IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 23, 2021, accepted June 3, 2021, date of publication September 23, 2021, date of current version September 29, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3114443

An Over Sampling Method of Unbalanced Data Based on Ant Colony Clustering

GAO YANG AND LIU QICHENG
School of Computer and Control Engineering, Yantai University, Yantai 264005, China

Corresponding author: Liu Qicheng (ytliuqc@163.com)

This work was supported in part by the Natural Science Foundation of Shandong Province under Grant ZR2016FM42, in part by the Primary Research and Development Plan of Shandong Province under Grant 2016GGX109004, in part by the National Natural Science Foundation of China under Grant 61702439, and in part by the ''13th Five-Year Plan'' Key Demonstration Project for the Innovation and Development of the Marine Economy of National Oceanic Administration under Grant YHC-ZB-P201701.

ABSTRACT Aiming at the low classification accuracy of unbalanced data sets, an improved SMOTE over-sampling algorithm ACC-SMOTE (Ant Colony Clustering Synthetic Minority Oversampling Technology) based on ant colony clustering is proposed. On the one hand, the improved ant colony clustering algorithm is used to divide a small number of samples into different sub-clusters, fully considered the imbalance between inter-cluster and intra-cluster data, and SMOTE algorithm is used to oversample the samples according to the proportion of sub-clusters, to reduce the imbalance of intra-class data. On the other hand, Tomek Links data cleaning technology is used to correct the oversampled samples in time, the quality of synthetic samples is guaranteed by eliminating noise in data sets and overlapping samples generated by sampling methods. The training data set and the test data set used in this paper are both UCI data sets. The experimental results show that this algorithm can significantly improve the classification accuracy of a few classes, thus improving the classification performance of the classifier.

INDEX TERMS Unbalanced data, ant colony clustering, Tomek Links, oversampling.

I. INTRODUCTION

Unbalanced data set means that there is a large gap in the number of samples contained in each category in the sample set. For categories with a large number of samples, it is called the majority (counterexample) sample, otherwise, it is the minority (positive example) sample [1]. In an unbalanced data set, classes with less samples may contain more critical information. For example, the events of human suffering from tumor diseases in medical diagnosis belong to a few categories, but if the tumor diseases are misdiagnosed as no lesions, the opportunity of early treatment may be missed, resulting in irreparable consequences [2].

The reason for unbalanced classification is that the classification result of the classifier is not ideal because of using a common classification method to deal with unbalanced data set. For example, if there are 998 counterexamples and 2 positive examples in a data set, you only need to select a learning method to predict the training set data as counterexamples, so that the learner generated by the learning method can achieve 99.8% classification accuracy. Such a learner can not play a positive role in the research of unbalanced data sets, because it often cannot predict positive examples. With the advent of the era of big data, this kind of problem commonly exists in the fields of fault detection, credit card fraud detection, network intrusion recognition, and e-mail classification. Therefore, how to deal with the unbalanced data set quickly and accurately is a hot topic of current academic research [3]–[6].

The problem of unbalanced data set is usually solved from two aspects of algorithm and data. At the data level, there are two options: increasing the number of positive samples and reducing the number of negative samples. Both of them have something in common, which is to balance the data set to a certain extent by changing the number of samples in each category. Representative algorithms include the synthesis of a few oversampling techniques [7] (Synthetic Minority Oversampling Technique, SMOTE). The method used at the algorithm level is mainly to introduce the cost matrix or improve the classification error rate after obtaining the balanced data set. The new classifier, which combines the data level method and the algorithm level method, can be more multivariate and robust.

The associate editor coordinating the review of this manuscript and approving it for publication was Mehul S. Raval.

Traditional classifiers predict a small number of classes in the unbalanced data set with low accuracy. Therefore, for the two solutions mentioned above, most of the current researchers increase the number of positive samples in the data set as the starting point. The SMOTE algorithm used in reference [7] aims at oversampling all positive samples, which is easy to produce overlapped new composite positive samples. The safe level SMOTE [8] algorithm will detect the number of adjacent similar samples of each minority sample, and set the safety level of each minority sample. When the value is higher than a certain threshold, the sample will be oversampled. To some extent, this method avoids the generation of noise data, but the synthetic data are too concentrated in a few categories, which may lead to overfitting phenomenon. DB-SMOTE [9] algorithm takes the distance between the cluster centers of positive samples as the standard to oversample, and selects the positive samples at the edge as seeds to avoid overfitting, but this method is easy to produce overlapping samples. WOHC [10] in consideration of overlapping samples and overfitting problems in oversampling, the method of weighted oversampling is adopted to avoid overlapping samples, this method can only prevent but not eliminate overlapping samples.

Literature [11] proposes an over-sampling algorithm AGNES-SMOTES based on hierarchical clustering and an improved SMOTE algorithm. The algorithm follows the following steps: filter the noise samples in the data set; Agnes algorithm was used to cluster the majority of samples and the minority of samples. Divide a few sub-clusters according to the number of sub-clusters obtained. According to the sampling weight and the probability distribution of a few subgroups, the oversampled samples were selected. The generation of new samples is restricted to a certain range by using the centroid method. Although the algorithm improves the classification accuracy of a few categories of samples to some extent, it still synthesizes some noise samples. Literature [12] proposes the ACOR algorithm, which is a general preprocessing framework to improve the performance of existing over-sampling algorithms to deal with imbalance problems. The main advantage of the ant colony algorithm is that it can make full use of the existing over-sampling algorithm to obtain the ideal training set. Although the algorithm improves the classification accuracy of positive examples to a certain extent, it fails to improve the fitness function and pheromone updating strategy of ant colony clustering, resulting in poor robustness of the ACOR algorithm.

For the above problems, this paper proposes an improved smote oversampling algorithm ACC-SMOTE based on Ant Colony Clustering (Ant Colony Clustering Synthetic Minority Oversampling Technique). First, the positive samples are divided into different sub-clusters by the improved ant colony clustering algorithm, and then the samples are oversampled by the smote algorithm according to the proportion of each positive sample cluster. Finally, the positive samples after the above operations are corrected in time by using the Tomek links method to clean up the noise data generated in the positive sample set, to improve the quality of the positive samples after synthesis. The ACC-SMOTE algorithm proposed in this paper is compared with the classic SMOTE [7], Safe-level SMOTE [8], DB-SMOTE [9], WOHC [10] and the experimental results show that this method improves the prediction accuracy of positive samples.

II. PREPARATORY THEORY

A. SMOTE ALGORITHM

Chawla put forward the method of synthesizing positive samples in 2002, that is, SMOTE [7]. Firstly, a positive sample is randomly selected, and one sample is selected from the nearest K samples. Then, a new sample is synthesized according to the sampling proportion and formula (1). To achieve the relative balance of the number of positive and negative samples in the unbalanced data set, the formula of synthesizing minority sample points is used to synthesize minority samples.

$$Y = X_1 + \text{rand} * (X_1 - X_2) \tag{1}$$

Among them, X_1 is positive samples, X_2 is one of the K samples closest to X_1, rand is a function of randomly generating a number from 0 to 1, Y represents the latest generated positive samples.

In this paper, when using SMOTE oversampling algorithm, we need to use the KNN algorithm to find the k nearest neighbor of a positive sample [13].

The idea of KNN algorithm is to randomly select a sample from the data set and calculate the distance between the sample and all the remaining samples in the data set, and then select the nearest k samples. If most of the selected k samples are from the same category, it can be inferred that this sample also belongs to this category. When calculating the k-nearest neighbor of a sample, Euclidean distance is usually selected to calculate the distance between two samples as a quantitative distance measurement formula, as shown below.

$$d_{ij} = \sqrt{\sum_{i,j=1}^{n} (x_i - x_j)^2} \tag{2}$$

where n is the number of samples, i and j are the number of samples in the data set.

B. BASIC PRINCIPLE OF ANT COLONY ALGORITHM

The idea of the ant colony clustering algorithm is to make ants move randomly in an area with a lot of data [14]. When an ant moves randomly to one of the regions with data, it calculates the similarity of the data in its domain to get the probability that the ant bears it, if the probability is larger than the random number, the ant picks up the data and carries it to move randomly, otherwise, the ant does not pick up the data and continues to move randomly. When the ant carries the data to move randomly to the blank area without data, the similarity of the blank area is calculated to get the probability of discarding the data, if the probability value is greater than the random number, the data will be discarded, otherwise, the discarding fails and the data will continue to move to another blank area for judgment. After the above process, the ant colony repeatedly picks up, moves, and discards, when it reaches the end of a cycle, it will get a final clustering result.

C. TOMEK LINK DATA CLEANING TECHNOLOGY

Tomek links is a key technology for data set cleaning, which can be used to clean up noisy data and overlapping samples caused by oversampling [15]. The main idea is to treat Tomek links as a pair of data instances, what are very close (European distance) but belong to different categories. Only when one of the data instances is noisy data or both data instances are on the boundary of their respective categories, can the two data instances form a pair of Tomek links. For example, given a pair of data(x_i, x_j), x_i, x_j belong to different categories, that is, a few classes x_i, majority class x_j, $d(x_i, x_j)$ is the Euclidean distance between them. If there is no other data instance x, which makes $d(x_i, x) < d(x_i, x_j)$ or $d(x, x_j) < d(x_i, x_j)$, this pair of data instances is called a pair of Tomek links. The noise data can be cleaned by the above method, that is, the data set no longer contains any two nearest data instances belonging to different categories.

Deleting the Tomek links pair can reduce the sample overlap between different categories, but when it is used as undersampling, the number of undersampling cannot be controlled, and the number of public samples that can be eliminated is relatively limited, so it is best to use it in combination with other methods as a data cleaning method [16]. When searching for Tomek links pairs in the oversampled dataset, it is mainly to clean up the composite samples and overlapping samples at the boundary. In order not to lose the data of counterexample samples, generally only the composite samples of Tomek links pairs in the boundary area are removed.

III. IMPROVED OVERSAMPLING ALGORITHM ACC-SMOTE

To obtain more information represented by positive samples in the data set, this paper aims to improve the SMOTE oversampling algorithm to get the ACC-SMOTE algorithm, so that the number of samples in each category is relatively balanced. This algorithm not only focuses on the imbalance between different classes of data sets but also considers the difference of the number of samples in different sub-clusters and the influence of noise data on samples. Ant colony clustering is used to get the optimal solution, which can provide more accurate clustering results for the next stage to use SMOTE algorithm to oversampling synthetic samples, to get more ideal synthetic samples. To eliminate the overlapped samples generated by the new synthesized samples, the algorithm adopts the Tomek Links method. ACC-SMOTE algorithm is mainly divided into three parts: ant colony clustering stage, oversampling stage, and synthetic data processing stage.

A. ANT COLONY CLUSTERING STAGE

To avoid data being picked up and put down repeatedly, which leads to slower clustering speed, and also to avoid the accuracy of clustering results being reduced due to randomness, the ant colony clustering method adopted by the ACC-SMOTE algorithm corresponds to a clustering scheme for each ant, and only the information element weight is included in the transfer equation. The reason is that the cluster center of each cluster needs to move constantly, and the transfer expectation is not easy to measure, so there is no transfer expectation in the transfer equation. When selecting a point's category in the search solution space, the pheromone is regarded as the proximity between the current point and each category. The point with the highest pheromone is selected by probability p, and the 1-p probability is selected by the roulette method according to the pheromone distribution.

$$p_{ij}^{k}(s) = \begin{cases} DTh+(1-DTh)\times\dfrac{\sigma_{ij}(s)}{\sum\limits_{s}\sigma_{ij}(s)}, & if\ j=\arg(\max\limits_{s}\sigma_{is}(s)) \\ (1-DTh)\times\dfrac{\sigma_{ij}(s)}{\sum\limits_{s}\sigma_{ij}(s)}, & \text{else} \end{cases} \tag{3}$$

The value of 1-DTh in formula (3) indicates the strength of the random factors in the search path, the greater the value, the greater the probability of choosing the path, leading to local optimization. So the direct transfer threshold $DTh = 0.9$, $\sigma_{ij}(s)$ means the pheromone contained in edge (i, j), when time is s.

In terms of updating pheromones, pheromones of all point class pairs on the scheme are added according to the optimization degree ($1/MSE$), each ant gets a scheme every iteration; then refresh the pheromone concentration value on each point class pair. The specific formula is as follows, in which all schemes have been arranged in ascending order of MSE.

$$\sigma_{ij}(s+m) = (1-emitP)\times\sigma_{ij}(s)+\Delta\sigma_{ij}(s) \tag{4}$$

$$\Delta\sigma_{ij}(s) = \sum_{k=1}^{lowNum}\Delta\sigma_{ij}^{k}(s) \tag{5}$$

$$\Delta\sigma_{ij}^{k}(s) = \begin{cases} P\_coe\times MSE_k^{-1}, & if\ sol(k,s)==j \\ 0, & \text{else} \end{cases} \tag{6}$$

The function of positive feedback of information is considered in equation (4), when the $emitP$ is small, the information positive feedback is dominant, and the convergence speed of the algorithm is faster. To improve the convergence speed of the ant colony clustering algorithm, the pheromone emission rate can be reduced to $emitP = 0.1$, the optimal number of refreshing pheromones lowNum=3, pheromone coefficient P_coe=10^7, MSE is the value of element in the array named sol.

The flow steps of the above algorithm are shown below.

Step1: Each ant corresponds to a solution to complete a clustering.

TABLE 1. Search solution space algorithm design.

```
Algorithm1.1:Search solution space algorithm design
Input: data set U, number of sub-clusters $n$
Output: $MSE$ of goodness of current ant clustering scheme
$AMT$←size(U)
for $s$=1 to $n$ do
    for $i$=1 to $AMT$ do
        $\tau_{si} = \sqrt{(\mathrm{U}_s - \mathrm{U}_i)^2}$
        for $j$=1 to $AMT$ do
            if $j \in \arg(\max(\tau_{si}))$
                $p_{ij}^{s}$=$0.9 + 0.1 \times \frac{\tau_{ij}}{\sum_s \tau_{si}}$
            else $p_{ij}^{s} = 0.1 \times \frac{\tau_{ij}}{\sum_s \tau_{ij}}$
            $j$=max($p_{ij}^{s}$)
            $\mathrm{u}_s$←$\mathrm{U}_j$
            $AMTu_s$= size($\mathrm{u}_s$)
        end
    end
end
$MSE$=$\frac{1}{AMT} \times (\sum_{s=1}^{n} \sqrt{\sum_{i=1}^{AMTu_s} (\mathrm{U}_s - \mathrm{U}_i)^2})$
```

Step2: Initialize, obtain a clustering result, and let all ants start from the clustering scheme.

Step3: Regarding the pheromone as the distance between a sample point and each subcluster, search the solution space according to formula (3), and then calculate the goodness of the current ant clustering scheme MSE and store it in $sol(M, AMT+1)$, where M is the number of ants, AMT is the number of samples, and the last column is used to store the MSE value of each ant.

Step4: Rely on the formula (4)-(6) to refresh the pheromone, and rely on the superiority of the plan to increase the pheromone of all point pairs in the plan.

Step5: Repeat steps Step2, Step3, Step4 until Num is greater than MaxNum or the program repeats CRN consecutive times and Num is greater than MinNum, ending the loop.

For Step1 and Step2, use the nearest distance roulette method, select the more scattered K data samples from the given AMT positive sample samples as the initial sub-cluster centers, and give the initial clustering plan according to the nearest distance, and save it in the array named sol of each ant, then let all ants start from the clustering scheme. In the update process of Step 4, the pheromone of the $emitP$ ratio will be reduced.

Calculate the MSE of the scheme according to the scheme obtained after clustering by each ant and store it in the last column of the array named sol. Add pheromone to the lowNum scheme with smaller MSE, and the pheromone of the remaining clustering schemes will have a certain percentage of volatilization, the pseudo-code of the algorithm design is shown below.

TABLE 2. Design of updating pheromone algorithm.

```
Algorithm1.2 : Design of updating pheromone algorithm
Input: Data set U, the number of ants $m$, and the $sol$ array of each ant $MSE$
Output: updated pheromone $\tau_{ij}$
$AMT$←size(U)
for $k$=1 to $m$ do
    for $i$=1 to $AMT$ do
        for $j$=1 to $AMT$ do
            if sol($k$,$i$)==$j$
                $\Delta\tau_{ij} = 10$^7× $MSE_k^{-1}$
            else $\Delta\tau_{ij} = 0$
             $\tau_{ij}$ =0.9× $\tau_{ij} + \Delta\tau_{ij}$
             $\Delta\tau_{ij} = \sum_{k=1}^{3} \Delta\tau_{ij}$
        end
    end
end
```

B. OVERSAMPLING STAGE

After improving ant colony clustering above, we can get sub-clusters of different minority classes, the four formulas in section III.A are the core of the algorithm, which directly determines the quality of clustering results. Using SMOTE algorithm to over-sample, the process can be roughly divided into the following two parts. Firstly, a few classes are divided into different sub-clusters, and the sampling proportion P is determined according to the proportion of each sub-cluster size in the sample, as following formula (7); secondly, according to formula (1), a few classes of different sub-clusters are oversampling, and the above process is repeated until the oversampling proportion is reached. When using SMOTE algorithm to synthesize new positive samples, the KNN algorithm is used, which has been introduced in section II. A.

$$T = \frac{(N_{max} - N_{min}) * (N_{min} - C_i)}{N_{min}} \tag{7}$$

In the above formula, N_{max} and N_{min} respectively represent the number of samples of counter-examples and positive examples; C_i represents the number of samples in the current sub-cluster; T represents the sampling proportion. The pseudo-code for implementing SMOTE oversampling for a single positive sample cluster is shown in Algorithm 2.

C. SYNTHESIS DATA SORTING STAGE

In general, it is impossible to eliminate the noise samples generated in the synthesis of new minority class samples, and the existence of noise samples will degrade the performance of the classifier. In addition, for a few classes, the small number of samples leads to the weak anti-noise ability of the class itself, so the noise data has a greater impact on the classification effect. A few samples are synthesized by oversampling so that the number of samples in the sample set is relatively balanced. If the neighbor samples of the seed

TABLE 3. Smote algorithm design.

```
Algorithm2: SMOTE algorithm design
Input: positive sample cluster mindata, negative sample number $m1$, positive sample number $m2$
Output: the new positive sample cluster new_mindata after the sample is synthesized
$AMT1$←size(mindata);
$T=(m2-m1)*(m1-AMT1)/m1$
for $i$=1 to $AMT1$ do
    bestclose[5]=KNN($\text{mindata}_i$,5)
    Y [$k$]=$\text{mindata}_i$+rand($\text{mindata}_i$-bestclose[$k$])
    new_mindata←Y[$k$]
end
$AMT2$←size(new_mindata);
While($AMT2$!=$T$)
    new=∅
    if $AMT2>T$
        new←randchose(new_mindata, $AMT2-T$);
        new_mindata←new_mindata-new;
    else new←randchose(mindata,($T$- $AMT2$)/5);
    for $i$=1 to ($T$- $AMT2$)/5 do
        bestclose[5]=KNN($\text{new}_i$,5)
        Y [$k$]= $\text{new}_i$+rand($\text{new}_i$ - bestclose[$k$])
        new_mindata←Y[$k$]
    end
end
```

TABLE 4. Tomek links algorithm design.

```
Algorithm3: TOMEK LINKS algorithm design
Input: data set new_mindata after oversampling
Output: Cleardata
$h$←size(new_mindata)
for $i$=1 to $h$ do
    for $j$=1 to $h$ do
        $d_{ij} = \sqrt{(\text{new\_mindata}_i - \text{new\_mindata}_j)^2}$
        j←min($d_{ij}$)
        if label($\text{new\_mindata}_i$)!=label($\text{new\_mindata}_j$)
            Cleardata←new_mindata-$\text{new\_mindata}_i$
            Cleardata←new_mindata-$\text{new\_mindata}_j$
        end
    end
end
```

samples are most of the class samples when oversampling, the synthesized samples will be noise data, which will not provide useful information when using the classifier for classification, on the contrary, it may mislead the classifier to make wrong judgment, and then reduce its classification accuracy. Therefore, the ACC-SMOTE algorithm uses Tomek Links data cleaning technology to process the newly synthesized samples, the realization idea is to find the distance d(x_i, x_j) from each sample in the given data set to the rest of the samples, and then determine whether the two samples belong to the same category, if so, it is a pair of Tomek Links, and delete the two samples. Its theory has been introduced in section II.C, here is the pseudo-code of the implementation of the Tomek Links algorithm.

Its effect is shown in figure 1. As shown in figure 1, (a) is the initial data set, the black five-pointed stars represents the positive sample, and the circle represents the counter sample. (b) is the data set after oversampling. The dotted box in (c) marks the Tomek links pair. (d) shows the data set after clearing the Tomek links pair.

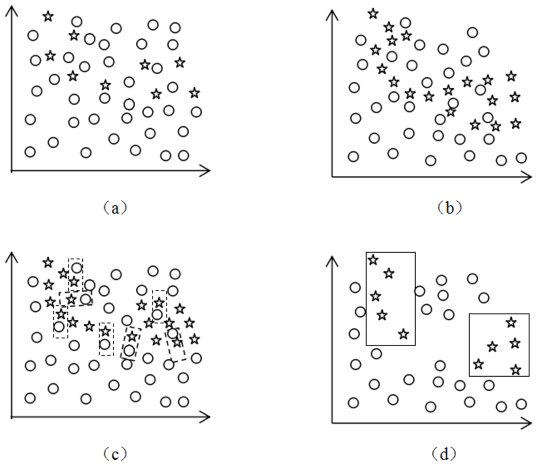

FIGURE 1. Effect picture of acc-smote algorithm.

IV. ANALYSIS OF EXPERIMENTAL RESULTS

In the experiment, we will divide the UCI data sets into test sets and training sets according to the principle of two eight points [17]. Using the above method to verify the effectiveness and feasibility of the algorithm. Due to the contingency of data set partition and new sample synthesis in the experiment, the following results are the average values obtained many times.

TABLE 5. Confusion matrix.

	Predictive positive class	Predictive negative class
Actual positive class	*TP*	*FN*
Actual negative class	*FP*	*TN*

A. VALUATING INDICATOR

For unbalanced data sets, the traditional accuracy of classification results and evaluation indicators are not applicable. Therefore, new evaluation indicators G-means [18] and F-value [19] are obtained from the confusion matrix shown in Table 5 below.

According to Table 5 above, the following evaluation indexes can be obtained:

Recall rate of positive samples:

$$rr_{TP} = \frac{TP}{TP + FN} \times 100\% \tag{8}$$

TABLE 6. Five sets of UCI data sets.

UCI data set	Minority/ Majority	Majority	Minority	Imbalance ratio
Blood	1/0	570	178	3.20
Breast Cancer	Yes/No	196	81	2.42
Ecoli	pp/other	268	52	5.15
Haberman	2/1	225	81	2.78
Ionoshpere	b/g	225	126	1.78
Pima	1/0	500	268	1.86
Yeast	ERL,POX, VAC,MEL/ other	1337	147	9.10

Recall rate of negative samples:

$$rr_{TN} = \frac{TN}{FP + TN} \times 100\% \tag{9}$$

Positive sample precision:

$$pr_{TP} = \frac{TP}{FP + TN} \times 100\% \tag{10}$$

G-means:

$$\text{G-means} = \sqrt{rr_{\mathrm{TN}} \times rr_{\mathrm{TP}}} \tag{11}$$

F-value:

$$\text{F-Value} = \frac{2 \times rr_{TP} \times pr_{TP}}{rr_{TP} + pr_{TP}} \tag{12}$$

The G-means value comprehensively considers the recall rate of positive samples and negative samples, only when both values are large, the G-means value will be large, which can more accurately show the classification effect of the model. The F-value focuses on the precision and recall rate of positive samples, which can more accurately show the classification accuracy of positive samples. In this paper, G-means value and F-value are used as evaluation indexes.

B. EXPERIMENTAL DATA

Before the experiment, some data need to be preprocessed. On the one hand, we delete the data with missing attributes. On the other hand, we need to turn multi -classification into two classification. The data set structure is shown in Table 6 below.

C. EXPERIMENTAL DATA COMPARISON OF EXPERIMENTAL RESULTS

To prove the validity and feasibility of the ACC-SMOTE algorithm, the classification results of C4.5 without any over-sampling algorithm are taken as the comparison criteria. The reason why C4.5 Classification Algorithm Combined with the ACC-SMOTE algorithm is that ACC-SMOTE is an over-sampling algorithm for the unbalanced data set, and it does not have a classification function, it needs a classification algorithm to verify its feasibility. Secondly, the oversampling algorithm is mostly verified by C4.5, at the same time, it is convenient for ACC-SMOTE algorithm to compare with other algorithms.

TABLE 7. Comparisons between the ACC-SMOTE algorithm and other algorithms.

Data set	algorithm	G-means/%	F-value/%
Blood	C4.5	56.17	38.89
	SMOTE+C4.5	59.18	41.27
	Safe-level SMOTE+C4.5	55.31	40.11
	DB-SMOTE+C4.5	57.96	39.68
	WOHC+C4.5	58.27	40.75
	ACC-SMOTE+C4.5	**60.23**	**43.87**
Breast cancer	C4.5	76.27	70.76
	SMOTE+C4.5	75.20	70.63
	Safe-level SMOTE+C4.5	75.40	69.68
	DB-SMOTE+C4.5	74.80	68.54
	WOHC+C4.5	76.50	**71.96**
	ACC-SMOTE+C4.5	**80.35**	70.83
Ecoli	C4.5	90.77	72.99
	SMOTE+C4.5	90.18	72.46
	Safe-level SMOTE+C4.5	**91.17**	73.52
	DB-SMOTE+C4.5	90.11	71.39
	WOHC+C4.5	90.18	73.52
	ACC-SMOTE+C4.5	85.85	**76.63**
Haberman	C4.5	48.60	32.28
	SMOTE+C4.5	54.36	37.76
	Safe-level SMOTE+C4.5	53.56	36.94
	DB-SMOTE+C4.5	54.81	38.63
	WOHC+C4.5	56.82	40.71
	ACC-SMOTE+C4.5	**59.54**	**43.32**
Ionoshpere	C4.5	78.35	73.53
	SMOTE+C4.5	79.63	74.14
	Safe-level SMOTE+C4.5	81.96	76.99
	DB-SMOTE+C4.5	82.30	77.56
	WOHC+C4.5	80.86	75.62
	ACC-SMOTE+C4.5	**86.25**	**82.56**
Pima	C4.5	66.05	59.96
	SMOTE+C4.5	65.19	55.86
	Safe-level SMOTE+C4.5	65.84	56.95
	DB-SMOTE+C4.5	68.52	58.88
	WOHC+C4.5	68.55	60.01
	ACC-SMOTE+C4.5	**68.74**	**60.12**
Yeast	C4.5	54.42	37.58
	SMOTE+C4.5	60.84	43.40
	Safe-level SMOTE+C4.5	64.29	47.52
	DB-SMOTE+C4.5	63.51	45.23
	WOHC+C4.5	67.57	48.38
	ACC-SMOTE+C4.5	**70.33**	**48.77**

SMOTE+C4.5, Safe-level SMOTE+C4.5, DB-SMOTE+C4.5, WOHC+C4.5, four algorithms are compared with ACC-SMOTE+C4.5 in seven different datasets (Blood, Breast cancer, Ecoli, Haberman, Ionosphere, Pima, Yeast), G-means and F-value are used as evaluation indexes.

The experimental results are shown in the following Table 7.

From Table 7 above, it can be concluded that in six different datasets (Blood, Breast cancer, Haberman, Ionosphere, Pima, and Yeast), the G-means (60.23, 80.35, 59.54, 86.25, 68.74, 70.33) of ACC-SMOTE algorithm are higher than other algorithms, and in six different datasets (Blood, Ecoli, Haberman, Ionosphere, Pima, and Yeast), the F-value (43.87, 76.63, 43.32, 82.56, 60.12, 48.77) of ACC-SMOTE algorithm are higher than other algorithms. Compared with the WOHC algorithm, the G-means and F-value of the algorithm are improved, on the one hand, the algorithm can effectively overcome the impact of noise samples; on the other hand, the algorithm can better handled the unbalanced data of samples with classification difficulties, effectively define the boundary of positive samples, and then improve the accuracy of classification.

V. CONCLUSION

For the problem of unbalanced data set, most of the existing methods only focus on the imbalance between different classes of data sets, without considering the difference of the number of samples in different sub-clusters of the same kind, after over-sampling, the training set may be marginalized, and the trained model is more likely to be overfitted, thus reducing the classification accuracy. To overcome the shortcomings of the existing oversampling algorithm, the ACC-SMOTE algorithm is proposed. Aiming at the problems existing in the process of synthesizing new samples, this algorithm introduces clustering ideas and data cleaning technology, which could effectively reduce the probability of imbalance and marginalization problems between and within clusters in the data set. It is verified that the ACC-SMOTE algorithm can improve the classification accuracy of a small number of samples, and the classification performance is better, which proves the feasibility and effectiveness of the algorithm.

In real life, there will be multiple classifications. In future work, we can further study the sampling method of multi-classification data set, to apply the ACC-SMOTE algorithm to more fields.

REFERENCES

[1] Z. Mo, Y. R. Gai, and G. L. Fan, "Credit card fraud classification based on GAN-AdaBoost-DT imbalance classification algorithm," *J. Comput. Appl.*, vol. 39, no. 2, pp. 618–622, 2019.

[2] M. A. Mazurowski, P. A. Habas, J. M. Zurada, J. Y. Lo, J. A. Baker, and G. D. Tourassi, "Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance," *Neural Netw.*, vol. 21, nos. 2–3, pp. 427–436, 2008.

[3] Z. Yang, W. H. Tang, A. Shintemirov, and Q. H. Wu, "Association rule mining-based dissolved gas analysis for fault diagnosis of power transformers," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 39, no. 6, pp. 597–610, Nov. 2009.

[4] J. Pun and Y. Lawryshyn, "Improving credit card fraud detection using a meta-classification strategy," *Int. J. Comput. Appl.*, vol. 56, no. 10, pp. 41–46, Oct. 2012.

[5] S. Kang, C. Liu, X. Fan, H. Li, and N. Yang, "Research on P2P botnets detection based on the ENN-ADASYN-SVM classification algorithm," *J. Chin. Comput. Syst.*, vol. 37, no. 2, pp. 216–220, 2016.

[6] P. Bermejo, J. A. Gamez, and J. M. Puerta, "Improving the performance of Naive Bayes multinomial in e-mail foldering by introducing distribution-based balance of datasets," *Expert Syst. Appl.*, vol. 38, no. 3, pp. 2072–2080, 2011.

[7] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, no. 1, pp. 321–357, 2002.

[8] C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, "Safe-Level-SMOTE: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem," in *Proc. Pacific-Asia Conf. Adv. Knowl. Discovery Data Mining*. Berlin, Germany: Springer-Verlag, 2009, pp. 475–482.

[9] Y. X. Liu, S. Liu, and T. Liu, "A new oversampling algorithm DB-SMOTE," *Comput. Eng. Appl.*, vol. 50, no. 6, pp. 92–95, 2014.

[10] Y. Xia, L. Li, Z. Xu, and H. Bae, "Weighted oversampling method based on Hierarchical clustering for unbalanced data," *Comput. Sci.*, vol. 46, no. 4, pp. 22–27, 2019.

[11] X. Wang, Y. Yang, M. Chen, Q. Wang, Q. Qin, H. Jiang, and H. Wang, "AGNES-SMOTE: An oversampling algorithm based on hierarchical clustering and improved SMOTE," *Sci. Program.*, vol. 2020, pp. 1–9, Sep. 2020.

[12] M. Li, A. Xiong, L. Wang, S. Deng, and J. Ye, "ACO resampling: Enhancing the performance of oversampling methods for class imbalance classification," *Knowl.-Based Syst.*, vol. 196, May 2020, Art. no. 105818.

[13] T. Cover, "Estimation by the nearest neighbor rule," *IEEE Trans. Inf. Theory*, vol. IT-14, no. 1, pp. 50–55, Jan. 1966.

[14] E. Lumer and B. Faieta, "Diversity and adaptation in population of clustering ants," in *Proc 3rd Int. Conf. Simulation Adapt. Behav. From Animal Animals*. Cambridge MA, USA: MIT Press, 1994, pp. 499–508.

[15] D. Debashree, B. Saroj kr, and P. Biswajit, "Redundancy-driven modified Tomek-link based undersampling: A solution to class imbalance," *Pattern Recognit. Lett.*, vol. 2017, pp. 1339–1351.

[16] R. M. Pereira, Y. M. G. Costa, and C. N. Silla, Jr., "MLTL: A multi-label approach for the tomek link undersampling algorithm," *Neurocomputing*, vol. 383, pp. 95–105, Mar. 2020.

[17] *UCI Machine Learning Repository*. Accessed: Jun. 10, 2017. [Online]. Available: http://archive.ics.uci.edu/ml/datasets

[18] C.-T. Su, L.-S. Chen, and Y. Yih, "Knowledge acquisition through information granulation for imbalanced data," *Expert Syst. Appl.*, vol. 31, no. 3, pp. 531–541, Oct. 2006.

[19] H. Hui, W. Wen-yuan, and M. Bing-huan, "Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning," in *Proc. Int. Conf. Intell. Comput.* Hefei, China: Springer, 2005, pp. 878–887.

GAO YANG is currently pursuing the M.S. degree with the School of Computer and Control Engineering, Yantai University.

His research interests include recommended systems and parallel computing.

LIU QICHENG received the Ph.D. degree in engineering from China University of Petroleum, Beijing. He was a Postdoctoral Researcher with the Department of Computer Science, Tsinghua University. He is currently a Professor with the School of Computer and Control Engineering, Yantai University. His research interests include cloud computing, big data, and data mining.