Classification of Imbalanced Data by Combining the Complementary Neural Network and SMOTE Algorithm

Piyasak Jeatrakul, Kok Wai Wong, and Chun Che Fung

School of Information Technology
Murdoch University
South Street, Murdoch, Western Australia 6150
[p.jeatrakul | k.wong | l.fung] @murdoch.edu.au

Abstract. In classification, when the distribution of the training data among classes is uneven, the learning algorithm is generally dominated by the feature of the majority classes. The features in the minority classes are normally difficult to be fully recognized. In this paper, a method is proposed to enhance the classification accuracy for the minority classes. The proposed method combines Synthetic Minority Over-sampling Technique (SMOTE) and Complementary Neural Network (CMTNN) to handle the problem of classifying imbalanced data. In order to demonstrate that the proposed technique can assist classification of imbalanced data, several classification algorithms have been used. They are Artificial Neural Network (ANN), k-Nearest Neighbor (k-NN) and Support Vector Machine (SVM). The benchmark data sets with various ratios between the minority class and the majority class are obtained from the University of California Irvine (UCI) machine learning repository. The results show that the proposed combination techniques can improve the performance for the class imbalance problem.

Keywords: Class imbalanced problem, artificial neural network, complementary neural network, classification, misclassification analysis

1 Introduction

In recent years, many research groups have found that an imbalanced data set could be one of the obstacles for many Machine Learning (ML) algorithms [1], [2], [3], [4]. In the learning process of the ML algorithms, if the ratio of minority classes and majority classes is significantly different, ML tends to be dominated by the majority classes and the features of the minority classes are recognize slightly. As a result, the classification accuracy of the minority classes may be low when compared to the classification accuracy of the majority classes. Some researchers have examined this problem under the balancing of the bias and variance problems [5].

According to Gu et al. [4], there are two main approaches to deal with imbalanced data sets: data-level approach and algorithm approach. While the data-level approach aims to re-balance the class distribution before a classifier is trained, the algorithm level approach aims to strengthen the existing classifier by adjusting algorithms to

recognize the smaller classes. There are three categories of data-level approach. These are the under-sampling technique, the over-sampling technique and the combined technique. For the under-sampling techniques, many algorithms have been proposed, for example Random under-sampling [1], Tomek links [6], Wilson's Edited Nearest Neighbor Rule (ENN) [7], and Heuristic Pattern Reduction (HPR) [8]. There are also several techniques applied for over-sampling methods such as Random over-sampling [1], and Synthetic Minority Over-sampling Technique (SMOTE) [3].

In order to evaluate the classification performance of an imbalanced data set, the conventional classification accuracy cannot be used for this purpose because the minority class has minor impact on the accuracy when compared to the majority class [4]. Therefore, alternative measures have to be applied. The Geometric mean (G-mean) and the area under the Receiver Operating Characteristic (ROC) curve have been applied to evaluate the classification performance for imbalanced data set [4]. They are good indicators for the class imbalance problem because they attempt to maximize and balance the performance of ML between the minority class and the majority class. G-mean and the area under ROC curve (AUC) are also independent of the imbalanced distribution [9].

In the reported literature, most research dealt with this problem with an aim to increase the classification performance of imbalanced data. They focused on examining the feasibility of re-distribution techniques for handling imbalanced data [1], [2], [3], [9]. Furthermore, several cases in the literature have presented that the combination of under-sampling and over-sampling techniques generally provides better results than a single technique [1]. By considering in a similar direction, this paper takes an approach by proposing alternative re-distribution techniques to enhance the classification performance. A combined technique based on both sampling techniques is also proposed.

In this paper, in order to re-balance the class distribution, the combined approaches of two techniques, Complementary Neural Network (CMTNN) and Synthetic Minority Over-Sampling Technique (SMOTE), are proposed. While CMTNN is applied as an under-sampling technique, SMOTE is used as an over-sampling technique. CMTNN is used because of its special feature of predicting not only the "truth" classified data but also the "false" data. SMOTE is applied because it can create new instances rather than replicate the existing instances. SMOTE is also the successful over-sampling technique applied commonly to the class imbalanced problem in the literature [1], [4].

2 The Proposed Techniques

In this section, the concepts of CMTNN and SMOTE are described. The proposed combined techniques will then be presented.

2.1 Complementary Neural Network (CMTNN)

CMTNN [10] is a technique using a pair of complementary feedforward backpropagation neural networks called Truth Neural Network (Truth NN) and

Falsity Neural Network (Falsity NN) as shown in Fig 1. While the Truth NN is a neural network that is trained to predict the degree of the truth memberships, the Falsity NN is trained to predict the degree of false memberships. Although the architecture and input of Falsity NN are the same as the Truth NN, Falsity NN uses the complement outputs of the Truth NN to train the network. In the testing phase, the test set is applied to both networks to predict the degree of truth and false membership values. For each input pattern, the prediction of false membership value is expected to be the complement of the truth membership value. Instead of using only the truth membership to classify the data, which is normally done by most convention neural network, the predicted results of Truth NN and Falsity NN are compared in order to provide the classification outcomes [11].

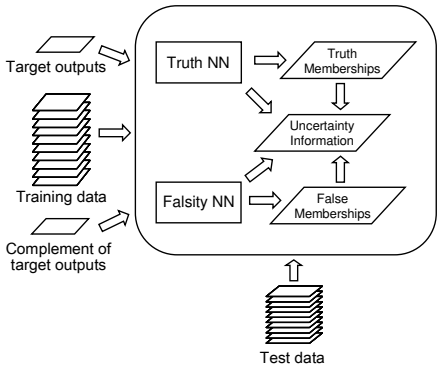

Fig. 1. Complementary neural network [11]

In order to apply CMTNN for under-sampling problem, Truth NN and Falsity NN are employed to detect and remove misclassification patterns from a training set. There are basically two ways to perform under-sampling [12].

Under-Sampling Technique I

a. The Truth and Falsity NNs are trained by truth and false membership values.

b. The prediction outputs *(Y)* on the training data *(T)* of both NNs are compared with the actual outputs *(O)*.

c. The misclassification patterns of Truth NN and Falsity NN $(M_{Truth}, M_{Falsity})$ are also detected if the prediction outputs and actual outputs are different.

For Truth NN : If $Y_{Truth\ i} \neq O_{Truth\ i}$ then $M_{Truth} \leftarrow M_{Truth} \ U \ \{T_i\}$ (2)

For Falsity NN : If $Y_{Falsity\ i} \neq O_{Falsity\ i}$ then $M_{Falsity} \leftarrow M_{Falsity} \ U \ \{T_i\}$ (3)

d. In the last step, the under-sampling for the new training set (T_c) is performed by eliminating the misclassification patterns detected by both the Truth NN (M_{Truth}) and Falsity NN $(M_{Falsity})$. $T_c \leftarrow T - (M_{Truth} \cap M_{Falsity})$ (4)

Under-Sampling Technique II

a. Repeat the step *a.* to *b.* of under-sampling technique I.

b. The under-sampling for the new training set (T_c) is performed by eliminating all misclassification patterns detected by the Truth NN (M_{Truth}) and Falsity NN $(M_{Falsity})$ respectively. $T_c \leftarrow T - (M_{Truth} \cup M_{Falsity})$ (5)

2.2 Synthetic Minority Over-sampling Technique (SMOTE)

SMOTE [3] is an over-sampling technique. This technique increases a number of new minority class instances by interpolation method. The minority class instances that lie together are identified before they are employed to form new minority class instances. This technique is able to generate synthetic instances rather than replicate minority class instances; therefore, it can avoid the over-fitting problem. The algorithm is described in Fig. 2.

O is the original data set
P is the set of positive instances (minority class instances)
For each instance *x* in *P*
    Find the *k*-nearest neighbors (minority class instances) to *x* in *P*
    Obtain *y* by randomizing one from *k* instances
    difference = *x* – *y*
    gap = random number between 0 and 1
    n = *x* + *difference* * *gap*
    Add *n* to *O*
End for

Fig. 2. The Synthetic Minority Oversampling Technique (SMOTE) [3]

2.3 The Proposed Combined Techniques

In order to obtain the advantages of using the combination between under-sampling and over-sampling techniques as presented in the literature [1] and [3], in this paper, CMTNN is applied as under-sampling while SMOTE is used for over-sampling. They are combined in order to better handle the imbalanced data problem. Four techniques can be derived by the combination as follows.

1. Under-sampling only the majority class using the CMTNN under-sampling technique I and then over-sampling the minority class using SMOTE technique
2. Under-sampling only the majority class using the CMTNN under-sampling technique II and then over-sampling the minority class using SMOTE technique
3. Over-sampling the minority class using SMOTE technique before under-sampling both classes using the CMTNN under-sampling technique I
4. Over-sampling the minority class using SMOTE technique before under-sampling both classes using the CMTNN under-sampling technique II

For all the proposed techniques mentioned above, the ratio between the minority and majority class instances after implementing SMOTE algorithm is 1:1.

3 Experiments and Results

Four data sets from the UCI machine learning repository [13] are used in the experiment. The data sets for binary classification problems include Pima Indians Diabetes data, German credit data, Haberman's Survival data, and SPECT heart data. These data sets are selected because they are imbalanced data sets with various ratios

between the minority class and the majority class. The characteristics of these four data sets are shown in Table I.

Table I. Characteristics of data sets used in the experiment.

Name of data set	No. of instances	No. of attributes	Minority class (%)	Majority class (%)
Pima Indians Diabetes data	768	8	34.90	65.10
German Credit data	1000	20	30.00	70.00
Haberman's Survival data	306	3	26.47	73.53
SPECT Heart data	267	22	20.60	79.40

For the purpose of establishing the classification model and testing it, each data set is first split into 80% training set and 20% test set. Furthermore, the cross validation method is used in order to reduce inconsistent results. Each data set will be randomly split ten times to form different training and test data sets. For the purpose of this study, the results of the ten experiments of each data set will be averaged.

In the experiment, after the training sets are applied by the proposed combined techniques, three different learning algorithms, which are ANN, SVM (kernel function = Radial Basis Function (RBF)), and k-NN (k=5) are used for the classification. The classification performance is then evaluated by G-mean and AUC. Furthermore, in order to compare the performance of the proposed techniques to others, the over-sampling technique, SMOTE, will be compared as the base technique. The other two under-sampling approaches, Tomek links [6] and ENN [7], are also used for this purpose. These comparison techniques are selected because they have been applied widely to the class imbalance problem [1], [9].

Table II. The results of G-Mean and AUC for each data set classified by ANN

	Pima Indian Diabetes data		German Credit data		Haberman's Survival data		SPECT Heart data	
Techniques	GM	AUC	GM	AUC	GM	AUC	GM	AUC
Original Data	70.12	0.8276	63.92	0.7723	33.11	0.5885	64.05	0.7590
a. ENN	72.64	0.8298	70.74	0.7794	50.45	0.6305	71.80	0.7895
b. Tomek links	73.11	0.8288	70.48	0.7793	51.88	0.6323	72.88	0.8178
c. SMOTE	74.30	0.8281	71.48	0.7777	58.60	0.6345	73.59	0.8241
d. Technique I (Majority) + SMOTE	75.55	0.8332	72.03	0.7855	60.00	0.6452	73.86	0.8374
e. Technique II (Majority) + SMOTE	74.53	0.8300	73.32	0.7873	62.78	0.6770	74.32	0.8273
f. SMOTE + Technique I	75.00	0.8285	71.52	0.7844	61.41	0.6653	73.00	0.8264
g. SMOTE + Technique II	74.96	0.8300	72.07	0.7860	58.59	0.6248	74.04	0.8373
Best technique	d	d	e	e	e	e	e	d
Second best	f	e & g	g	g	f	f	g	g

The experimental results in Table II, III and IV show that four proposed techniques combined CMTNN and SMOTE generally performs better than other

techniques, in terms of G-mean and AUC in each learning algorithm (ANN, SVM, and k-NN). They improve the performance significantly when comparing to the results of original data sets. The proposed techniques f (SMOTE + CMTNN technique I) can improve G-mean up to 45.41% on Haberman's Survival data classified by SVM. Moreover, technique g (SMOTE + CMTNN technique II) generally present the better technique in the experiments.

The results of the ANN classifier in Table II show that the combined technique d (CMTNN technique I (Majority) + SMOTE) and technique e (CMTNN technique II (Majority) + SMOTE) present the best results of G-mean and AUC. Technique g also presents the second best performance in most cases. The proposed combined techniques (technique d e f g) show the improvement significantly when comparing to the results of G-mean on original test sets from 5.43% to 29.67%. In addition, when the results of technique d. and e. are compared to the base technique (SMOTE), the results of G-mean show the improvement from 0.73% to 4.73%.

In Table III, SVM is employed as a classifier. The results show that technique g (SMOTE + CMTNN technique II) presents the best performance on two test sets. The significant improvement by technique g is up to 13.19% on German Credit data when compared to the base technique, SMOTE. ENN and Tomek links technique also perform well on some test sets. This is because they can broaden the margin between two classes by eliminating instances near the separating hyperplane [1].

Table III. The results of G-Mean and AUC for each data set classified by SVM

	Pima Indian Diabetes data		German Credit data		Haberman's Survival data		SPECT Heart data	
Techniques	GM	AUC	GM	AUC	GM	AUC	GM	AUC
Original Data	67.81	0.8294	56.78	0.7660	19.13	0.6520	71.81	0.7249
a. ENN	73.04	0.8281	70.01	0.7842	53.16	0.7105	77.15	0.7717
b. Tomek links	72.83	0.8231	70.73	0.7846	49.61	0.6982	76.72	0.7681
c. SMOTE	74.32	0.8247	58.03	0.7381	58.33	0.6336	71.59	0.7253
d. Technique I (Majority) + SMOTE	74.75	0.8144	60.03	0.7573	61.16	0.6505	73.08	0.7349
e. Technique II (Majority) + SMOTE	74.89	0.8177	66.84	0.7626	60.92	0.6732	74.80	0.7503
f. SMOTE + Technique I	74.11	0.8262	67.87	0.7805	64.54	0.6843	74.39	0.7466
g. SMOTE + Technique II	75.57	0.8306	71.22	0.7902	58.32	0.6204	75.33	0.7555
Best technique	g	g	g	g	f	a	a	a
Second best	e	Origin	b	b	d	b	b	b

In Table IV, k-NN (k=5) is used as a classifier. Technique g (SMOTE + CMTNN technique II) show the best and the second best performance in every test set. While Technique f (SMOTE + CMTNN technique I) show the best outcome in two test sets, ENN perform well only on SPECT Heart data.

In order to explain why the proposed combined techniques outperform other techniques, the characteristics of the both techniques need to be discussed. On one

hand, SMOTE technique gains the benefits of avoiding the over-fitting problem of the minority class by interpolating new minority class instances rather than duplicating the existing instances [1]. On the other hand, the misclassification analysis using CMTNN can enhance the quality of the training data by removing possible misclassification patterns from data sets.

Table IV. The results of G-Mean and AUC for each data set classified by k-NN (k=5)

	Pima Indian Diabetes data		German Credit data		Haberman's Survival data		SPECT Heart data	
Techniques	GM	AUC	GM	AUC	GM	AUC	GM	AUC
Original Data	65.27	0.7665	59.35	0.7483	40.11	0.5741	68.00	0.8121
a. ENN	71.15	0.7817	64.40	0.7566	46.47	0.5915	77.56	0.8369
b. Tomek links	72.06	0.7865	67.42	0.7625	47.57	0.5918	74.10	0.8148
c. SMOTE	71.78	0.7742	68.69	0.7518	55.82	0.5836	74.20	0.8005
d. Technique I (Majority) + SMOTE	72.11	0.7938	69.32	0.7572	56.28	0.5927	74.64	0.8264
e. Technique II (Majority) + SMOTE	73.17	0.7956	69.94	0.7686	57.50	0.6050	74.53	0.8030
f. SMOTE + Technique I	73.95	0.8104	72.35	0.7785	56.39	0.6226	74.13	0.8121
g. SMOTE + Technique II	73.42	0.8058	71.21	0.7719	59.30	0.6302	75.30	0.8179
Best technique	f	f	f	f	g	g	a	a
Second best	g	g	g	g	e	f	g	d

For generalization, when the proposed techniques are compared, technique g (SMOTE + CMTNN technique II) constantly presents the best or the second best in most cases among different classification algorithms. This is because when the training data is applied by SMOTE technique, it can create larger and less specific decision boundaries for the minority class [3]. Consequently, when the data is applied by CMTNN as under-sampling, the training data is eliminated all possible misclassification patterns detected by both the Truth NN and Falsity NN. Moreover, when a number of instances removed from the training sets are compared, it is found that misclassification instances eliminated by technique g are greater than other combined techniques. The lesser noise the training set retains the better performance the learning algorithm performs.

However, in some cases, for example Haberman's Survival data, technique g cannot gain the better results than other techniques. This is because it removes lots of instances from the training set. While technique g removes misclassification instances between 14% and 24% in other data sets, it eliminates instances up to 55% in Haberman's Survival data. As a consequence, a number of remaining instances of this data is not enough for the learning algorithms (ANN and SVM) to generalize the correct results. Therefore, in summary, although the combined technique g consistently presented better results in this paper, the number of instances removed by technique g is also a major constraint which is able to affect the classification performance on the class imbalanced problem.

4 Conclusions

This paper presents the proposed combined techniques to re-distribute the data in classes to solve the class imbalance problem. They are the integration of under-sampling techniques using Complementary Neural Network (CMTNN) and the over-sampling technique using Synthetic Minority Over-sampling Technique (SMOTE). The experiment employs three types of machine learning algorithms for classifying the test sets including ANN, SVM, and k-NN. The results of classification are evaluated and compared in terms of performance using the widely accepted measures for the class imbalance problem, which are G-mean and AUC. The results obtained from the experiment indicated that the proposed combined technique by SMOTE and CMTNN generally performs better than other techniques in most test cases

References

[1] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *SIGKDD Explorations Newsletter,* vol. 6, pp. 20-29, 2004.

[2] J. Laurikkala, "Improving identification of difficult small classes by balancing class distribution," in *Proceedings of the 8th Conference on AI in Medicine in Europe: Artificial Intelligence Medicine*: Springer-Verlag, 2001.

[3] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research,* vol. 16, pp. 321-357, 2002.

[4] Q. Gu, Z. Cai, L. Zhu, and B. Huang, "Data mining on imbalanced data sets," in *Advanced Computer Theory and Engineering, 2008. ICACTE '08. International Conference on*, 2008, pp. 1020-1024.

[5] T. D. Gedeon, P. M. Wong, and D. Harris, "Balancing bias and variance: Network topology and pattern set reduction techniques," in *From Natural to Artificial Neural Computation*. vol. 930: Springer Verlag, 1995, pp. 551-558.

[6] I. Tomek, "Two Modifications of CNN," *Systems, Man and Cybernetics, IEEE Transactions on,* vol. 6, pp. 769-772, 1976.

[7] D. L. Wilson, "Asymptotic properties of nearest neighbor rules using edited Data," *Systems, Man and Cybernetics, IEEE Transactions on,* vol. 2, pp. 408-421, 1972.

[8] T. D. Gedeon and T. G. Bowden, "Heuristic pattern reduction," in *International Joint Conference on Neural Networks*. vol. 2 Beijing, 1992, pp. 449-453.

[9] R. Barandela, J. S. Sanchez, V. Garcia, and E. Rangel, "Strategies for learning in class imbalance problems," *Pattern Recognition,* vol. 36, pp. 849-851, 2003.

[10] P. Kraipeerapun, C. C. Fung, and S. Nakkrasae, "Porosity prediction using bagging of complementary neural networks," in *Advances in Neural Networks – ISNN 2009*, 2009, pp. 175-184.

[11] P. Kraipeerapun and C. C. Fung, "Binary classification using ensemble neural networks and interval neutrosophic sets," *Neurocomput.,* vol. 72, pp. 2845-2856, 2009.

[12] P. Jeatrakul, K. W. Wong, and C. C. Fung, "Data cleaning for classification using misclassification analysis," *Journal of Advanced Computational Intelligence and Intelligent Informatics,* vol. 14, no. 3, pp. 297-302, 2010.

[13] A. Asuncion and D. J. Newman, "UCI Machine Learning Repository," University of California, Irvine, School of Information and Computer Sciences, 2007.