Multi-classification Prediction of RNA-binding Proteins based on Machine Learning

Haodong Suo

School of Computer Science and Technology, Hainan University, Haikou, 570000, China
dyx0803@stu.xjtu.edu.cn

Abstract. RNA-Binding Proteins (RBPs) have a great impact on Ribose Nucleic Acid (RNA) stability, transport, translation, splicing and other functions. Predicting the function and mechanism of the action of RNA-binding proteins is of great significance for the fields of cell signaling and metabolic regulation, as well as diagnosis of disease mechanisms. Although there are already some mature methods in the field of RBPs prediction, most of these models are binary-classification models. These models only predict whether the protein to be tested is a particular RNA-binding protein or not. Therefore, this paper focuses on the analysis and processing of pdb_data_no_dups and data_seq two data sets, combined with random forest, decision tree and K neighborhood classifiers, committing to developing a group of multi-classification prediction models. The experimental results show that in the 5-fold cross-validation, the prediction accuracy of the random forest, decision tree and K neighborhood model in this paper can reach 0.73, 0.75 and 0.92 respectively.

Keywords: RNA-binding proteins, multi-classification prediction, machine learning

1 Introduction

With the in-depth development of genome and proteome sequencing technology, more and more biological databases have accumulated huge amounts of biological data, which are inevitably huge in variety, scale and redundancy. Under such circumstances, the experimental methods of traditional biology can no longer meet the requirements for high efficiency, rapidity and accuracy [1]. As a result, the discipline of bioinformatics has been formed by combining the knowledge of statistics, life sciences and computer science, and has been widely applied in practice [2]. It brings great convenience to genome and protein sequence analysis, drug design and development, disease diagnosis and treatment.

Protein-related research is an essential division in the area of bioinformatics because proteins intervene in various biological activities such as cell structure and morphology construction, gene expression, cell growth and tissue repair, immune defence, etc., and at the same time are an important source of energy for the human

P. Kar et al. (eds.), *Proceedings of the 2023 International Conference on Image, Algorithms and Artificial Intelligence (ICIAAI 2023)*, Advances in Computer Science Research 108,
https://doi.org/10.2991/978-94-6463-300-9_38

body [1]. Nucleic acid binding proteins (NABP) are a class of proteins with the ability to bind Deoxyribo Nucleic Acid (DNA) or Ribose Nucleic Acid (RNA). In living organisms, they can regulate gene expression, control DNA repair, and participate in DNA recombination and cell division and other life activities by binding to nucleic acid molecules. RNA-binding proteins, as a kind of NABP, can interact with specific sequences, structures, or chemical modifications in RNA, thus influence on RNA stability, transport, translation, splicing and other functions 2]. Their functional diversity and wide range of functions make RNA-binding proteins important molecules in cellular metabolism and signalling pathways.

As a consequence, the study of RNA-binding proteins can provide us with a deeper understanding of the molecular mechanisms of gene expression regulation, and also help to discover new biomolecules and targets for drug design. And predicting the function and mechanism of action of RNA-binding proteins is of great significance in studying the mechanisms of cell signalling and metabolic regulation, the occurrence and development of diseases such as reproductive system diseases, neurological disorders, cardiovascular diseases, cancers and other diseases and how to manage them 3]. To accurately predict the function of RBPs, it is necessary to classify various types of RBPs with different structures and sequences.

Although there are already some mature methods in the field of RBPs prediction, most of these models are binary-classification models. There are few multi-classification studies and prediction tools for RNA-binding proteins. Therefore, in this paper, this work developed a set of 4-classification prediction models based on biological sequence feature representation and machine learning methods. The experimental results show that in the 5-fold cross-validation, the prediction accuracy achieved ideal effectiveness.

2 Materials and methods

2.1 Data sets

The dataset utilized in this paper is divided into two sections: pdb_data_no_dups and data_seq; The content in the pdb_data_no_dups (D1) dataset is RNA binding protein metadata, including classification, extraction methods, crystallization methods, resolution, pH value, etc; data_seq (D2) dataset contains the structural sequence of the former one's related proteins. Both these two datasets are available at https://www.kaggle.com/datasets/shahir/protein-data-set.

2.2 Data sets pre-processing

Firstly, this work needs to process two scattered datasets. Use the merge function to merge the two datasets D1, D2 with the structureId as index, using left joins where the structureId columns are the same. If there is no structureId in D2 that matches D1, then the columns in D2 will be populated with NaN values.

Secondly, descriptive statistics are performed on D3 using the describe function in the pandas library to compute the statistical information of each numerical feature in the dataset, as a way to have a greater overview of the basic situation of the dataset such as the data distribution and missing values. D3 dataset's situation is shown in Fig.1.

	residueCount_x	resolution	structureMolecularWeight
count	471811.000000	449845.000000	4.718110e+05
mean	6249.411993	3.020053	9.249303e+05
std	23602.912835	3.090108	3.016951e+06
min	0.000000	0.480000	3.143800e+02
25%	456.000000	2.000000	5.261474e+04
50%	1140.000000	2.500000	1.308344e+05
75%	4518.000000	3.100000	6.331348e+05
max	313236.000000	70.000000	9.773054e+07

	crystallizationTempK	densityMatthews	densityPercentSol
count	317806.000000	390156.000000	390278.000000
mean	290.882456	2.850614	54.196381
std	8.903673	0.824283	10.269266
min	4.000000	0.000000	0.000000
25%	291.000000	2.320000	46.890000
50%	293.000000	2.670000	53.950000
75%	295.000000	3.190000	61.360000
max	398.000000	99.000000	92.000000

	phValue	publicationYear	residueCount_y
count	340901.000000	414031.000000	471149.00000
mean	6.830511	2010.458932	6257.93182
std	2.461170	7.035084	23618.38381
min	0.000000	201.000000	0.00000
25%	6.100000	2007.000000	456.00000
50%	7.000000	2012.000000	1140.00000
75%	7.500000	2015.000000	4528.00000
max	724.000000	2018.000000	313236.00000

Fig. 1. D3 data content (Original)

By printing this information, it is found that some of the attributes of some proteins in D3 have null values. Therefore, this work continues to process D3 by deleting the proteins with null values, and also deleting "publicationYear", "chainId", " macromoleculeType_x", "chainId", "macromoleculeType_x" and "macromoleculeType_y", and then filtered out the top four proteins of the RNA-binding protein family (Ribosome Transferase, Hydrolase, and Ligase) to obtain the dataset D4. Partial D4 dataset is shown in Fig. 2.

structureId	classification	experimentalTechnique	resolution	structureMolecularWeight	crystallizationMethod	crystallizationTempK
1AUE	TRANSFERASE	X-RAY DIFFRACTION	2.33	24203.73	VAPOR DIFFUSION, HANGING DROP	277.0
1AUE	TRANSFERASE	X-RAY DIFFRACTION	2.33	24203.73	VAPOR DIFFUSION, HANGING DROP	277.0
1AUK	HYDROLASE	X-RAY DIFFRACTION	2.10	52423.45	VAPOR DIFFUSION, HANGING DROP	291.0

Fig. 2. Partial D4 dataset (Original)

To properly assess the dataset's features, the correlation coefficient between the columns (features) in D4 is calculated using the corr function in the pandas library, and the matrix of correlation coefficients returned by the corr function is visualized using the heatmap function in the seaborn library. The correlation coefficient plot for D4 is shown in Fig. 3.

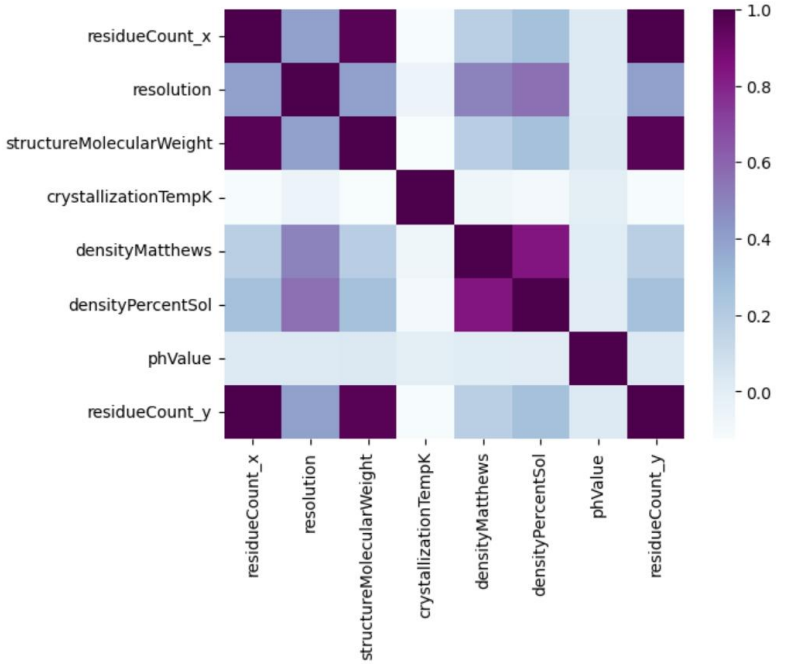

Fig. 3. The correlation coefficient plot for D4 (Original)

The value of each matrix member in the correlation coefficient matrix indicates the correlation coefficient between two variables. The correlation coefficient can have values between [-1, 1], with the bigger the absolute number indicating a better correlation between the two variables. And the sign indicates whether the correlation

is positive or negative. Correlation coefficients are usually plotted using a heat map, where the darker the colour, the closer the correlation coefficient is to 1. By looking at the correlation coefficient plot of D4, a strong correlation was found between "residueCount_x", "residueCount_y", "densityMatthews " and "densityPercentSol". As a result, D4 is processed further by deleting these four features with the goal to increase the prediction model's reliability and generality while also reducing computing complexity. At the same time, protein identifiers, secondary information such as "classification", "experimentalTechnique", "pdbxDetails" and "resolution" were deleted to get the dataset D5.

Subsequently, the feature encoding of the dataset D5 was performed using the fit_transform function contained in the OrdinalEncoder class of the sklearn library, which converted the feature information of the features such as the crystallisation method, the crystallisation temperature, and the protein sequences to numerical values, to obtain the dataset D6. In addition, the MinMaxScaler class of the preprocessing module in the sklearn library was continued to be used to normalise the maximum and minimum values of the dataset D6 by transforming the data with different ranges, units or biases into similar ranges of values to obtain the dataset D7. The normalisation process facilitates quantitative analysis using mathematical methods and boosts the rapidity and efficiency of model training.

Finally, the train_test_split function divides the dataset D7 into training and test sets, with the training set having an 80% share and the test set having a 20% share.

2.3 Classifiers

Decision tree. A decision tree is a type of tree structure. It can be either a binary or a non-binary tree 4. The full dataset space serves as the tree's root node. Each non-leaf node represents a feature property test, each branch reflects the outcome of the feature attribute in a defined range, and each leaf node carries a category. The process of employing a decision tree for decision-making begins at the root node and proceeds to the leaf node by testing the matching feature characteristics in the item to be categorized and selecting output branches depending on their results. The decision outcome is the category recorded in the leaf node. Fig. 4 depicts the decision tree's structure.

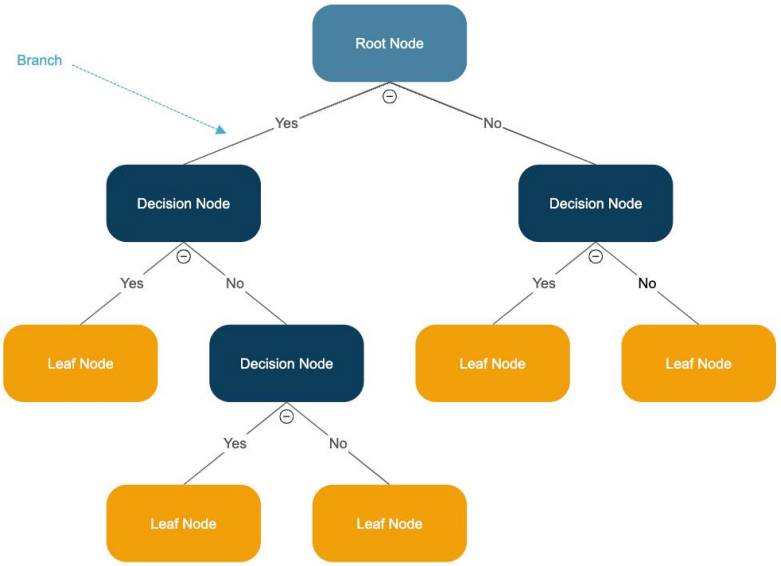

Fig. 4. The structure of decision tree [4]

The decision tree classifier has the advantage of having an algorithm that is simple to comprehend and interpret. However, they are unstable, and even if the data undergoes minor changes, the optimal decision tree structure may suffer significant changes.

Random Forest. Random Forest constructs several decision trees on the basis of decision trees by randomly selecting features and samples, and Ensemble learning synthesizes the prediction results of many decision trees 5. More formally, the decision tree is the fundamental unit of RF, and each decision tree is a classifier. By randomly selecting samples and features from the dataset, N classification results are run on N trees. Then, RF combines all categorization voting results and chooses the category with the most votes as the final output. The classification decision function is as follow:

$$H(x) = arg\ max \sum_{i=1}^{k} I(h_i(x) = Y) \tag{1}$$

where H_i denotes the i^{th} base decision tree and Y is the category to which the predicted sample belongs.

Random forest is a collection of decision trees. Therefore, for large-scale datasets, Random Forest algorithm can be used for parallel computing to speed up the training of models, and RF algorithm can process high-dimensional datasets, which is suitable for the processing of multi feature scenes. However, for the type and number of input samples, the Random forest algorithm has certain limitations, and needs to carefully adjust the parameters and select features. In addition, the Random forest algorithm

needs to occupy more computing resources, so it needs to set appropriate computing resource parameters, otherwise the algorithm will not work properly. Fig. 5 depicts the RF's structure.

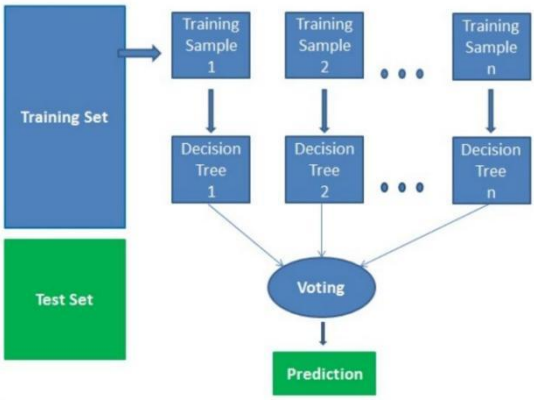

Fig. 5. The structure of random forest [5]

K neighborhood. The K Nearest Neighbor (KNN) algorithm is a classic instance based classification algorithm [6]. In KNN, to classify a new data, the first step is to find the K most similar data (neighbors) in the data, and then vote on the labels of these K neighbors to select the label that appears the most as the classification for the new data. The KNN calculating formula is as follow:

$$p(x, C_j) = \sum_{d_i \in KNN} Sim(x, d_i) y(d_i, C_j) \quad (2)$$

where x denotes the feature vector of the new sample, $sim(x, d_i)$ is the distance, and $y(d_i, C_j)$ is the category attribution function.

The advantages of K-nearest neighbor algorithm are: simple theory, easy implementation, no need for training, no restrictions on data, and it can handle multi classification problems [7-8]. Its drawback is also obvious: KNN needs to calculate the distance between each new data and all samples, and search for K nearest neighbor data in the sample set. Therefore, its computational complexity will sharply increase as the feature dimension of the dataset increases. Fig. 6 depicts the KNN's structure.

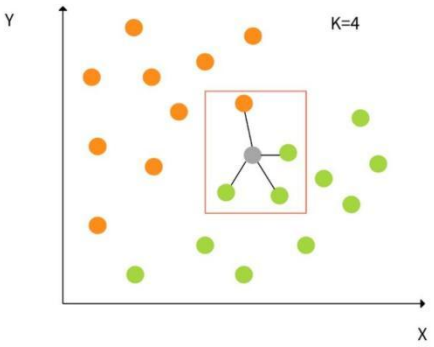

Fig. 6. The structure of KNN [7]

2.4 Grid Search CV

Grid Search CV is actually two words: GridSearch and CV, which stand for grid search and cross validation 7. Grid search is the process of altering parameters in steps within a given parameter range, then utilizing the altered parameters to train the classifier and discover the best correct parameter on the validation set. Grid search, in other words, creates and evaluates models for each combination of algorithm parameters specified in the grid. Cross validation is a technique for validating machine learning models by resampling available data. This paper created three grid search instances for hyperparameter tuning of each of the three classifiers.

3 Results and discussion

3.1 Evaluation Criteria

After the model is constructed and run, in order to have a clear understanding of the results of the model run, it need to assess the calculation with some model evaluation Criteria, commonly used model evaluation criteria are shown below.

Accuracy. Accuracy (ACC) is defined as the proportion of the number of samples that were correctly identified by a model that predicts on a test set to the total amount of samples. Accuracy is one of the simplest and most direct evaluation metrics and is usually used to assess the performance of binary or multiclassification models. Equation 1 depicts the ACC formula.

$$ACC = (TP + TN)/(TP + FN + TN + FP) \tag{3}$$

where TP denotes the number of samples that are both positive and predicted to be positive, and TN denotes the number of samples that are both negative and predicted to be negative, the number of samples that are actually negative and predicted to be

positive is referred to as FP, while the number of samples that are actually positive and projected to be negative is referred to as FN.

The advantage of the ACC is that it is simple and intuitive, easy to calculate and understand. However, in some cases, accuracy does not reflect the real performance of the model well because it does not take into account the sample distribution between different categories. Therefore, it is necessary to choose other model measures, such as confusion matrix, recall, F1-measure, etc.

Precision. Precision (Pre) is used to measure how many of all the samples predicted as positive cases by the classification model are actually positive cases. This metric is important for application scenarios where the accuracy of the prediction results needs to be guaranteed to be relatively high and misclassification is not desired. Equation 2 depicts the Pre formula.

$$Pre = TP/(TP + FP) \tag{4}$$

Recall. Recall is used to describe the proportion of a classification model that actually succeeds in capturing a particular category of a predicted sample. That is, for all actual positive examples, how many positive examples can the model correctly predict. Equation 3 depicts the recall formula.

$$Recall = TP/(TP + FN) \tag{5}$$

F1-measure. F1-measure is a combination of accuracy and recall and takes a value in the range of [0,1], with greater values indicating better model performance. If the value of F1-measure is 1, it means the performance of the model is perfect, and if the value of F1-measure is 0, it means the performance of the model is very poor. Equation 4 depicts the F1-measure formula.

$$F1 = 2 \times Pre \times Recall \,/\, (Pre + Recall) \tag{6}$$

3.2 Results

By running tests on the test set using the tuned three classifiers and plotting the confusion matrices of the respective models, it can accurately assess the classification performance of the models and provide a basis for model improvement and optimisation. The three classifier confusion matrices are shown in Fig. 7-8.

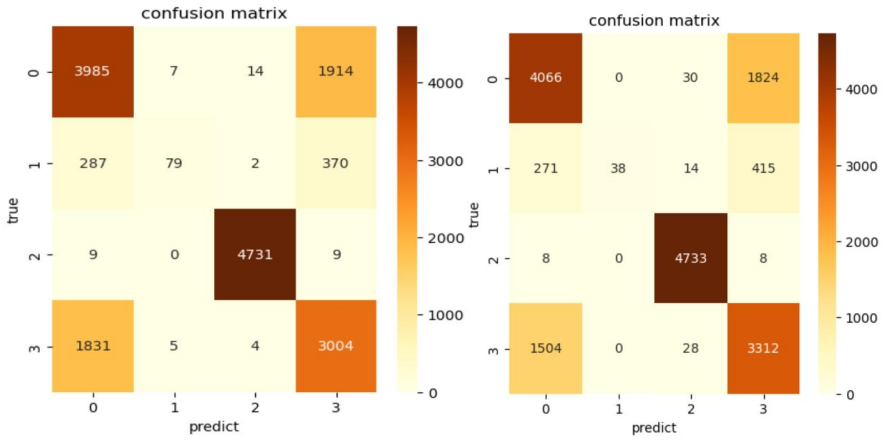

Fig. 7. The confusion matrix of decision tree and RF (Original)

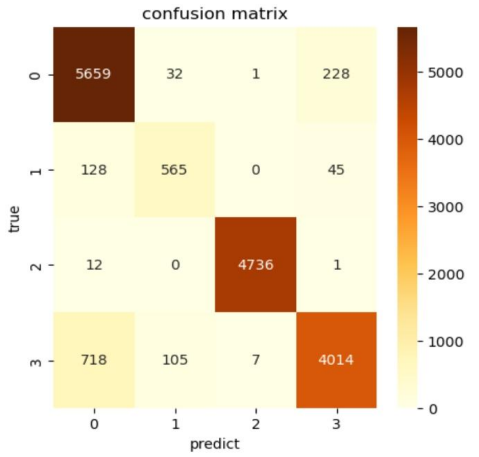

Fig. 8. The confusion matrix of KNN (Original)

The Confusion matrix is a highly essential indicator in the multi-classification problem for evaluating the effectiveness of classification of the model since it can be used to determine the magnitude of the ACC, Pre, Recall, and F1 measures [9-10]. The comparison of the three classifier evaluation metrics is shown in table 1.

Table 1. The comparison of the three classifier evaluation metrics

Method	Classification	Precision	Recall	F1-score	Accuracy
Decision Tree	0.0	0.6520	0.6731	0.6624	0.7260
	1.0	0.8681	0.1070	0.1906	

RF	2.0	0.9958	0.9962	0.9960	0.7476
	3.0	0.5671	0.6201	0.5924	
	0.0	0.6952	0.6868	0.6910	
	1.0	1.0000	0.0515	0.0979	
	2.0	0.9850	0.9966	0.9908	
	3.0	0.5958	0.6837	0.6367	
KNN	0.0	0.8683	0.9559	0.9100	0.9214
	1.0	0.8048	0.7656	0.7847	
	2.0	0.9983	0.9973	0.9978	
	3.0	0.9361	0.8287	0.8791	

By looking at the table, it can find that the best model among the three classifiers in the experiments of this paper is KNN, whose prediction accuracy is as high as 0.92, and whose Recall value indicates that the actual prediction accuracy is informative. The worst decision tree model can also achieve an accuracy of 0.73, however, its prediction of RBP with category 1.0 is poorer, with a Recall value of only 0.1, indicating that the number of samples it correctly predicts to be positive examples is small, and the ACC value is mainly contributed by negative example samples. All three classifiers predicted well for RBP with category 2.0, with all their metrics close to 1. Combined with the overall judgement of the three types of models, this group of models achieved good results for the prediction of RNA-binding protein multi-classification.

4 Conclusion

RNA-binding proteins have a significant impact on the normal operation of RNA biological functions, and the prediction of the functions and mechanisms of action of RNA-binding proteins is of extraordinary significance in the fields of cellular signalling and metabolic regulation, and diagnosis of disease mechanisms. This paper develops a set of predictive models for RBPs based on protein sequences and machine learning, which provide new ideas for low-level learners. By preprocessing two datasets, pdb_data_no_dups and data_seq, combined with three classical classifiers, namely, decision tree, RF, and KNN, which have been optimised with hyperparameters, the prediction accuracies on the test set are 0.73, 0.75, and 0.92, respectively. The findings from experimentation suggest that the set of models produces accurate predictions.

As mentioned before, the classification algorithm used in the prediction model developed in this paper is the most basic method with low robustness and generalisation, so the prediction performance may drop substantially when facing more complex experimental situations. Therefore, in the future learning process, on

one hand, this paper will try to use more professional feature extraction and feature selection methods in order to deal with more complex protein sequences; on the other hand, this work will try to integrate multiple classifier algorithms and make full use of their respective advantages with the aim to improve the model's robustness and generalisation and make the prediction findings more credible.

References

1. K Singh R, Lee J K, Selvaraj C, et al. Protein engineering approaches in the post-genomiera. Current Protein and Peptide Science, 2018, 19(1): 5-15.
2. Chengxin Z, Freddolino P L, Yang Z. COFACTOR: improved protein function prediction by combining structure, sequence and protein-protein interaction information. Nucleic Acids Research, 2017, (W1): W291-W299.
3. Wu Y, Liu Y, He A, et al. Identification of the six-RNA-binding protein signature for prognosis prediction in bladder cancer. Frontiers in Genetics, 2020, 11: 89-92.
4. Quinlan J R.Induction of Decision Tree.Machine Learning, 1986(1):81-106.
5. Breiman L. Random forest. Mach. Learn., 2001, 45:5-32.
6. Nigsch F, Bender A, Buuren B V, et al. Melting point prediction employing k-nearest neighbor algorithms and genetic parameter optimization. Journal of Chemistry and Information Model, 2006, 46: 2412-2422.
7. S. Yadav and S. Shukla, Analysis of k-Fold Cross-Validation over Hold-Out Validation on Colossal Datasets for Quality Classification, in 2016 IEEE 6th International Conference on Advanced Computing, India, 2016, pp. 78–83.
8. Powers D M. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. arXiv preprint arXiv:201016061, 2020.
9. Deng X, Liu Q, Deng Y, et al. An improved method to construct basic probability assignment based on the confusion matrix for classification problem. Information Sciences, 2016, 340: 250-261.
10. Y. Cai, D. Ji, and D. Cai, A KNN Research Paper Classification Method Based on Shared Nearest Neighbor, 2010, p. 5.