ISSN 2616-7433 Vol. 2, Issue 8: 05-12, DOI: 10.25236/FSST.2020.020802

Research on Classification of Imbalanced Data Set Based on TMDSMOTE Algorithm

Wei Sun[1], Chen Cheng[1,*], Gaiqing Yu[1]

1 Information Engineering College, Shanghai Maritime University, Shanghai 201306, China
**Corresponding Author*

ABSTRACT. *Scholars represented by Chawla proposed the SMOTE algorithm with the core idea of random upsampling. By constructing positive samples artificially, the number of negative samples and positive samples in the data set tended to be balanced. For SMOTE algorithm, scholars have proposed many improved algorithms. Considering the above problems, this paper proposes an improved algorithm TMDSMOTE algorithm, which not only considers the problem of sample distribution marginalization, but also considers the complexity of the algorithm.*

KEYWORDS: *TMDSMOTE Algorithm, Research on Classification*

1. Introduction

In real life, imbalanced datasets exist widely, such as cancer diagnosis datasets, network intrusion datasets, etc. In these datasets, the identification of a small number of samples is often the focus of classification and has more reference value. In cancer diagnosis, if a cancer patient is misdiagnosed as normal, it may cause life threatening [17]. Traditional classification methods have some disadvantages when dealing with imbalanced datasets, and the classification effect is not good [10].

Scholars represented by Chawla proposed the SMOTE algorithm with the core idea of random upsampling. By constructing positive samples artificially, the number of negative samples and positive samples in the data set tended to be balanced [7]. For SMOTE algorithm, scholars have proposed many improved algorithms. For example, Scholars represented by Wang Chaoxue[19] proposed an improved SMOTE algorithm, which improved the shortcomings of the SMOTE algorithm and used a roulette algorithm to select the minority samples in the minority samples[9]. The article [1] cannot control the positive sample generation area and the number of samples, and the sample distribution is easily marginalized[8]. However, these methods have problems such as the easy marginalization of sample distribution, the complexity of algorithm calculation. Considering the above problems, this paper proposes an improved algorithm TMDSMOTE algorithm, which not only considers the problem of sample distribution marginalization, but also considers the complexity of the algorithm [4].

2. Traditional algorithms and principles

2.1 Smote algorithm

SMOTE is an improved scheme based on the random oversampling algorithm[3]. But it easily leads to the problem of algorithm overfitting[11]. The basic idea of the SMOTE is to artificially synthesize new samples based on the minority samples and add them to the data set, that is, first group the positive samples according to the typical distance calculation formula which also known as Euclidean distance[2]. Suppose a data set sample $X = \{x_1, x_2, x_3, ..., x_n\}$, $x_1, x_2, x_3, \ldots, x_n$ is the dimension of sample X, $Y = \{y_1, y_2, y_3, ..., y_n\}$, y1,y2,y3,…,yn is the dimension of sample Y. Then the Euclidean distance d between sample X and sample Y is:

$$d = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{1}$$

The six samples closest to Euclidean were grouped. According to the idea of clustering, positively close samples are also positively close[5]. The SMOTE constructs a new positive sample

ISSN 2616-7433 Vol. 2, Issue 8: 05-12, DOI: 10.25236/FSST.2020.020802

X_{new} randomly and randomly on the line connecting the two samples in each group of 6 samples

$$X_{new} = X + rand(0,1) \times (Y_i - X) \qquad i = 1,2,\cdots,6 \tag{2}$$

Where X is positive class sample, Y_i is the i-th nearest neighbor sample of X, and *rand (0,1)* represents a random number between 0 and 1[12]. Perform multiple iterations according to formula (2) to make the positive and negative data sets balanced.

3. Improvement of classification algorithm for imbalanced data sets

Tmdsmote algorithm. SMOTE has two obvious shortcomings. One is that it cannot solve the problem of marginalization of the positive sample distribution of the data set, and the other is that the calculation complexity is large. In the Article[11], Zhao Qinghua and others proposed two algorithms, TSMOTE and MDSMOTE. But they can only solve the problematic aspect of the SMOTE algorithm. This article proposes the TMDSMOTE (TriangleMaxDistance SMOTE) algorithm for the above problems. Compared with MDSMOTE and TSMOTE, TMDSMOTE has improved the effect[6]. TMDSMOTE only focuses on the 4 points of the centroid point of the positive sample, the farthest point, the second farthest point, and the third farthest point from the centroid point of the positive sample.

$$X_{new} = X_c + random(m,n) \times (X_{\max} - X_c) \tag{3}$$

X_{new} is the new sample point, X_c is the centroid point of all positive samples, $0 \leqslant m < 1$, $0 \leqslant n < 1$ in *random (m, n)*, $X_{\max} = \{X_{f\max}, X_{s\max}, X_{t\max}\}$ indicates One of the points in the farthest point X_{fmax}, the second farthest point X_{smax} and the third farthest point X_{tmax}.

This algorithm not only overcomes the problem of marginalization of the new sample distribution of SMOTE, but also it only needs to iterate once, and the algorithm is simple and efficient to implement. Generate a batch of new sample points according to formula (3) to directly balance the entire data set.

The detailed steps of the TMDSMOTE are as follows:

Input: In the original sample data $T = \{(x_1, y_1), (x_2, y_2), ..., (x_t, y_t)\} \in (R^n \times Y)^t$,set the minority group as positive class $X_{positive}$, and the majority class as negative class $X_{negative}$. The number of samples is $nP_{positive}$ and $nN_{negative}$ respectively[14].
STEP1:Calculate the centroid of positive samples $X_c = (\frac{1}{n}\sum_{i=1}^{n} X_{i1}, \frac{1}{n}\sum_{i=1}^{n} X_{i2}, \frac{1}{n}\sum_{i=1}^{n} X_{i3}, \cdots, \frac{1}{n}\sum_{i=1}^{n} X_{in})$, where n is the number of positive samples. Traverse all positive samples to find the three sample points with the largest distance from the center of mass. The largest sample point X_{fmax}, the second largest sample point X_{smax} and the third largest sample point X_{tmax}. The distance here is calculated by the Euclidean distance formula: $d(x, y) = \sqrt{(x_1, y_1)^2 + (x_2, y_2)^2 + \cdots + (x_n, y_n)^2} = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$
STEP2:Three samples Xfmax,Xsmax and Xtmax form a triangle, and the sample itself is the vertex of the triangle. A positive sample PTMD is randomly generated on the line between a randomly selected point and the center of mass.
STEP3: The standard SMOTE is used to synthesize the minority sample Xnegative, and the new set of samples is recorded as PS.
STEP4: Let $nN = P_{TMD} + P_s$, repeat Step2 until $nN = nN_{negative}$. nN is the sample set of the minority class obtained by the algorithm.

4. Experimental results and analysis

In the experiment, *F1*, *F-value*, and *G-mean* are commonly used to evaluate the merits of the classification algorithm in imbalanced data sets(the random forest classification is used here). These three indicators are based on the confusion matrix expanded, the definition of the confusion matrix[18] is shown in *Table 1*, and *Table 2* gives the calculation formulas of the three indicators[16].

ISSN 2616-7433 Vol. 2, Issue 8: 05-12, DOI: 10.25236/FSST.2020.020802

Table 1 Two-class confusion matrix

Model classification	Forecast category is positive	Forecast category is negative
Actual category is positive	TP	FN
Actual category is negative	FP	TN

Table 2 The calculation formulas for evaluation criteria

Performance evaluation index	Formula
F1	$F1 = \frac{2 \times \Pr ecison \times \operatorname{Re} call}{\Pr ecison \times \operatorname{Re} call} \frac{2}{\frac{1}{Precision} + \frac{1}{\operatorname{Re} call}}$
$F - value$	$F_{value} \frac{(1+\beta^2) \times \Pr ecison \times \operatorname{Re} call}{\beta^2 \times \Pr ecison + \operatorname{Re} call}$
$G - mean$	$G_{mean} = \sqrt{\frac{TP}{TP+FN} \times \frac{TN}{FP+TN}}$

Precision and recall are defined[15] as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}} \tag{5}$$

Among them, F-value comprehensively evaluates the recall and precision[15]. The parameterβ is the harmonic average of precision and recall [16]. G-mean comprehensively study on the classification accuracy of positive and negative prediction.

4.1 Experimental environment settings

In order to test the performance of the improved algorithm in this paper, a reasonable range selection is made for m and n in the generated samples. Therefore, in this experiment, 8 sets of imbalanced data sets shown in Table 3 are used as the test set, and the Python programming environment are used to simulate the improved algorithm. See the table below for details.

Table 3 8 imbalanced data sets in detail

Data set	Total sample size	Number of attributes	Number of positive sample	Number of negative sample	Imbalance ratio
Glass	2308	19	329	1979	1:6.02
Heart disease	218	13	80	138	1:1.725
Pima Indian Diabetes	768	8	268	500	1:1.87
Thyroid	215	5	35	180	1:5.14
Yeast	459	7	30	429	1:14.3
Poker	244	10	8	236	1:29.5
Credit card fraud	284807	30	492	284315	1:577.876
General diabetes (Tianchi)	6642	40	545	6097	1:11.187

Each experiment randomly divides 70% into the training set and 30% into the test set; the seed used to generate the random number generator in the random forest is set to 38; SMOTE needs to be grouped, and the sample of each group is set to 6; TMDSMOTE does not Need to group, construct a new sample according to formula (2); set the number of random forest decision trees to 10, and simulate the average of 1,000 times to obtain *F1*, *F-value* and *G-mean*.

4.2 Analysis of experimental results

(1) Analysis of *g-mean*, *f1* and *f-value* evaluation indicators

ISSN 2616-7433 Vol. 2, Issue 8: 05-12, DOI: 10.25236/FSST.2020.020802

Table 4 compares the *G-mean* index of the algorithms on 8 different data sets. The experimental results show that in the G-mean dimension, TMDSMOTE is better than the traditional SMOTE, MDSMOTE, and TSMOTE.

Compared with SMOTE, the TMDSMOTE shows improvement in six data sets: glass, heart disease, Pima Indian diabetes, thyroid, yeast, and general diabetes (Tianchi). Especially in the Pima Indians diabetes data set, the *G-mean* value increased from 0.569744727 to 0.69369561, and the effect is very obvious.

Compared with MDSMOTE, the TMDSMOTE shows an improvement effect on the six data sets of heart disease, thyroid, yeast, playing cards, credit card fraud, and general diabetes (Tianchi).

Compared with TSMOTE, the TMDSMOTE has an improved effect on the four data sets of heart disease, playing cards, credit card fraud, and general diabetes (Tianchi). At the same time, the TMDSMOTE has lower time complexity and consumes less time[13].

Table 4 G-mean index results on different algorithms

Data set	G-mean			
	SMOTE	MDSMOTE	TSMOTE	TMDSMOTE
Glass	0.989013897	0.9931128997753704	0.9926538173788777	0.991693928
Heart disease	0.77151675	0.7854047569672647	0.7953206337036055	0.796235194
Pima Indian diabetes	0.569744727	0.6945731385939815	0.70041707822958	0.69369561
Thyroid	0.953462589	0.9941379211903968	0.9986038776773677	0.997119307
Yeast	0.469573817	0.5538250556031593	0.5585242930564195	0.556589846
Poker	0.6770032	0.45961940777125543	0.45961940777125543	0.487903679
Credit card fraud	0.918883585	0.8878918983363524	0.8888918883363514	0.88991283
General diabetes (Tianchi)	0.981268511	0.9843071963953085	0.982585267855749	0.984591421

Table 5 is a comparison of the *F1* index of the algorithms on 8 different data sets. The experimental results show that in the dimension of F1, TMDSMOTE is better than the traditional SMOTE, MDSMOTE, and TSMOTE.

Compared with SMOTE, the TMDSMOTE on 8 different data sets, except for the slightly smaller F1 index on the yeast dataset, the F1 index of TMDSMOTE in other data sets has been improved, indicating that the improved algorithm for positive and negative samples Forecast accuracy has improved. Among them, the playing card data set has increased from 0.222222222 to 0.460023.

Compared with MDSMOTE, TMDSMOTE for heart disease, thyroid, yeast, playing cards, credit card fraud, and general diabetes (Tianchi) all show improved results;

Compared with TSMOTE, the TMDSMOTE has improved effects on the three datasets of heart disease, playing cards, and ordinary diabetes (Tianchi). At the same time, TMDSMOTE has a lower time complexity and a lower time cost.

Table 5 F1 index results on different algorithms

Data set	F1			
	SMOTE	MDSMOTE	TSMOTE	TMDSMOTE
Glass	0.984771574	0.9928630000000012	0.9924000000000011	0.991476
Heart disease	0.731707317	0.7385240000000001	0.7502329999999996	0.751393
Pima Indian diabetes	0.467741935	0.614135	0.6206779999999998	0.612957
Thyroid	0.952380952	0.982789	0.9985719999999999	0.995012
Yeast	0.333333333	0.307094	0.315438	0.314407
Poker	0.222222222	0.43335499999999955	0.43335499999999955	0.460023
Credit card fraud	0.838926174	0.842173	0.84225	0.842184
General diabetes (Tianchi)	0.978328173	0.984095999999999	0.9824099999999987	0.98439

Table 6 is a comparison of the F-value index of the algorithms on 8 different data sets. The experimental results show that in the F-value dimension, TMDSMOTE is better than the traditional SMOTE, MDSMOTE and TSMOTE.

ISSN 2616-7433 Vol. 2, Issue 8: 05-12, DOI: 10.25236/FSST.2020.020802

Compared with SMOTE, TMDSMOTE on 8 different data sets, except for the slightly smaller F-value index on the credit card fraud dataset, the F-value index of TMDSMOTE in other data sets has been improved, indicating that the optimization algorithm has improved the prediction accuracy of positive and negative samples.Among them, the index value of pima Indian diabetes data increased from 0.395095368 to 0.58907, and the effect was most obvious, which was nearly doubled.

Compared with MDSMOTE, TMDSMOTE also shows an improvement in the six data sets of heart disease, thyroid, yeast, playing cards, credit card fraud, and general diabetes (Tianchi);

Compared with TSMOTE, TMDSMOTE has improved results on the four data sets of heart disease, playing cards, credit card fraud, and general diabetes (Tianchi). At the same time, TMDSMOTE has lower time complexity and less time cost.

Table 6 F-value index results on different algorithms

Data set	F-value			
	SMOTE	MDSMOTE	TSMOTE	TMDSMOTE
Glass	0.981781377	0.9889630000000009	0.9882300000000009	0.986706
Heart disease	0.663716814	0.7022010000000003	0.7176890000000001	0.718605
Pima Indian diabetes	0.395095368	0.5901850000000001	0.6015469999999999	0.58907
Thyroid	0.925925926	0.9896470000000005	0.9977769999999999	0.995232
Yeast	0.256410256	0.3213790000000001	0.32753000000000015	0.325735
Poker	0.333333333	0.3611399999999994	0.3611399999999994	0.383364
Credit card fraud	0.842318059	0.8101520000000001	0.8102520000000001	0.831081081
General diabetes (Tianchi)	0.969325153	0.9749359999999999	0.9722089999999993	0.975386

(2) Comparative analysis of time consumption

Table 7 and Figure 1 compare the time consumption of the algorithms on eight different data sets. The experimental results show that in terms of time consumption, in general, TMDSMOTE has less time cost, so the effect is better.

Table 7 Comparison results of time consumption on different algorithms

Data set	Time consumption comparison(s)	
	TSMOTE	TMDSMOTE
Glass	15.691968441009521	15.23019790649414
Heart disease	7.0741801261901855	6.797799110412598
Pima Indian diabetes	9.755429744720459	9.316598415374756
Thyroid	5.017945289611816	4.757171154022217
Yeast	5.973762273788452	6.695849418640137
Poker	6.956589698791504	5.315240383148193
Credit card fraud	3764.8539032936096	3627.69520974159
General diabetes (Tianchi)	38.7529354095459	36.73433184623718

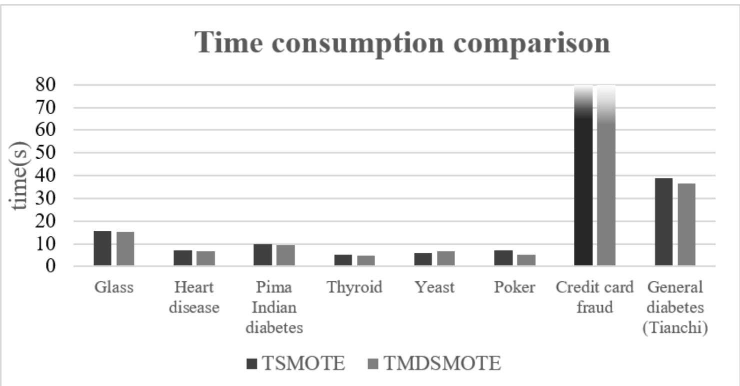

Fig.1 Time consumption comparison

ISSN 2616-7433 Vol. 2, Issue 8: 05-12, DOI: 10.25236/FSST.2020.020802

(3) Analysis of *random (m, n)* value range

The comparison of the three indexes F1, F-value, and G-mean of different random (m, n) values of TMDSMOTE is shown in Table 8, Table 9, and Table 10, which are the calculation results of 1000 simulations. Table 7 shows that on six different data sets, the F1 index works best when random (0.8, 1.0) is taken on the glass and heart disease data sets. When random (0.2,0.4) is selected on the Pima Indians diabetes data set, the F1 index works best. The F1 index works best when random (0.6,0.8) is taken on the thyroid and playing card data sets. On the diabetes (Tianchi) dataset, when random (0.0,1.0) is taken, the F1 index works best. Table 8 shows that on six different data sets, the F1 index works best when random (0.8, 1.0) is taken on the glass data set. When random (0.2, 0.4) is selected on the Pima Indians diabetes data set, the F1 index works best. The F1 index works best when random (0.6, 0.8) is taken on the thyroid and playing card data sets. On the data set of heart disease and diabetes (Tianchi), the F1 index works best when random (0.0, 1.0) is taken. Table 9 shows that on six different data sets, the F1 index works best when random (0.8, 1.0) is taken on the glass and heart disease data sets. When random (0.2, 0.4) is selected on the Pima Indians diabetes data set, the F1 index works best. The F1 index works best when random (0.6, 0.8) is taken on the thyroid and playing card data sets. On the diabetes (Tianchi) dataset, when random (0.0, 1.0) is taken, the F1 index works best.

Table 8 F1 index results of TMDSMOTE on different random (m, n)

Data set	F1					
	[0.0,0.2)	[0.2,0.4)	[0.4,0.6)	[0.6,0.8)	[0.8,1.0)	[0.0,1.0)
Glass	0.9925070000000012	0.9910120000000012	0.9915210000000012	0.9910200000000011	0.993266000000001	0.9926570000000009
Heart disease	0.732873	0.7285619999999998	0.7299230000000003	0.7374930000000002	0.744683	0.7436970000000003
Pima Indian diabetes	0.6087830000000001	0.620564	0.613917	0.6148060000000003	0.6182229999999997	0.6176099999999999
Thyroid	0.999089	0.999524	0.9986949999999999	0.999565	0.9823990000000004	0.9954450000000001
Yeast	0.3166819999999998	0.3800189999999966	0.4466889999999995	0.4666899999999995	0.3600179999999967	0.4666899999999995
Poker	0.982710999999999	0.9837929999999987	0.9809679999999987	0.9847929999999987	0.9839199999999987	0.985629999999999

Table 9 F-value index results of TMDSMOTE on different random (m, n)

Data set	F-value					
	[0.0,0.2)	[0.2,0.4)	[0.4,0.6)	[0.6,0.8)	[0.8,1.0)	[0.0,1.0)
Glass	0.988697000000001	0.9857320000000009	0.986541000000001	0.985920000000001	0.9894260000000008	0.9886370000000008
Heart disease	0.69434	0.6877259999999997	0.6911330000000001	0.6994890000000002	0.7102800000000004	0.7107519999999999
Pima Indian diabetes	0.5827600000000003	0.5981830000000002	0.5905930000000003	0.5940200000000003	0.5944489999999999	0.594354
Thyroid	0.99908	0.999259	0.9994630000000001	0.9998210000000001	0.9900520000000003	0.9954000000000001
Yeast	0.2655759999999997	0.31669199999999953	0.37225199999999936	0.3889199999999993	0.30002399999999957	0.3889199999999993
Poker	0.9726949999999996	0.9744139999999998	0.9700429999999998	0.9759889999999998	0.9747129999999993	0.9773129999999997

Table 10 G-mean index results of TMDSMOTE on different random (m, n)

Data set	G-mean					
	[0.0,0.2)	[0.2,0.4)	[0.4,0.6)	[0.6,0.8)	[0.8,1.0)	[0.0,1.0)
Glass	0.9929679548063066	0.9910657102692053	0.9915741024737528	0.9911947384475455	0.9933921432343294	0.9929087617274273
Heart disease	0.7804596884	0.7765786029	0.77809237998	0.78420239277	0.7905251622	0.79013437745

ISSN 2616-7433 Vol. 2, Issue 8: 05-12, DOI: 10.25236/FSST.2020.020802

	881063	957991	42364	44411	04922	05323
Pima Indian diabetes	0.6900796511999819	0.6998570321029561	0.6945508204936036	0.6955965034390351	0.6977995322646255	0.6973936720043823
Thyroid	0.9994416006147486	0.9995346258924559	0.9997209241668784	0.9999069747222927	0.9944259030536448	0.9972080030737434
Yeast	0.3393127023011055	0.4030508652763317	0.47376154339498633	0.49497474683058273	0.3818376618407353	0.49497474683058273
Poker	0.9828988579225463	0.9839750392504968	0.9812351209307707	0.9849629881327522	0.9841727443757082	0.9857938732780405

In summary, when random (m, n) takes m = 0.6, n = 0.8, or m = 0.8, n = 1.0, TMDSMOTE has more effective effects on three different indicators: F1, F-value, and G-mean. Therefore, when TMDSMOTE proposed in this paper is used to consider the random number random (0.6, 0.8) or random (0.8, 1.0) when generating a new sample set, the effect is generally the best.

5. Conclusion

The TMDSMOTE algorithm proposed in this paper is an optimization algorithm of MDSMOTE and TSMOTE, which improves the problems of sample distribution marginalization and high time complexity. At the same time, a more reasonable range value analysis is made for m,n in random (m, n) when generating samples. However, there are still many noise samples in the optimization algorithm in this paper. In future research, we will focus on the introduction of a typical correlation analysis (CCA) for initial sample screening and the secondary screening combined with the GAN idea to generate an effective evaluation of the reasonableness of the samples.

References

[1] Chawla N V, Bowyer K W, Hall L O (2002). SMOTE:synthetie minority over- sampling technique[J].Journal of Artificial Intelligence Researeh, no.16, pp.321-357.

[2] Guangyuan Deng (2019). Research and development of power transformer vibration monitoring and diagnosis algorithms and system software based on the Internet of Things [D]. Zhejiang University.

[3] Xu Jin, Lei Wang, Guozi Sun, et al (2019). An Undersampling Method for Unbalanced Data Based on Centroid Space [J]. Computer Science, vol.46, no.2, pp.50-55.

[4] Xinai Xu (2018). Recognition and separation algorithm for data overlap between classes of unbalanced fiber sensing data sets [J] .Laser Magazine, vol.39, no.11, pp.120-125.

[5] Xueyan Wen, Liying Zhao, Kesheng Xu, et al (2018). Application of Improved MDSMOTE and FC-SVM in Imbalanced Data Set Classification [J]. Journal of Harbin University of Science and Technology, vol.23, no. 4, pp.87-94.

[6] Pengfei Zhang, Yigui Wang, Zhijun Zhang (2019). Research on personalized recommendation algorithm integrating tags and multiple information [J]. Computer Engineering and Applications, vol.55, no.5, pp.159-165.

[7] Aiying Yin, Yunbing Wu, Xiaohua Yang (2018). Hybrid sampling algorithm for unbalanced data for manufacturing industry [J]. Computer Engineering and Design, vol.39, no.4, pp.1053-1058.

[8] Xueyan Wen, Jianan Chen, Weipeng Jing, et al (2018). Optimization research on classification model for imbalanced data sets [J]. Computer Engineering, vol.44, no.4, pp.268-273 + 293.

[9] Wei Yi, Li Mao, Jun Sun, Linhai Wu (2018). Research on classification of improved Smote algorithm on imbalanced data sets [J] .Computer and Modernization, no.3, pp.83-88.

[10] Guoquan Wang (2017). Research on feature selection algorithm for high-dimensional unbalanced data [D]. Harbin Institute of Technology.

[11] Qinghua Zhao, Yihao Zhang, Jianfen Ma, et al (2018). Research on Improved SMOTE Classification Algorithm for Non-balanced Data Sets [J]. Computer Engineering and Applications, vol.54, no.18, pp. 168-173.

[12] Yan Zhang (2017). Research on outlier detection for unbalanced data [D]. Qingdao University of Science and Technology.

[13] Yan Li, Yihua Li, Jinhuan Wang (2017). A new music personalized recommendation algorithm based on LDA-MURE model [J]. Journal of Jilin University (Science Edition), vol.55, no.2, pp.371-375.

[14] Yunyi Pei (2016). A Study on Affective Analysis of Chinese Travel Reviews [D]. Beijing Jiaotong University.

ISSN 2616-7433 Vol. 2, Issue 8: 05-12, DOI: 10.25236/FSST.2020.020802

[15] Huizhen Zhao, Fuxian Liu, Longyue Li (2016). Collaborative fuzzy C-means algorithm for K-nearest neighbor estimation coordination coefficients [J]. Computer Engineering and Applications, vol.52, no.19, pp.19-24 + 30.
[16] Ruolei Chen (2013). Research on Prediction Methods of Inherently Irregular Protein Structures Based on Multi-scale and Multi-feature [D]. Harbin Engineering University.
[17] Lina Liu, Zhilou Yu, Huaxiang Zhang (2011). Dimension reduction method for imbalanced data sets [J]. Information Technology and Informatization, no.5, pp.62-64.
[18] Shufeng Yang (2009). Application of classification technology in medical diagnosis [D]. Shantou University.
[19] Chaoxue Wang, Zhengmao Pan, Lili Dong, etc (2013). Research on classification of unbalanced data sets based on improved SMOTE [J]. Computer Engineering and Applications, vol.49, no.2, pp.184-187.