A Novel Density-Based Spatial Clustering of Application with Noise Method for Data Clustering

Yuchang Si[1]

Software College, Shenyang Normal University
Shenyang 110034, China
siyuchang@163.com

Abstract. The traditional methods are easy to generate a large number of fake samples or data loss when classifying unbalanced data. Therefore, this paper proposes a novel DBSCAN (density-based spatial clustering of application with noise) for data clustering. The density-based DBSCAN clustering decomposition algorithm is applied to most classes of unbalanced data sets, which reduces the advantage of most class samples without data loss. The algorithm uses different distance measurements for disordered and ordered classification data, and assigns corresponding weights with average entropy. The experimental results show that the new algorithm has better clustering effect than other advanced clustering algorithms on both artificial and real data sets.

Keywords: Data clustering, DBSCAN, Distance measurement.

1. Introduction

As a common data mining technology, cluster analysis has been widely used in many fields. For different types of data, scholars have proposed different clustering algorithms. But for any data, in order to obtain good clustering results, we must choose the appropriate clustering algorithm in advance. Traditional clustering algorithms are mostly proposed for quantitative data, such as the classic k-means algorithm, which has excellent performance, simple method and fast calculation speed [1-3]. However, the K-means algorithm is difficult to handle qualitative (categorized) data. Compared with K-means algorithm, the K-modes algorithm is improved in two aspects: first, the difference degree is used to calculate the distance of different attribute values under the same attribute; second, the class center is updated by mode. Although K-modes algorithm can cluster categorized data, it has some limitations [4-6].

With the rapid development of computer and communication technology, the Internet, industry and other fields produce a large amount of data, how to find and use the valuable data in these large amounts of data has become a hot spot of current research. Therefore, data mining has attracted great attention, and data imbalance is a difficult problem in data mining. This problem refers to the situation where the number of instances of a certain class in the data set is much less than the number of instances of other classes, where the number of instances is more called the majority class (negative class), the number of instances is less called the minority class (positive class), and the ratio of the number of majority classes to the number of minority classes is called the imbalance ratio. Traditional machine learning algorithms, such as SVM, decision tree [7], KNN [8], etc., all assume that the data set is classified under the condition of balance. However, in real life, unbalanced data exist in many fields, such as credit card fraud [9], protein classification [10], fault diagnosis [11] and cancer diagnosis [12], etc. If traditional machine learning classification algorithms are used, they tend to be negative in order to pursue higher accuracy when processing unbalanced data sets. However, in the binary classification, the misclassification costs of positive and negative classes are often different. For example, in credit card fraud, misjudging fraud events as normal events will lead to unpredictable losses. In cancer diagnosis, misdiagnosing cancer patients with relatively small incidence groups as normal people will lead to patients missing the best treatment opportunity, which may seriously lead to life threats. Therefore, it is significant to study efficient unbalanced data classification algorithms [13,14].

In recent years, many scholars at home and abroad have done a lot of research on unbalanced data classification. When dealing with unbalanced data classification, the improvement method mainly includes two levels: data preprocessing level and classification algorithm level. The data preprocessing level is also known as the data balance method, which changes the distribution of data samples or eliminates the imbalance by sampling unbalanced data. The representative methods include under-sampling, over-sampling and mixed sampling. At the classification algorithm level, existing algorithms are modified to improve the model's ability to recognize a few classes. Typical algorithms include cost sensitive method, single class learning method and integrated learning method [15,16].

Chawla et al. [17] put forward a SMOTE (synthetic minority over-sampling technique). Unlike the random over-sampling algorithm, SMOTE did not simply copy or rotate to increase the number of samples, but extended the number of samples by synthesis of new, non-repetitive small samples and interpolation processing, and this method did not cause data loss. Based on a new perspective of training multi-class integrated classification models, Pakrashi et al. [18] combined multiple independent multi-class classifiers with sensor fusion characteristics of Kalman filter to build a multi-class integrated classification algorithm. On the basis of SMOTE, Lv et al. [19] combined SMOTE with AdaBoost to put forward SMOTEBoost algorithm, which synthesized instances from a few classes to indirectly change the weights, and further improved the performance of the classification model. However, if the unbalance rate of the original data set was relatively high, when the over-sampling algorithm was used to obtain the optimal balanced data set (the unbalance rate is 1:1), more fake samples could be generated, resulting in a particularly large number of data set samples, thus slowing down the algorithm. Based on SMOTE-Boost algorithm, Popel et al. [20] proposed a hybrid algorithm based on random under-sampling and boosting, called RUSBoost. Random under-sampling randomly deleted most instances to form a balanced dataset. However, due to the limitation of random under-sampling, the obtained data set could lose some useful information in most classes. Rayhan et al. [21] proposed an algorithm based on clustering under-sampling combined with Adaboost, called CUSBoost. The algorithm divided the original data set into majority classes and minority classes, and used the k-means clustering algorithm to divide the majority classes into multiple clusters. Then the majority of classes were under-sampled by randomly selecting 50% of the instances, so that the useful information of most classes could be preserved as much as possible under the condition of under-sampling. Ahmed et al. [22] combined under-sampling and over-sampling algorithms to propose the RSYNBagging algorithm, which used random under-sampling algorithm in the process of odd iterations, ADASYN over-sampling algorithm in even iterations, and Bagging algorithm in ensemble learning to vote and get classification results. Elyan et al. [23] put forward CDSMOTE algorithm. In this algorithm, k-means clustering algorithm was used to divide most classes into multiple clusters, and then SMOTE algorithm was applied to a few class samples to balance the data set and avoid the loss of information. However, because the classifier was relatively simple, it did not achieve good results. Although the above methods improve the classification performance of unbalanced data to a certain extent, there are some problems such as data loss or excessive data volume.

To solve these problems, this paper proposes a classification algorithm for unbalanced data of random forest based on DBSCAN cluster decomposition and over-sampling. The algorithm divides the unbalanced data set into majority class and minority class. Firstly, the DBSCAN algorithm is applied to the majority class for clustering, the majority class is decomposed into multiple subclasses, and then the minority class data is processed using Borderline-SMOTE over-sampling algorithm. Finally, this method is combined with random forest. The experimental results show that this algorithm can effectively improve the classification effect of unbalanced data.

2. Related Works

DBSCAN [24] (density-based spatial clustering of applications with noise) is a density-based clustering algorithm. Its purpose is to discover clusters of arbitrary shapes, and the parameters describe the tightness of the sample distribution of the field, where the field radius of the density is defined, and the threshold of the core point is defined. When eps and $MinPts = 5$ are set, the algorithm principle is shown in Figure 1. The algorithm selects a core object (yellow dot in the figure), expands continuously to the area of density reachability, and derives the maximum density connected sample set from the density reachability relationship, which is a cluster.

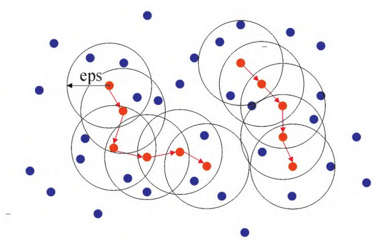

Fig. 1. Schematic diagram of DBSCAN algorithm

Borderline-SMOTE [25] is an improved SMOTE algorithm. The algorithm divides the minority samples into three categories according to the boundary, those with more than half the minority samples around are called Safe, those with less than half the minority samples are called Danger, and those with no minority samples around are called Noise. Since the Safe group may be misclassified as small, while the Danger group may be misclassified as large, the Borderline-SMOTE algorithm only samples the Danger group [26].

3. Random Forest Unbalanced Data Classification Algorithm Based on DBSCAN Cluster Decomposition and Over-sampling

In this paper, a new classification algorithm for unbalanced data of random forest based on the combination of density-based cluster decomposition technique and over-sampling technique is proposed. By applying DBSCAN algorithm to unbalanced data set, most classes in the data set are decomposed into clusters, and most classes are divided into multiple subclasses to reduce the advantages of most classes in the data set. Then over-sampling technique is used to increase the number of minority classes and improve the advantage of minority classes in the data set. For the unbalanced data set A, it is decomposed into data set A_c first, and then the average value of the class after the class decomposition is calculated. If the number of A few class data is still less than the average number, then the over-sampling algorithm is used to reevaluate the decomposed dataset A_c, and after using the over-sampling algorithm, a new dataset A_{co} is created, which is the result of the class decomposition and over-sampling. Finally, random forest classification technology is applied to data set A_{co}. the classification flow chart of unbalanced data of random forest based on DBSCAN cluster decomposition and over-sampling is shown in Figure 2.

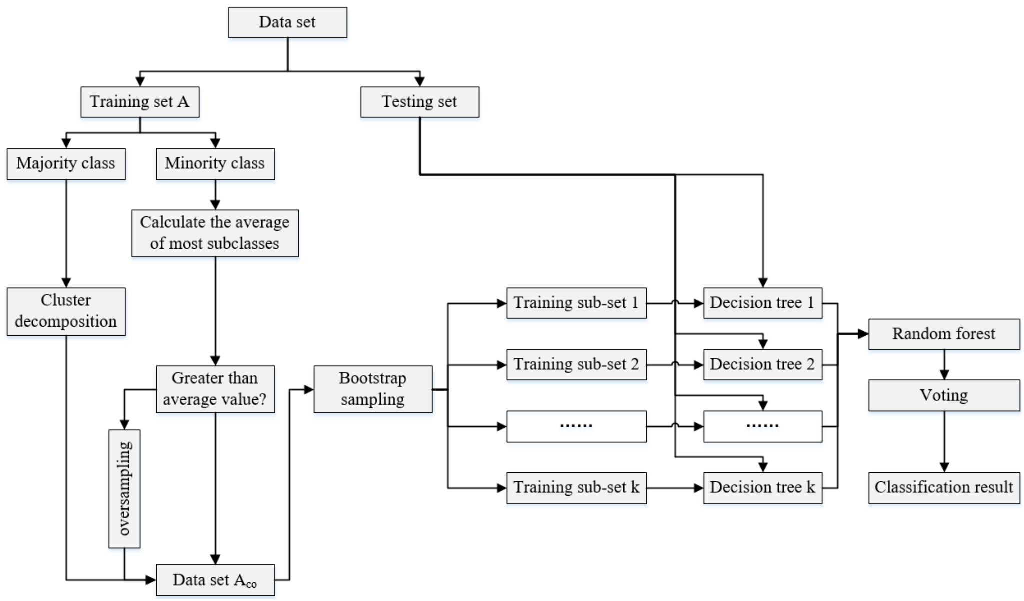

Fig. 2. Flow chart of random forest unbalanced data classification algorithm based on DBSCAN cluster decomposition and over-sampling

The balanced data set was obtained by DBSCAN clustering decomposition and over-sampling, and then the stochastic forest classification model was established. Set the scale of random forest classification model to t, then the specific steps are as follows.

1. Step l: Use the DBSCAN clustering algorithm for most classes $negative = (n_1, n_2, \cdots, n_m)$ in sample set A, and finally divide them into cluster $(c_1, c_2, \cdots, c_n)$.
2. Step2: Calculate clusters $(c_1, c_2, \cdots, c_n)$. If the number of samples of a few classes is larger than the average, go to Step5; If the number of samples in a few classes is less than the average value, then the subclass closest to the average is chosen as the criterion of over-sampling number, and the number of samples in a few classes is calculated as $pnum$, and the number of samples in a majority class as $nnum$.
3. Step3: Divide a few classes into Safe samples, Danger samples and Noise samples according to boundaries. Write the Danger sample as $(p'_1, p'_2, \cdots, p'_a)$, sample number as $dnum$. Calculate the k nearest neighbors of each sample p'_i in the Danger sample to a few classes of POSItives, and set a sampling ratio based on the

ratio of $nnum$ and $pnum$ to determine the sampling ratio N. According to the sampling rate N, s K-nearest neighbors are randomly selected for linear interpolation with sample p_i' to synthesize a few samples p_{new}:

$$p_{new} = p_i' + rand(0,1) \times d_j (j = 1, 2, \cdots, s). \tag{1}$$

Where d_j denotes the distance between p_i' and k nearest neighbors.

4. Step4: Add the synthetic minority samples to the original minority class to form a new minority class $positive-cl$.
5. Step5: Synthesize a new balanced dataset A_{co} by combining most class subsets and few class datasets.
6. Step6: Calculate the number of A_{co} samples N of the new balanced data set, use Bootstrap to randomly extract N times and generate a sub-training set with the same number of samples as the original training set. This process is repeated t times to obtain t sub-training sets and build t classification decision trees.
7. Step7: During splitting, the number of R features $(r < R)$ is randomly selected from the R features of the training sample as the split feature set of the current node, and the features with the smallest Gini coefficient [27,28] are selected for branching until no features are available or the Gini coefficient is optimized, and the decision tree stops splitting.
8. Step8: Use the generated t decision trees to form a random forest, and the final output results are decided by voting the classification results of each decision tree.

3.1. Cluster Decomposition

Cluster decomposition is to apply the clustering algorithm to the data set, divide the same type of data in the data set into a group and decompose it into multiple subsets. There are two main purposes, one is to reduce the advantages of most classes, and the other is not to generate any data information loss. The current popular method is to cluster the data set using k-means algorithm [29,30], but this method is susceptible to noise interference. To solve this problem, this paper uses the noise-insensitive DBSCAN clustering algorithm, and applies the algorithm to most classes of the dataset to generate multiple subclasses of most classes, thus reducing the impact of outliers on the model, and there is no need to set k values in advance like k-means algorithm.

The binary sorting task is as follows:

$$h(X) : X \to Y. \tag{2}$$

In $h(X)$, it maps each instance x_i to $y_i \in N, P$. After using cluster decomposition, a new classification task $h'(X)$ is obtained:

$$h'(X) : X \to Y'. \tag{3}$$

Mapping each instance x_i to $y_i' \in N_{c1}, N_{c2}, N_{c3}, \cdots, P$ to transform the data can cluster negative class N into multiple subclasses, effectively reducing the dominance of negative class N in the data set, which also means that the original binary classification problem is transformed into a multi-classification problem by cluster decomposition method.

3.2. Few Classes Over-sampling

Since the distribution tightness of samples is not the same, after applying DBSCAN clustering decomposition algorithm, new majority classes or minority classes may be generated from the original majority class instance. Therefore, it is necessary to calculate whether the number of samples of minority classes exceeds the average number of samples of subclasses of majority classes. If not, it is necessary to over-sample minority classes.

The most popular over-sampling algorithm at present is SMOTE algorithm proposed by Chawla. SMOTE algorithm uses K-nearest neighbor algorithm to generate synthetic instances through feature space instead of data space. Many scholars [31,32] have verified that this algorithm has achieved good results on unbalanced data. But the example generated by SMOTE algorithm may have problems such as sample overlap and noise. In order to avoid these problems, Borderline-SMOTE algorithm is used in this paper for over-sampling. It only over-samples the boundary samples of Danger group, which can avoid noise problems in a few classes. It also reduces the influence of the intra-class imbalance problem in a few class samples.

Borderline-SMOTE, like SMOTE, requires the selection of a majority class and a minority class as inputs, and the majority class's sample size is the reference for the number of minority class samples to be synthesized. In this paper, we choose most subclasses of the Borderline-SMOTE as the input to the Borderline-SMOTE, and most subclasses of the Borderline-SMOTE are $negative - cl$. The purpose of using the over-sampling method is to increase the number of minority samples, further reduce the imbalance ratio, reduce the advantage of the majority class, make the positive and negative classes tend to balance, and improve the classification accuracy of minority samples.

3.3. Random Forest

The principle of ensemble learning is to combine multiple weak classifiers to get a better result than a single weak classifier, and the most commonly used ensemble learning algorithms are divided into Boosting and Bagging. Random forest [33,34], a variant of Bagging algorithm, is an integrated algorithm containing multiple decision trees. Bootstrap technology is adopted when training samples, and N samples are randomly selected from the original data set as the training set of a tree. A part of the training sample features is randomly selected each time to build a decision tree. Pruning is not carried out in the training and growth process of each decision tree. Finally, the final result of the classifier is determined by voting. Due to the randomness of random forest, the risk of over-fitting is avoided and the classification accuracy is improved.

4. Experiments and Analysis

In this paper, the cluster analysis of the new algorithm is carried out on the artificial data set and the real data set respectively, and the results are compared with the traditional DBSCAN algorithm, MDBSCAN method [35] and the clustering results in this paper to further illustrate the effectiveness of the new DBSCAN algorithm. The clustering results will be compared from two perspectives of clustering accuracy and clustering distance. All algorithms in this paper are implemented by R language programming.

4.1. Dataset Specification

The sample points of the manual dataset are divided into three categories, and each sample contains two attributes, the first attribute is an unordered attribute, and the second attribute is an ordered attribute. The detailed values of the data set are shown in Table 1, where the sample point attribute 1 of category 1 takes random values in A, B, and attribute 2 takes random values in $1, 2, 3, 4$. The value of sample point attribute 1 of category 2 is C, and the value of attribute 2 is random in $4, 5, 6, 7$. Sample point attribute 1 of class 3 takes the value D, and attribute 2 takes a random value in $7, 8, 9$.

Table 1. Manual dataset description

Type	Attribute 1 value	Attribute 2 value	Sample number
1	A,B	1,2,3,4	100
2	C	4,5,6,7	100
3	D	7,8,9	100

Real data set Zoo, Balance scale, Hayes Roth are from UCI data sets (http://archive.ics.uci.edu/ml/). The attributes of these data sets are qualitative attributes, and there are differences in the attribute conditions among the three data sets. Some data sets contain both ordered attributes and unordered attributes, and some data sets contain only ordered attributes. The data set is described in detail in Table 2.

Table 2. Manual dataset description

Data set	Total attribute number	Number of ordered attributes	Sample class number
Zoo	16	1	7
Hayes Roth	4	2	3
Balance scale	4	4	3

4.2. Comparison of Clustering Accuracy

The clustering accuracy rate is used as the evaluation index of the experimental effect. The data set of the known experiment contains the category of each sample. During the experiment, the data will be gathered into category k, the value of which is equal to the real number of categories. The result after clustering will be compared with the known categories before clustering. The accuracy rate of clustering is calculated according to the following formula:

$$ACC = \frac{\sum_{i=1}^{k} a_i}{n}. \tag{4}$$

Where, a_i represents the number of samples correctly classified in class i, and n represents the total number of samples.

For the experimental data set, the three algorithms are calculated 100 times respectively. Equation (4) is used to calculate the clustering accuracy ACC of each time, and the average accuracy rate of each algorithm (i.e. the average accuracy rate ACC of 100 clustering, recorded as AACC) is used to compare the advantages and disadvantages of the algorithm. Detailed results are shown in Table 3. According to the experimental results, the proposed algorithm achieves a higher average accuracy rate than DBSCAN algorithm on all the 4 data sets. Compared with MDBSCAN, in addition to slightly poor performance on Zoo data set, the algorithm in this paper performs better on the other three data sets, especially on Balance-scale data set, which shows better than MDBSCAN, because the data sets are all ordered attributes, and the distance measurement method of the improved algorithm in this paper should be closer to the reality.

Table 3. AACC comparison with three algorithms/%

Method	Manual data	Zoo	Hayes Roth	Balance scale
DBSCAN	77.89	81.67	41.70	51.94
MDBSCAN	92.37	**87.03**	43.96	51.94
Proposed	**100.00**	86.53	45.61	**61.39**

4.3. Comparison of Clustering Distance

The clustering accuracy can only be calculated if the data class is known. However, the actual application needs to cluster the data just because the data category is unknown, so using the intra-class distance and inter-class distance after clustering to measure the clustering effect is more in line with the actual situation. The goal of clustering is to make the intra-class distance as small as possible and the inter-class distance as large as possible. The intra-class distance AAD and inter-class distance AED are used to measure clustering. The calculation formula is as follows:

$$AAD(C_r) = \frac{\sum_{x_i \in C_r} \sum_{x_j \in C_r} D(x_i, x_j)}{n_r^2}. \tag{5}$$

$$AED(C_r, C_t) = \frac{\sum_{x_i \in C_r} \sum_{x_j \in C_t} D(x_i, x_j)}{n_r n_t}. \tag{6}$$

Where C_r represents the $r-th$ class, n_r represents the number of samples within the $r-th$ class, $r \in 1, 2, \cdots, k$.

The AAD and AED are calculated according to the clustering results of the three algorithms respectively. Because the distance measurements of the three algorithms are inconsistent, the distance between samples, min-max, should be standardized before calculation. The average intra-class distance and average inter-class distance of the three algorithms on the manual dataset are shown in Table 4 (average results of 100 experiments).

It can be seen that the average intra-class distance of the improved algorithm on the artificial data set is smaller. But that does not necessarily mean the improved algorithm is better. Because the average inter-class distance is also decreasing on the artificial data set, an index is needed to evaluate the clustering effect by integrating the average inter-class distance and the average inter-class distance. A new index called cluster discrimination index (CDI) is adopted, whose value is determined by the average ratio of the average intra-class distance to the average inter-class distance. Generally speaking, a smaller value of CDI indicates a better identification of the cluster structure of the dataset, and the CDI of the cluster discrimination index is calculated as follows:

$$CDI = \frac{1}{k} \sum_{r=1}^{k} \frac{AAD(C_r)}{\frac{1}{k-1} \sum_{t \neq r} A(C_r, C_t)}. \tag{7}$$

Table 4. Average intra-class distance between classes comparison with three algorithms on the artificial data set/%

Class Type	DBSCAN C1	DBSCAN C2	DBSCAN C3	MDBSCAN C1	MDBSCAN C2	MDBSCAN C3	Proposed C1	Proposed C2	Proposed C3
C1	62.49	96.55	96.97	24.90	75.87	76.24	13.13	71.46	72.83
C2	96.55	46.85	88.88	75.87	26.64	75.34	71.46	14.05	72.20
C3	96.97	88.88	34.22	76.24	75.34	25.88	72.83	72.20	10.65

5. Conclusion

The improved algorithm proposed in this paper is more suitable for mixed classification data, simple disordered classification data or ordered classification data. Compared with the common DBSCAN algorithm and MDBSCAN algorithm, the proposed algorithm is closer to the actual data type and has better clustering effect. Although the improved algorithm has expanded the scope of application and improved the clustering effect, there are still some problems worth studying, such as: the cluster number K needs to be determined in advance, and the appropriate K value cannot be selected automatically; The central point of clustering is randomly selected, so there may be certain differences in the clustering results each time, etc. These problems may be determined by the nature of this kind of algorithm. Although many literatures have discussed this, it is still an interesting research direction in this field.

Most of the existing classification data clustering algorithms ignore the sequence characteristics of ordered classification data, and directly take it as unordered classification data to calculate. In this paper, the traditional DBSCAN algorithm is improved to make it suitable for mixed classification data (including disordered classification data and ordered classification data). The improved algorithm adopts the corresponding distance measurement for disordered attributes and ordered variables respectively, and the weight between attributes is assigned by the average entropy of attributes. This method not only considers the difference of different attribute values of the same attribute, but also considers the relationship between different attributes. Experimental results show that the improved algorithm performs better than DBSCAN algorithm and MDBSCAN algorithm.

6. Conflict of Interest

All authors disclosed no relevant relationships.

Acknowledgments. None.

References

1. Sarwar T, Seifollahi S, Chan J, et al. The secondary use of electronic health records for data mining: Data characteristics and challenges[J]. ACM Computing Surveys (CSUR), 2022, 55(2): 1-40.
2. Lin T, Li H, Yin S. Modified pyramid dual tree direction filter-based image de-noising via curvature scale and non-local mean multi-grade remnant multi-grade remnant filter[J]. International Journal of Communication Systems, 2018, 31(16).
3. Sunhare P, Chowdhary R R, Chattopadhyay M K. Internet of things and data mining: An application oriented survey[J]. Journal of King Saud University-Computer and Information Sciences, 2022, 34(6): 3569-3590.
4. Dorman K S, Maitra R. An efficient k-modes algorithm for clustering categorical datasets[J]. Statistical Analysis and Data Mining: The ASA Data Science Journal, 2022, 15(1): 83-97.
5. Xie J, Wang M, Lu X, et al. DP-k-modes: A self-tuning k-modes clustering algorithm[J]. Pattern Recognition Letters, 2022, 158: 117-124.
6. Teng L, Li H, Yin S. Im-MobiShare: An improved privacy preserving scheme based on asymmetric encryption and bloom filter for users location sharing in social network[J]. Journal of Computers, 2019, 30(3): 59-71.
7. Bansal M, Goyal A, Choudhary A. A comparative analysis of K-nearest neighbor, genetic, support vector machine, decision tree, and long short term memory algorithms in machine learning[J]. Decision Analytics Journal, 2022, 3: 100071.
8. Zhang S, Li J, Li Y. Reachable distance function for KNN classification[J]. IEEE Transactions on Knowledge and Data Engineering, 2023, 35(7): 7382-7396.
9. Alarfaj F K, Malik I, Khan H U, et al. Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms[J]. IEEE Access, 2022, 10: 39700-39715.
10. Bao W, Gu Y, Chen B, et al. Golgi_DF: Golgi proteins classification with deep forest[J]. Frontiers in Neuroscience, 2023, 17: 1197824.
11. Huang T, Zhang Q, Tang X, et al. A novel fault diagnosis method based on CNN and LSTM and its application in fault diagnosis for complex systems[J]. Artificial Intelligence Review, 2022, 55(2): 1289-1315.

12. Dessale M, Mengistu G, Mengist H M. Nanotechnology: a promising approach for cancer diagnosis, therapeutics and theragnosis[J]. International Journal of Nanomedicine, 2022, 17: 3735.
13. Arora G, Dubey A K, Jaffery Z A, et al. A comparative study of fourteen deep learning networks for multi skin lesion classification (MSLC) on unbalanced data[J]. Neural Computing and Applications, 2023, 35(11): 7989-8015.
14. Wang X, Ren H, Ren J, et al. Machine learning-enabled risk prediction of chronic obstructive pulmonary disease with unbalanced data[J]. Computer Methods and Programs in Biomedicine, 2023, 230: 107340.
15. Teng L, Li H, Yin S, et al. A Modified Advanced Encryption Standard for Data Security[J]. International Journal of Network Security, 2020, 22(1): 112-117.
16. Yu J, Li H, Yin S L, et al. Dynamic gesture recognition based on deep learning in human-to-computer interfaces[J]. Journal of Applied Science and Engineering, 2020, 23(1): 31-38.
17. Chawla N V, Bowyer K W, Hall L O, et al. SMOTE: synthetic minority over-sampling technique[J]. Journal of artificial intelligence research, 2002, 16: 321-357.
18. Nweke H F, Teh Y W, Mujtaba G, et al. Multi-sensor fusion based on multiple classifier systems for human activity identification[J]. Human-centric Computing and Information Sciences, 2019, 9: 1-44.
19. Lv M, Ren Y, Chen Y. Research on imbalanced data: based on SMOTE-AdaBoost algorithm[C]//2019 3rd International Conference on Electronic Information Technology and Computer Engineering (EITCE). IEEE, 2019: 1165-1170.
20. Popel M H, Hasib K M, Habib S A, et al. A hybrid under-sampling method (HUSBoost) to classify imbalanced data[C]//2018 21st international conference of computer and information technology (ICCIT). IEEE, 2018: 1-7.
21. Rayhan F, Ahmed S, Mahbub A, et al. Cusboost: Cluster-based under-sampling with boosting for imbalanced classification[C]//2017 2nd international conference on computational systems and information technology for sustainable solution (csitss). IEEE, 2017: 1-5.
22. Ahmed S, Mahbub A, Rayhan F, et al. Hybrid methods for class imbalance learning employing bagging with sampling techniques[C]//2017 2nd International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS). IEEE, 2017: 1-5.
23. Elyan E, Moreno-Garcia C F, Jayne C. CDSMOTE: class decomposition and synthetic minority class oversampling technique for imbalanced-data classification[J]. Neural computing and applications, 2021, 33: 2839-2851.
24. Fahim A. Adaptive Density-Based Spatial Clustering of Applications with Noise (ADBSCAN) for Clusters of Different Densities[J]. Computers, Materials & Continua, 2023, 75(2).
25. Guo J, Wu H, Chen X, et al. Adaptive SV-Borderline SMOTE-SVM algorithm for imbalanced data classification[J]. Applied Soft Computing, 2024, 150: 110986.
26. Yin S, Li H. Hot region selection based on selective search and modified fuzzy C-means in remote sensing images[J]. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2020, 13: 5862-5871.
27. Meng F, Zhao D, Zhang X. A fair consensus adjustment mechanism for large-scale group decision making in term of Gini coefficient[J]. Engineering Applications of Artificial Intelligence, 2023, 126: 106962.
28. Petersson J. Using the Gini coefficient for assessing heterogeneity within classes and schools[J]. SN Social Sciences, 2023, 3(11): 186.
29. Jisi A, Yin S. A new feature fusion network for student behavior recognition in education[J]. Journal of Applied Science and Engineering, 2021, 24(2): 133-140.
30. Teng L. Brief Review of Medical Image Segmentation Based on Deep Learning[J]. IJLAI Transactions on Science and Engineering, 2023, 1(02): 01-08.
31. Wongvorachan T, He S, Bulut O. A comparison of undersampling, oversampling, and SMOTE methods for dealing with imbalanced classification in educational data mining[J]. Information, 2023, 14(1): 54.
32. Bao Y, Yang S. Two novel SMOTE methods for solving imbalanced classification problems[J]. IEEE Access, 2023, 11: 5816-5823.
33. Yin S, Li H, Laghari A A, et al. An Anomaly Detection Model Based On Deep Auto-Encoder and Capsule Graph Convolution via Sparrow Search Algorithm in 6G Internet-of-Everything[J]. IEEE Internet of Things Journal, 2024.
34. Yin S. Object Detection Based on Deep Learning: A Brief Review[J]. IJLAI Transactions on Science and Engineering, 2023, 1(02): 1-6.
35. Qian J, Zhou Y, Han X, et al. MDBSCAN: A multi-density DBSCAN based on relative density[J]. Neurocomputing, 2024, 576: 127329.

Biography

Yuchang Si is with the Software College, Shenyang Normal University. His research direction is image processing, computer application and AI.