and Francis Quek: "Attribute bagging: improving accuracy of classifier ensembles by using random feature subsets", *Pattern Recognition*, Vol. 30, No. 6, pp. 1291–1302, 2003
- (17) Keel Datasets, available from <http://www.keel.es/datasets.php>.