Improvement of Decision Trees Based on the Quality Control of Artificial Instances of Over-sampling

Hyontai Sug

Abstract**—** In order to surmount the problem of neglecting minor data instances in data mining models of comprehension like decision trees or rule learners, over-sampling technique based on SMOTE was considered for validation. The quality of the artificially generated instances is validated by resorting to different and more reliable data mining algorithms other than C4.5 or RIPPER, which are the two target data mining algorithms of comprehension for improvement. On the condition that more reliable or accurate data mining algorithms are available for target data sets, they were used to check the quality of the generated over-sampled instances. The validity of the suggested idea was checked by experiment using two data sets in medicine domain, where the understandability of data mining models is important, and the experiment generated very good results.

Keywords**—**Decision trees, rule learner, classification, data selection.

I. Introduction

COMPREHENSIBILITY of the result of data mining is an important issue, because we may want to utilize found knowledge models for some important areas [1]. For example, medicine area highly requires the understanding of the found knowledge, because it is related to human life. There are several data mining algorithms for understandability. Among them decision trees and rule set learners can be representatives [2]. There are many examples that used such data mining algorithms successfully [3, 4, 5]. Even though decision trees are considered one of good data mining tools, they may not generate good classification performance for a minor class, because they are trained to achieve a maximum accuracy for the whole data set. But, in real world data sets for data mining, a minor class may be more important than the others, for example, in medical data [6]. For more accurate classification of these minor classes in decision trees over-sampling may be applied. But, simple over-sampling may have limited effect only, because the same instances are supplied multiple times for training. On the other hand, we may supply some very similar data instances of the minor classes by generating the instances artificially. SMOTE[7] is one of the representative over-sampling method that generates artificial instances of minor classes. The method generates artificial instances based on K-nearest neighbors algorithm, and success was reported for a decision tree algorithm and rule generator. But, we know that incorrect training instances are easy to lead to classifiers of lower performance. So, supplying possibly correct instances is an important task for better classifiers. In this paper we want to check the quality of the artificially generated data instances by SMOTE empirically so that we may find better classifiers like decision trees or rule sets by supplying good artificial instances. This paper is the extension of previous work in SCI 2014 [8]. In section 2 we discuss our experiment method, and in section 3 conclusions are provided.

.

II. Empirical Procedure

A. Experiment Method

There are several data mining algorithms that generate knowledge models of understandability. Among them C4.5 [9] and RIPPER [10] can be representative data mining algorithms of classification to generate decision trees and rule sets respectively [11]. SMOTE tries to generate artificial instances of a minor class as a way of over-sampling to build better decision tree of C4.5 and rule set of RIPPER. The artificial instances are made based on K-nearest neighbors algorithm and randomization on continues values of related attributes of neighboring instances. But, we may doubt that there is some possibility that the quality of the newly generated instances may not be good as expected because of the randomization on the continuous values. So, we want to check the class of artificially generated instances by SMOTE using more a accurate classifier, if it is available.

In the following experiments, we first check the accuracy of three different data mining algorithms, C4.5, RIPPER, and a more accurate classifier X, using the original data sets and the original data set plus artificial instances generated from SMOTE. Experiments were performed using medicine data sets called BUPA liver disorder and echocardiogram in the UCI machine learning repository [12]. For better objectivity, the experiment is based on 10-fold cross validation and a data mining tool called Weka [13] is used. Weka is a comprehensive data mining tool.

H. Sug is with Division of Computer and Information Engineering, Dongseo University, Busan, 617-716 Korea (phone: +82-51-320-1733; fax: +82-51-327-8955; e-mail: sht@ gdsu.dongseo.ac.kr).

B. *Experiment on BUPA Liver Disorder Data Set*

The data set contains two classes, and each class has 145 and 200 instances. So, the class having 145 instances is a minor class. Let it be class 1. The data set has six continuous attributes. Table 1 shows the accuracy of three different data mining algorithms, C4.5, RIPPER, and random forests [14] for the data set. Default parameters are used for C4.5 and RIPPER. The parameters of 100 trees and one are given as the parameter of the number of trees and the number of attributes to be used in random selection for the random forests respectively. In the table TP rate means true positive rate.

Table 1. Accuracy of the three different data mining algorithms for the BUPA liver disorder data

		C4.5	RIPPER	Random Forests
Accuracy(%)		68.6957	64.6377	75.942
TP rate	Class 1	0.531	0.469	0.593
	Class 2	0.8	0.775	0.88

The following is the decision tree of C4.5 for the BUPA liver disorder data. The size of the tree is 51, and the number of leaves is 26. The numbers in parentheses in the terminal nodes of tree consist of two parts. The first number is the number of instances belonging to the class of the terminal node, and the second number is the number of instances not belonging to the class of the terminal node. If there is only one number in the parentheses, it is the first one.

```
gammagt <= 20
|  sgpt <= 19
|  |  gammagt <= 7: 1 (4.0)
|  |  gammagt > 7
|  |  |  alkphos <= 77: 2 (42.0/6.0)
|  |  |  alkphos > 77
|  |  |  |  mcv <= 90
|  |  |  |  |  sgpt <= 16: 2 (3.0)
|  |  |  |  |  sgpt > 16
|  |  |  |  |  |  sgpt <= 17: 1 (2.0)
|  |  |  |  |  |  sgpt > 17
|  |  |  |  |  |  |  mcv <= 89: 2 (2.0)
|  |  |  |  |  |  |  mcv > 89: 1 (2.0)
|  |  |  |  mcv > 90: 1 (5.0)
|  sgpt > 19
|  |  sgot <= 20
|  |  |  drinks <= 3: 1 (31.0/1.0)
|  |  |  drinks > 3
|  |  |  |  sgpt <= 23: 2 (3.0)
|  |  |  |  sgpt > 23: 1 (5.0)
|  |  sgot > 20
|  |  |  drinks <= 5
|  |  |  |  sgpt <= 26: 2 (21.0/8.0)
|  |  |  |  sgpt > 26: 1 (15.0/3.0)
|  |  |  drinks > 5: 1 (5.0)
gammagt > 20
|  drinks <= 5
|  |  drinks <= 3
|  |  |  alkphos <= 65: 2 (42.0/6.0)
|  |  |  alkphos > 65
|  |  |  |  sgot <= 24
|  |  |  |  |  gammagt <= 29: 1 (12.0/1.0)
|  |  |  |  |  gammagt > 29
|  |  |  |  |  |  mcv <= 87: 2 (7.0/1.0)
|  |  |  |  |  |  mcv > 87
|  |  |  |  |  |  |  mcv <= 92: 1 (9.0/2.0)
|  |  |  |  |  |  |  mcv > 92: 2 (2.0)
|  |  |  |  sgot > 24
|  |  |  |  |  sgpt <= 39: 2 (12.0)
|  |  |  |  |  sgpt > 39
|  |  |  |  |  |  sgpt <= 48: 1 (7.0/2.0)
|  |  |  |  |  |  sgpt > 48: 2 (4.0)
|  |  drinks > 3: 2 (41.0/3.0)
|  drinks > 5
|  |  drinks <= 12
|  |  |  sgpt <= 21: 2 (10.0/1.0)
|  |  |  sgpt > 21
|  |  |  |  sgot <= 22: 1 (11.0/1.0)
|  |  |  |  sgot > 22: 2 (44.0/18.0)
|  |  drinks > 12: 1 (4.0)
```

The following is a rule set generated by RIPPER. The number of rules is three.

```
(gammagt <= 21) and (sgpt >= 20) => class=1 (85.0/22.0)
(drinks >= 6) and (sgpt >= 36) => class=1 (35.0/12.0)
 => class=2 (225.0/59.0)
```

After generating the three data mining models based on the original data set, over-sampling rate of 100%, 200%, 300%, and 400% is applied for the minor class of class 1 using SMOTE. So, additional instances of 145, 290, 335, and 580 of class 1 are added to the original data set to make new training data sets for each respective over-sampling rate. Table 2 shows the result of the experiment.

Table 2. Accuracy of the three different data mining algorithms for the BUPA liver disorder data in different over-sampling rates

Over-sampling rate			C4.5	RIPPER	Random Forests
100%	Accuracy(%)		72.8571	74.4898	81.8367
	TP rate	Class 1	0.8	0.8	0.869
		Class 2	0.625	0.665	0.745
200%	Accuracy(%)		74.0157	74.6457	83.622
	TP rate	Class 1	0.841	0.848	0.94
		Class 2	0.52	0.525	0.61
300%	Accuracy(%)		77.5641	77.9487	83.4615
	TP rate	Class 1	0.876	0.878	0.953
		Class 2	0.485	0.495	0.49

400%	Accuracy(%)		83.1351	83.3514	85.6216
	TP rate	Class 1	0.927	0.934	0.975
		Class 2	0.485	0.47	0.425

We can see some positive effect of the over-sampling from table 2. In order to check the quality of the over-sampled data by SMOTE, all the over-sampled instances of SMOTE are checked by the more accurate classifier, the random forests. The random forests trained by the original data is used. While 1,163 distinct instances are checked to belong to true positive, the other 274 distinct instances are checked to belong to false positive among the over-sampled instances. Using these two groups of over-sampled instances and the original data set, two more experiment were run. Table 3 shows the result of the experiment using over-sampled instances of true positive plus the original data set.

Table 3. Accuracy of the three different data mining algorithms for over-sampled instances of true positive plus the original BUPA liver disorder data

		C4.5	RIPPER	Random Forests
Accuracy(%)		89.0584	88.9257	91.1141
TP rate	Class 1	0.952	0.952	0.986
	Class 2	0.49	0.48	0.42

The following is the decision tree for the additional 1,163 instances of true positive plus the original data. So, the number of instances for class 1 and class 2 becomes 1,308 and 200 respectively.

```
sgpt <= 18
|   alkphos <= 64.133435
|   |   gammagt <= 7.865058: 1 (3.0)
|   |   gammagt > 7.865058
|   |   |   gammagt <= 14.485685: 2 (14.0)
|   |   |   gammagt > 14.485685
|   |   |   |   gammagt <= 15.711306: 1 (6.0)
|   |   |   |   gammagt > 15.711306
|   |   |   |   |   gammagt <= 51.406185: 2 (17.0)
|   |   |   |   |   gammagt > 51.406185
|   |   |   |   |   |   gammagt <= 54.809379: 1 (2.0)
|   |   |   |   |   |   gammagt > 54.809379: 2 (2.0)
|   alkphos > 64.133435
|   |   mcv <= 90.00546
|   |   |   drinks <= 3.276505
|   |   |   |   sgpt <= 17.981779
|   |   |   |   |   alkphos <= 72.260495: 2 (5.0/1.0)
|   |   |   |   |   alkphos > 72.260495: 1 (20.0/1.0)
|   |   |   |   sgpt > 17.981779: 2 (3.0)
|   |   |   drinks > 3.276505: 2 (11.0/1.0)
|   |   mcv > 90.00546
|   |   |   sgot <= 19.911888: 1 (40.0)
|   |   |   sgot > 19.911888
|   |   |   |   alkphos <= 69.434712: 2 (3.0)
|   |   |   |   alkphos > 69.434712: 1 (22.0/1.0)
sgpt > 18
|   gammagt <= 20.922489
|   |   mcv <= 88
|   |   |   drinks <= 3.781926
|   |   |   |   sgot <= 19.969756: 1 (116.0/1.0)
|   |   |   |   sgot > 19.969756
|   |   |   |   |   drinks <= 0.54402
|   |   |   |   |   |   alkphos <= 71.274603
|   |   |   |   |   |   |   gammagt <= 10.481602: 1 (2.0)
|   |   |   |   |   |   |   gammagt > 10.481602: 2 (8.0)
|   |   |   |   |   |   alkphos > 71.274603: 1 (20.0/1.0)
|   |   |   |   |   drinks > 0.54402: 1 (29.0)
|   |   |   drinks > 3.781926
|   |   |   |   sgpt <= 24.492435: 2 (7.0)
|   |   |   |   sgpt > 24.492435: 1 (5.0/1.0)
|   |   mcv > 88: 1 (573.0/9.0)
|   gammagt > 20.922489
|   |   mcv <= 86
|   |   |   gammagt <= 34.457056
|   |   |   |   mcv <= 83.472404: 1 (10.0)
|   |   |   |   mcv > 83.472404
|   |   |   |   |   sgpt <= 27: 2 (6.0)
|   |   |   |   |   sgpt > 27
|   |   |   |   |   |   alkphos <= 65: 2 (3.0)
|   |   |   |   |   |   alkphos > 65: 1 (9.0/1.0)
|   |   |   gammagt > 34.457056
|   |   |   |   drinks <= 7.041949
|   |   |   |   |   alkphos <= 79: 2 (13.0)
|   |   |   |   |   alkphos > 79
|   |   |   |   |   |   alkphos <= 85: 1 (2.0)
|   |   |   |   |   |   alkphos > 85: 2 (2.0)
|   |   |   |   drinks > 7.041949: 1 (2.0)
|   |   mcv > 86
|   |   |   sgot <= 22.93147
|   |   |   |   alkphos <= 58.036529
|   |   |   |   |   drinks <= 1.140679: 2 (3.0)
|   |   |   |   |   drinks > 1.140679
|   |   |   |   |   |   gammagt <= 44.420089
|   |   |   |   |   |   |   sgpt <= 21.295916: 2 (4.0/1.0)
|   |   |   |   |   |   |   sgpt > 21.295916: 1 (31.0/2.0)
|   |   |   |   |   |   gammagt > 44.420089: 2 (2.0)
|   |   |   |   alkphos > 58.036529: 1 (235.0/13.0)
|   |   |   sgot > 22.93147
|   |   |   |   sgpt <= 35.039271
|   |   |   |   |   drinks <= 6
|   |   |   |   |   |   mcv <= 89.097902: 2 (10.0)
|   |   |   |   |   |   mcv > 89.097902
|   |   |   |   |   |   |   mcv <= 91.496043
|   |   |   |   |   |   |   |   drinks <= 3.520671
|   |   |   |   |   |   |   |   |   drinks <= 0.789578
|   |   |   |   |   |   |   |   |   |   gammagt <= 39: 2 (3.0)
|   |   |   |   |   |   |   |   |   |   gammagt > 39: 1 (3.0)
|   |   |   |   |   |   |   |   |   drinks > 0.789578: 1 (8.0)
|   |   |   |   |   |   |   |   drinks > 3.520671: 2 (4.0)
```

```
| | | | | | | mcv > 91.496043
| | | | | | | | gammagt <= 34
| | | | | | | | | drinks <= 5.340442: 2 (7.0/1.0)
| | | | | | | | | drinks > 5.340442: 1 (2.0)
| | | | | | | | gammagt > 34: 2 (13.0)
| | | | | drinks > 6
| | | | | | alkphos <= 70.387946: 2 (3.0)
| | | | | | alkphos > 70.387946
| | | | | | | mcv <= 99
| | | | | | | | mcv <= 97.166303
| | | | | | | | | mcv <= 93.639192: 1 (8.0/1.0)
| | | | | | | | | mcv > 93.639192: 2 (2.0)
| | | | | | | | mcv > 97.166303: 1 (13.0)
| | | | | | | mcv > 99: 2 (2.0)
| | | | sgpt > 35.039271
| | | | | sgot <= 44.812937
| | | | | | drinks <= 2.143685
| | | | | | | alkphos <= 64.460977: 2 (4.0)
| | | | | | | alkphos > 64.460977
| | | | | | | | mcv <= 89.088071
| | | | | | | | | alkphos <= 80.211483: 1 (2.0)
| | | | | | | | | alkphos > 80.211483: 2 (2.0)
| | | | | | | | mcv > 89.088071: 1 (6.0)
| | | | | | drinks > 2.143685: 1 (170.0/9.0)
| | | | | sgot > 44.812937
| | | | | | mcv <= 94.193511: 2 (8.0)
| | | | | | mcv > 94.193511
| | | | | | | gammagt <= 144.782125: 1 (6.0/1.0)
| | | | | | | gammagt > 144.782125: 2 (2.0)
```

The size of the tree is 109, and the number of leaves is 55. The following is a rule set generated by RIPPER.

(gammagt >= 22) and (alkphos <= 65) and (drinks <= 4) and (drinks <= 1) => class=2 (25.0/3.0)
(sgpt <= 19) and (alkphos <= 65) and (sgot >= 17) => class=2 (33.0/5.0)
(mcv <= 88) and (sgpt <= 19) and (alkphos <= 70) => class=2 (7.0/0.0)
(gammagt >= 28) and (mcv <= 89) and (gammagt >= 42) => class=2 (29.0/9.0)
(gammagt >= 23) and (sgot >= 45) => class=2 (18.0/5.0)
(gammagt >= 30) and (sgpt <= 34) and (sgot >= 24) and (alkphos <= 92) => class=2 (27.0/7.0)
(sgpt <= 22) and (drinks >= 4) and (mcv <= 90) => class=2 (20.0/5.0)
(sgpt <= 15) and (alkphos <= 72) => class=2 (13.0/5.0)
=> class=1 (1336.0/67.0)

The total number of rules is nine.

Table 4 shows the result of the experiment using over-sampled instances of false positive plus the original data set.

Table 4. Accuracy of the three different data mining algorithms for the over-sampled instances of false positive plus the original BUPA liver disorder data

		C4.5	RIPPER	Random Forests
Accuracy(%)		70.5977	73.3441	79.483
TP rate	Class 1	0.924	0.895	0.969
	Class 2	0.25	0.395	0.43

The following is the decision tree for the additional 274 instances of false positive plus the original data. So, the number of instances for class 1 and class 2 becomes 419 and 200 respectively.

```
mcv <= 87
| sgot <= 34
| | sgpt <= 18.009007
| | | drinks <= 2.279779
| | | | gammagt <= 15.628985
| | | | | sgpt <= 14.492658: 2 (3.0)
| | | | | sgpt > 14.492658: 1 (4.0)
| | | | gammagt > 15.628985: 2 (5.0)
| | | drinks > 2.279779: 2 (10.0)
| | sgpt > 18.009007: 1 (103.0/38.0)
| sgot > 34: 2 (8.0)
mcv > 87
| sgpt <= 21
| | gammagt <= 20.835356
| | | sgpt <= 9: 2 (5.0)
| | | sgpt > 9: 1 (94.0/26.0)
| | gammagt > 20.835356: 2 (40.0/15.0)
| sgpt > 21
| | sgot <= 44.543404: 1 (328.0/67.0)
| | sgot > 44.543404
| | | drinks <= 5.481014: 2 (7.0)
| | | drinks > 5.481014
| | | | sgpt <= 87: 2 (7.0/1.0)
| | | | sgpt > 87: 1 (5.0)
```

The size of the tree is 25, and the number of leaves is 13. The following is a rule set generated by RIPPER.

(sgpt <= 19) and (alkphos <= 64) and (alkphos >= 55) => class=2 (30.0/5.0)
(mcv <= 87) and (drinks >= 4) and (sgpt <= 29) => class=2 (18.0/0.0)
(mcv <= 88) and (gammagt >= 42) => class=2 (26.0/7.0)
(sgot >= 23) and (sgpt <= 35) and (gammagt >= 21) and (sgpt <= 27) and (drinks <= 8) => class=2 (27.0/4.0)
(drinks <= 0.5) and (sgot >= 28) and (alkphos <= 74) => class=2 (11.0/1.0)
(alkphos >= 72) and (sgot >= 47) => class=2 (8.0/0.0)
=> class=1 (499.0/97.0)

The total number of rules is seven. Table 5 shows the summary of the two experiments in table 3 and table 4 for easy comparison.

Table 5. The summary of the result of experiments for the two different groups of over-sampled instances for the BUPA liver disorder data

	Over-sampled instances of TP	Over-sampled instances of FP
Size of data set	1,508	619
Size of class 1	1,308	419
Size of class 2	200	200
Accuracy of C4.5	89.0584%	70.5979%
Size of the tree	109	25
Accuracy of RIPPER	88.9257%	73.3441%
Number of the rules	9	7

If we compare the result of the experiment, we can find that the true positive instances checked by the random forests are doing better than the false positive instances. Note that adding smaller number of over-sampled instances of class 1 may affect smaller decrease in TP rate of class 2 as we can see in table 2. But, if we look at true positive rate of class 2 in table 4, the three values are worse than those values of over-sampling rate of 100% or 200% in table 2. On the contrary, over-sampled instances of true positive generated similar true positive rate with those of over-sampling rate of 400% in table 2, while the accuracy of the three algorithms are better.

C. Experiment on the Echocardiogram Data Set

Originally echocardiogram data set contains two classes of 74 instances, and the other 58 instances have no classes. So, one more class is added as 'unknown' for convenience. As a result, a new data set with three classes is used for the experiment, and each class has 50, 24, and 58 instances for class 0, class 1, and class u respectively. So, the class having 24 instances is a minor class, which is class 1. The data set has twelve continuous attributes. Table 6 shows the accuracy of three different data mining algorithms, C4.5, RIPPER, and LMT [15] for the data set.

Table 6. Accuracy of the three different data mining algorithms for echocardiogram data

		C4.5	RIPPER	LMT
Accuracy(%)		54.5455	63.6364	70.4545
TP rate	Class 0	0.6	0.82	0.8
	Class 1	0.583	0.792	0.625
	Class u	0.483	0.414	0.655

The following is the decision tree of C4.5. The attributes are named for compact representation of the generated knowledge models. Table 7 has the original name of each attribute.

Table 7. The meaning of each attribute Ai

Attribute	The original name of attribute
A1	Survival
A2	Still-alive
A3	Age-at-heart-attack
A4	Pericardial-effusion
A5	Fractional-shortening
A6	EPSS
A7	LVDD
A8	Wall-motion-score
A9	Wall-motion-index
A10	Mult
A11	Name
A12	Group

```
A1 <= 10
|  A2 <= 0: u (3.02/1.0)
|  A2 > 0
|  |  A12 <= 1: 1 (9.27/0.27)
|  |  A12 > 1
|  |  |  A8 <= 17.83
|  |  |  |  A4 <= 0
|  |  |  |  |  A6 <= 10.3: 1 (2.69/0.4)
|  |  |  |  |  A6 > 10.3: u (9.1/1.0)
|  |  |  |  A4 > 0
|  |  |  |  |  A10 <= 0.812: 1 (2.35/0.02)
|  |  |  |  |  A10 > 0.812: u (2.35/0.33)
|  |  |  A8 > 17.83: 1 (8.79/0.44)
A1 > 10
|  A12 <= 1: 0 (19.29/4.29)
|  A12 > 1
|  |  A4 <= 0
|  |  |  A6 <= 7: u (22.21/4.39)
|  |  |  A6 > 7
|  |  |  |  A3 <= 53: u (5.55/1.0)
|  |  |  |  A3 > 53: 0 (36.29/12.68)
|  |  A4 > 0
|  |  |  A10 <= 0.643: 0 (3.03/0.03)
|  |  |  A10 > 0.643
|  |  |  |  A7 <= 4.58: 0 (2.82/0.82)
|  |  |  |  A7 > 4.58: u (5.23)
```

The size of the tree is 27, and the number of leaves is 14. The following is a rule set generated by RIPPER.

(A1 <= 7) and (A8 >= 18) => class=1 (12.0/0.0)
(A2 >= 1) and (A9 >= 1.36) and (A3 >= 61) => class=1 (12.0/3.0)
(A1 <= 5) and (A1 >= 2) => class=1 (4.0/1.0)
(A12 <= 1) => class=0 (15.0/0.0)
(A12 >= 2) and (A1 >= 36) and (A7 >= 3.59) => class=0 (14.0/1.0)
=> class=u (75.0/22.0)

The total number of rules is six.

Over-sampling rate of 100%, 200%, 300%, and 400% is applied for the minor class using SMOTE. So, additional instances of 24, 48, 72, and 96 of class 1 are added to the original data set for each respective over-sampling rate. Table 8 shows the result of the experiment.

Table 8. Accuracy of the three different data mining algorithms for echocardiogram data in different over-sampling rates

Over-sampling rate			C4.5	RIPPER	LMT
100%	Accuracy(%)		64.7436	71.7949	67.3077
	TP rate	Class 0	0.64	0.76	0.64
		Class 1	0.583	0.792	0.625
		Class u	0.483	0.414	0.655
200%	Accuracy(%)		72.2222	75.0	73.8889
	TP rate	Class 0	0.66	0.9	0.76
		Class 1	0.889	0.931	0.875
		Class u	0.569	0.397	0.522
300%	Accuracy(%)		70.5882	79.902	82.3529
	TP rate	Class 0	0.56	0.9	0.76
		Class 1	0.906	0.969	0.969
		Class u	0.5	0.431	0.638
400%	Accuracy(%)		78.0702	81.1404	83.7719
	TP rate	Class 0	0.62	0.92	0.78
		Class 1	0.95	0.983	0.638
		Class u	0.569	0.362	0.586

We can see some positive effect of over-sampling from table 8. As before, in order to check the quality of the over-sampled data by SMOTE, all the over-sampled instances of SMOTE are checked by the more accurate classifier, LMT. The LMT trained by the original data is used. While 198 distinct instances are checked to belong to true positive, the other 35 distinct instances are checked to belong to false positive. Using these two groups of over-sampled instances and the original data set, two more experiment were run. Table 9 shows the result of the experiment using the over-sampled instances of true positive plus the original data set.

Table 9. Accuracy of the three different data mining algorithms for over-sampled instances of true positive plus the original echocardiogram data

		C4.5	RIPPER	LMT
Accuracy(%)		84.8485	86.3636	87.5758
TP rate	Class 0	0.6	0.92	0.78
	Class 1	0.986	0.968	0.973
	Class u	0.534	0.414	0.586

The following is the decision tree for the additional 198 instances of true positive plus the original data. So, the number of instances for class 0, class 1, and class u becomes 50, 222, and 58 respectively.

```
A2 <= 0
|  A1 <= 38
|  |  A12 <= 1.478174: 0 (10.97/2.97)
|  |  A12 > 1.478174
|  |  |  A4 <= 0.463744
|  |  |  |  A6 <= 7.01939: u (18.29/1.72)
|  |  |  |  A6 > 7.01939
|  |  |  |  |  A10 <= 0.928
|  |  |  |  |  |  A8 <= 15.7296
|  |  |  |  |  |  |  A10 <= 0.669913: 0 (7.15/1.51)
|  |  |  |  |  |  |  A10 > 0.669913
|  |  |  |  |  |  |  |  A6 <= 12.036545
|  |  |  |  |  |  |  |  |  A5 <= 0.253506: 0 (6.25/1.23)
|  |  |  |  |  |  |  |  |  A5 > 0.253506: u (3.42/0.35)
|  |  |  |  |  |  |  |  A6 > 12.036545: u (7.22/0.27)
|  |  |  |  |  |  A8 > 15.7296: u (4.5)
|  |  |  |  |  A10 > 0.928: 0 (4.01/0.01)
|  |  |  A4 > 0.463744
|  |  |  |  A10 <= 0.669913: 0 (2.01/0.01)
|  |  |  |  A10 > 0.669913
|  |  |  |  |  A7 <= 4.58: 0 (2.85/0.85)
|  |  |  |  |  A7 > 4.58: u (3.54)
|  A1 > 38
|  |  A6 <= 5.9: u (3.19/1.18)
|  |  A6 > 5.9: 0 (14.87/0.05)
A2 > 0
|  A1 <= 12
|  |  A12 <= 1.999737: 1 (194.7/0.83)
|  |  A12 > 1.999737
|  |  |  A8 <= 17.848927
|  |  |  |  A6 <= 8.7: 1 (6.99/0.65)
|  |  |  |  A6 > 8.7
|  |  |  |  |  A10 <= 0.818417
|  |  |  |  |  |  A5 <= 0.17222: u (4.78/1.31)
|  |  |  |  |  |  A5 > 0.17222: 1 (3.16/0.54)
|  |  |  |  |  A10 > 0.818417: u (5.63/0.22)
|  |  |  A8 > 17.848927: 1 (18.27/0.8)
|  A1 > 12
|  |  A4 <= 0.519236: 0 (6.19/1.19)
|  |  A4 > 0.519236: u (2.01)
```

The size of the tree is 41, and the number of leaves is 21. The following is a rule set generated by RIPPER for the over-sampled true positive instances plus the original data. It consists of five rules.

(A1 >= 12) and (A12 >= 1) => class=0 (71.0/23.0)
(A1 >= 7.5) => class=u (28.0/4.0)
(A12 >= 2) and (A9 <= 1.41) and (A5 <= 0.22) => class=u (7.0/1.0)
(A10 <= 0.28) => class=u (2.0/0.0)
=> class=1 (222.0/3.0)

Table 10 shows the result of the experiment using over-sampled instances of false positive plus the original data set.

Table 10. Accuracy of the three different data mining algorithms for over-sampled instances of false positive plus the original echocardiogram data

		C4.5	RIPPER	LMT
Accuracy(%)		70.0599	74.2525	72.4551
TP rate	Class 0	0.56	0.94	0.66
	Class 1	0.932	0.915	0.915
	Class u	0.586	0.397	0.586

The following is the decision tree for the additional 35 instances of false positive plus the original data. So, the number of instances for class 0, class 1, and class u becomes 50, 59, and 58 respectively. The following is the decision tree of C4.5.

```
A2 <= 0
|  A12 <= 1.48626
|  |  A10 <= 0.884189
|  |  |  A5 <= 0.225: u (2.28)
|  |  |  A5 > 0.225: 0 (5.52/1.52)
|  |  A10 > 0.884189: 0 (9.0)
|  A12 > 1.48626
|  |  A1 <= 38
|  |  |  A4 <= 0.499229
|  |  |  |  A6 <= 7: u (18.15/1.72)
|  |  |  |  A6 > 7
|  |  |  |  |  A10 <= 0.928
|  |  |  |  |  |  A8 <= 15.67
|  |  |  |  |  |  |  A10 <= 0.643: 0 (7.14/1.5)
|  |  |  |  |  |  |  A10 > 0.643
|  |  |  |  |  |  |  |  A6 <= 12
|  |  |  |  |  |  |  |  |  A5 <= 0.253: 0 (6.24/1.22)
|  |  |  |  |  |  |  |  |  A5 > 0.253: u (3.36/0.35)
|  |  |  |  |  |  |  |  A6 > 12: u (7.11/0.27)
|  |  |  |  |  |  A8 > 15.67: u (4.48)
|  |  |  |  |  A10 > 0.928: 0 (4.03/0.03)
|  |  |  A4 > 0.499229
|  |  |  |  A10 <= 0.643: 0 (2.01/0.01)
|  |  |  |  A10 > 0.643
|  |  |  |  |  A7 <= 4.58: 0 (2.83/0.83)
|  |  |  |  |  A7 > 4.58: u (3.46)
|  |  A1 > 38: 0 (12.91/1.91)
A2 > 0
|  A1 <= 12
|  |  A12 <= 1.999737: 1 (37.87/0.54)
|  |  A12 > 1.999737
|  |  |  A6 <= 8.716833: 1 (10.85/0.83)
|  |  |  A6 > 8.716833
|  |  |  |  A8 <= 19
|  |  |  |  |  A10 <= 0.812
|  |  |  |  |  |  A5 <= 0.17: u (4.87/1.38)
|  |  |  |  |  |  A5 > 0.17: 1 (3.07/0.52)
|  |  |  |  |  A10 > 0.812: u (6.01/0.29)
|  |  |  |  A8 > 19: 1 (6.96/0.31)
|  A1 > 12
|  |  A4 <= 0.861196: 0 (6.83/1.83)
|  |  A4 > 0.861196: u (2.01)
```

The size of the tree is 43, and the number of leaves is 22. The data set generated similarly sized tree compared to the true positive over-sampled instances plus the original data, even though it has smaller number of instances of class1. The following is a rule set generated by RIPPER for the over-sampled true positive instances plus the original data. It consists of five rules.

```
(A1 >= 12) and (A12 >= 1) and (A6 >= 7.1) => class=0 (42.0/9.0)
(A1 >= 10) and (A3 >= 66) => class=0 (16.0/5.0)
(A1 >= 10) => class=u (38.0/6.0)
(A12 >= 2) and (A3 >= 67) and (A3 <= 77) => class=u (7.0/0.0)
 => class=1 (64.0/6.0)
```

Table 11 shows the summary of the experiment described in table 9 and table 10 for easy comparison.

Table 11. The summary of the result of experiments for two different groups of over-sampled instances for the echocardiogram data

	Over-sampled instances of TP	Over-sampled instances of FP
Size of data set	330	167
Size of class 0	50	50
Size of class 1	222	59
Size of class u	58	58
Accuracy of C4.5	84.8485%	70.0599%
Size of the tree	41	43
Accuracy of RIPPER	86.3636%	74.2525%
Number of the rules	5	5

If we compare the result of the experiment, we can find that the true positive instances checked by LMT are doing better than the false positive instances by LMT. More encouraging results are the size of the tree and the number of rules. The over-sampled data in true positive do not generate a bigger tree or more rules, even though the size of training data set has been increased. This implies that the quality of the data set is very good for the target data mining algorithms of C4.5 and IPPER. Note that adding smaller number of over-sampled instances of class 1 may affect some change in TP rate of class 0 or class u that belong to major classes as we can see in table 9 and table 10.

III. Conclusion

Data mining algorithms are made to achieve predictive ability as accurately as possible, so data instances in minority group are often neglected, because the minority group often do not have enough data instances for accurate prediction. In order to surmount the problem, some over-sampling technique that generates artificial instances in the minority group may be used, and SMOTE has been considered a good technique for that purpose. But, even though the instances are generated based on the well known nearest neighbors algorithm, it might be possible that the quality of the artificially generated instances is not as good as expected. In this paper we showed how we may surmount the problem by resorting to a different and more reliable data mining algorithm other than C4.5 or RIPPER, which are two target data mining algorithms for improvement. More reliable or accurate data mining algorithms were used to check the quality of the generated over-sampled instances. The validity of the suggested idea was checked by experiment using two data sets for liver and heart in medicine domain, where the understandability of the data mining models is important, and the experiment showed very good results.

ACKNOWLEDGMENT

H. Sug thanks professor Douglas D. Dankel II who worked for the department of computer and information science and engineering, University of Florida, USA for his invaluable comments about the idea.

REFERENCES

[1] U. Fayyad, G. Piatetsky-Shapiro, P. Smith, "From Data Mining to Knowledge Discovery in Databases," *AI Magazine*, Fall 1996, pp. 37-54.

[2] J. Han, M. Kamber, J. Pei, *Data Mining: concepts and techniques*. Morgan Kaufmann Publishers, Inc., 2011.

[3] Y. Hui, Z. Longqun, and L. Xianwen, "Classification of Wetland from TM imageries based on Decision Tree", *WSEAS Transactions on Information Science and Applications*, issue 7, vol. 6, July 2009, pp. 1155-1164.

[4] A. Kumar, S. Kumar, "Decision Tree based Learning Approach for Identification of Operating System Processes," *WSEAS Transactions on Computers*, vol. 13, 2014, pp. 277-288.

[5] M.M. Mazid, A.B.M.S. Ali, K.S. Tickle, "Input space reduction for Rule Based Classification", *WSEAS Transactions on Information Science and Applications*, issue 6, vol. 7, June2010, pp. 749-759.

[6] J. Li, A.W. Fu, H. He, J. Chen, H. Jin, D. McAullay, G. Williams, R. Sparks, C. Kelman, "Mining Risk Patterns in Medical Data," in *Proceedings of KDD 2005*, pp. 770-775.

[7] N.V. Chawla, K.W. Dowyer, L. O. Hall, W. P. Kegelmeyer, , "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research*, vol. 16, 2002, pp. 321-357.

[8] H. Sug, D.D. Dankel II, "Selection of Artificial Data of Minority for Better Data Mining," in *Proc. of the 2nd International Conference on Systems, Control and Informatics*, 2014, pp. 191-194.

[9] J.R. Quinlan, *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, Inc., 1993.

[10] W.W. Cohen, "Fast Effective Rule Induction," in *Proc. 12th International Conf. on Machine Learning,* 1995, pp. 115–123.

[11] X. Wu, V. Kumar, J.R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G.J. McLachlan, A. Ng, B. Liu, P.S. Yu, Z. Zhou, M. Steinbach, D.J. Hand, D. Steinberg, "Top 10 algorithms in data mining," *Knowledge and Information System*, vol. 14, 2008, pp.1-37.

[12] A. Frank and A. Suncion, *UCI Machine Learning Repository* [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Sciences, 2010.

[13] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten, "The WEKA Data Mining Software: An Update*," SIGKDD Explorations,* vol. 11, issue 1, 2009.

[14] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, issue 1, 2001, pp. 5-32.

[15] N. Landwehr, M. Hall, E. Frank, "Logistic Model Trees," *Machine Learning*, vol. 95, no.1-2, 2005, pp.161-205.

Hyontai Sug received BS degree in computer science and statistics from Busan national university, Korea, in 1983, and MS degree in computer science from Hankuk university of foreign studies, Korea, in 1986, and Ph.D. degree in computer and information science and engineering from university of Florida, USA in 1998. He was a researcher of Agency for Defense Development, Korea from 1986 to 1992, and a full time lecturer of Pusan university of foreign studies, Korea from 1999 to 2001. Currently, he is a professor of Dongseo university, Korea since 2001. His research interests include data mining and database applications.