ORIGINAL RESEARCH
published: 28 January 2022
doi: 10.3389/fgene.2021.818841

A SNARE Protein Identification Method Based on iLearnPlus to Efficiently Solve the Data Imbalance Problem

Dong Ma, Zhihua Chen, Zhanpeng He and Xueqin Huang*

Institute of Computing Science and Technology, Guangzhou University, Guangdong, China

Machine learning has been widely used to solve complex problems in engineering applications and scientific fields, and many machine learning-based methods have achieved good results in different fields. SNAREs are key elements of membrane fusion and required for the fusion process of stable intermediates. They are also associated with the formation of some psychiatric disorders. This study processes the original sequence data with the synthetic minority oversampling technique (SMOTE) to solve the problem of data imbalance and produces the most suitable machine learning model with the iLearnPlus platform for the identification of SNARE proteins. Ultimately, a sensitivity of 66.67%, specificity of 93.63%, accuracy of 91.33%, and MCC of 0.528 were obtained in the cross-validation dataset, and a sensitivity of 66.67%, specificity of 93.63%, accuracy of 91.33%, and MCC of 0.528 were obtained in the independent dataset (the adaptive skip dipeptide composition descriptor was used for feature extraction, and LightGBM with proper parameters was used as the classifier). These results demonstrate that this combination can perform well in the classification of SNARE proteins and is superior to other methods.

Keywords: SNARE protein identification, ASDC features, SMOTE, data imbalance, machine learning

OPEN ACCESS

Edited by:
Quan Zou,
University of Electronic Science and Technology of China, China

Reviewed by:
Xiangxiang Zeng,
Hunan University, China
Xun Wang,
China University of Petroleum, Huadong, China

****Correspondence:***
Zhihua Chen
czhgd@gzhu.edu.cn

Specialty section:
This article was submitted to Computational Genomics, a section of the journal Frontiers in Genetics

Received: *20 November 2021*
Accepted: *14 December 2021*
Published: *28 January 2022*

Citation:
Ma D, Chen Z, He Z and Huang X (2022) A SNARE Protein Identification Method Based on iLearnPlus to Efficiently Solve the Data Imbalance Problem.
Front. Genet. 12:818841.
doi: 10.3389/fgene.2021.818841

INTRODUCTION

Soluble N-ethylmaleimide-sensitive fusion protein attachment protein receptor (SNARE) proteins are a small superfamily of proteins. They have an uncomplicated domain structure, and a feature of them is the SNARE motif—an evolutionarily conserved heptanucleotide repeat consisting of 60–70 amino acids. (Jahn and Scheller, 2006) They can be divided into Q-SNAREs and R-SNAREs pursuant to the structural characteristics of SNAREs. Functionally, SNAREs are most likely associated with various aspects of membrane transport specificity, and they are a key element in membrane fusion and are necessary for stable fusion intermediates. (Schoch et al., 2001) SNARE proteins are involved in membrane vesicle transport, such as synaptic transmission between nerve cells (synaptic vesicle transport) and plant disease resistance (disease resistance signaling). In addition, SNAREs are also implicated in the formation of some mental disorders. (Wang et al., 2018)

It is relatively complex to explore the function of a particular protein in the field of biology, the general prediction method is based on Protein-Protein-Interaction (PPI) (Hu et al., 2011; Zhai et al., 2020; Sundar and Narmadha, 2021) and protein structure information (Kinjo and Nakamura, 2012; Sharma and Srivastava, 2021). In the subsequent process, the specific function of detection through the complex biological experiment needs to be clear, which greatly increases the difficulty and the resources required of the properties that determine protein function, thus reducing the efficiency due to unavoidable time consumption.

In recent years, with the development of machine learning, many methods have achieved good results in various fields, such as Nature Language Processing (NLP) and computer vision (Jin et al., 2021). In addition, the classification task is one of the most basic applications in machine learning, and relevant research is has matured. (Ke et al., 2017) Nguyen Quoc Khanh Le, et al. (Le et al., 2019) employed PSSM profiles and 2D CNN to identify SNARE proteins. Su, Xin, et al. (Su et al., 2019) applied the multiscale convolutional network to the identification of antimicrobial peptides, so it is appropriate to apply machine learning to protein classification tasks.

In this paper, multiple feature extraction algorithms are used to extract different features, obtain the best performance descriptor through performance comparison, and then perform data enhancement processing on the extracted features of this descriptor to address the problem of sample imbalance in the data to a certain extent. Finally, the processed feature data and raw data of the independent test set were used to train the classifier to obtain the eventual model.

MATERIALS AND METHODS

The task of protein sequence classification models based on machine learning generally includes five main steps: protein sequence data collection, feature extraction and processing, classifier construction and optimization, model performance evaluation, and result visualization. (Liu et al., 2019; Guo et al., 2020; Tao et al., 2020; Chen Z et al., 2021; Li et al., 2021) The details of the first three steps determine whether the classification performance is satisfactory, while the last two steps are only a further explanation of the experimental results and determined by objective evaluation indicators, so the sequence classification task is mainly carried out using the first three steps. **Figure 1** illustrates the research flow of this paper.

Datasets

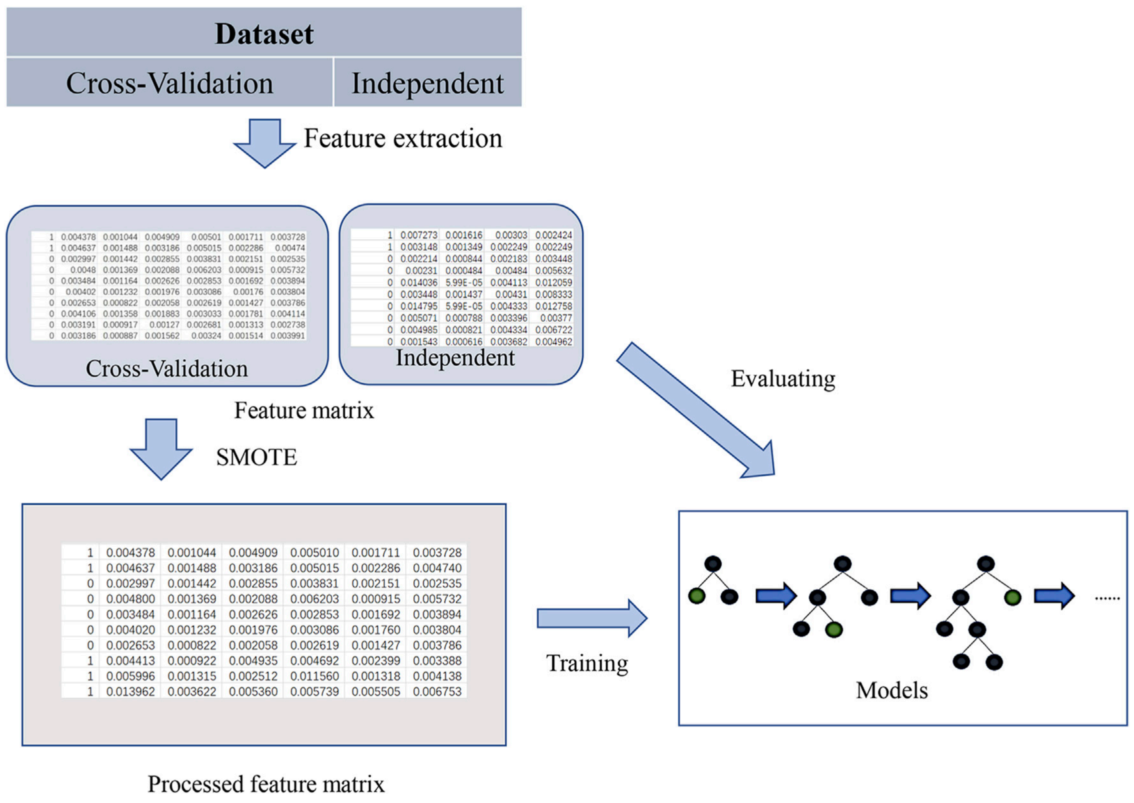
The research object of this paper are SNARE proteins, which are generally downloaded from the UniProt database. As the research object is a specific type of protein, less sequence data can be obtained for a specific protein compared to other non-specific types of common proteins, which leads to the final dataset being easily unbalanced, i.e., the number of nonspecific proteins in the dataset is greater than the number of specific proteins. The dataset used in this study was from other similar tasks. (Le and Nguyen, 2019) The number of SNARE proteins in this dataset was only one-tenth that of non-SNARE proteins, including 697 SNARE proteins as positive samples and 7,378 vesicle transport proteins as negative samples. During the experiment, 90% of them were extracted for the training of the model, and the rest were used as independent validation sets to evaluate the generalization ability of the model.

Feature Extraction and Processing

Biological sequence data are generally stored in a FASTA file format, and each sequence data is represented by the letter of the nucleotide or amino acid constituting the molecule. As the number of molecules composing the biological sequence is not fixed, the length of the sequence is inconsistent. However, traditional machine learning models can only deal with fixed-dimension data in digital format, so it is necessary to encode source sequence data into restricted-length digital data to meet the input requirements of the model, which is the feature extraction of sequential data. Descriptors are used in the first step of biological sequence analysis. They extract various biological sequence features from multiple perspectives, such as amino acid composition, biochemical characteristics, and residue composition, with different emphases and features. Consequently, these algorithms may have different performances for various sequence analysis tasks. Typically, the most appropriate algorithm for a given task needs to be obtained by testing various feature extraction algorithms on the dataset and comparing the performance of each algorithm.

Treatment of Data Imbalance

As mentioned above, the number of positive samples in the dataset used in this paper is only one 10th of the number of negative samples, which will lead to unbalanced recognition of positive and negative samples in training process and affect the final classification results (Zou et al., 2016; Cheng et al., 2018; Azad et al., 2019; Priya and Sivaraj, 2021; Shao et al., 2021). The model trained with unbalanced data will be more inclined to fit the negative instances with a large number, which will lead to the degradation of the model's classification performance for the small number of positive samples. Since there are more negative samples in the dataset than positive samples, if the source files are directly used for training, the classifier will learn too many negative samples, thus reduce the recognition ability of the model for positive samples, but this is contrary to our main purpose. Therefore, it is necessary to adopt some strategies to alleviate the problem of sample imbalance. The relatively small number of specific proteins in nature and the widespread sample imbalance in the field of biological sequence classification had also led to abundant research on the processing of unbalanced data. (Chao et al., 2019; Kaur et al., 2019; Yang et al., 2020; Ao et al., 2021a; Shao and Liu, 2021) The most common are oversampling and downsampling. Oversampling is balanced by adding redundant samples to a small number of positive samples, and the strategy can improve the recognition ability of positive samples to a certain extent, but it simply repeats positive examples and overemphasizes existing positive examples, which would urge the risk of overfitting positive examples. In the downsampling method, only a portion of the negative samples is selected for lower sampling to reduce the number of negative samples. However, this method can only improve the model's classification ability of positive samples to a certain extent. Because a few of the counterexample data are discarded, their influence in the overall sample is reduced, which may result in a large deviation model, and greatly affect the overall performance.



1	0.004378	0.001044	0.004909	0.00501	0.001711	0.003728
1	0.004637	0.001488	0.003186	0.005015	0.002286	0.00474
0	0.002997	0.001442	0.002855	0.003831	0.002151	0.002535
0	0.0048	0.001369	0.002088	0.006203	0.000915	0.005732
0	0.003484	0.001164	0.002626	0.002853	0.001692	0.003894
0	0.00402	0.001232	0.001976	0.003086	0.00176	0.003804
0	0.002653	0.000822	0.002058	0.002619	0.001427	0.003786
0	0.004106	0.001358	0.001883	0.003033	0.001781	0.004114
0	0.003191	0.000917	0.00127	0.002681	0.001313	0.002738
0	0.003186	0.000887	0.001562	0.00324	0.001514	0.003991

1	0.007273	0.001616	0.00303	0.002424
1	0.003148	0.001349	0.002249	0.002249
0	0.002214	0.000844	0.002183	0.003448
0	0.00231	0.000484	0.00484	0.005632
0	0.014036	5.99E-05	0.004113	0.012059
0	0.003448	0.001437	0.00431	0.008333
0	0.014795	5.99E-05	0.004333	0.012758
0	0.005071	0.000788	0.003396	0.00377
0	0.004985	0.000821	0.004334	0.006722
0	0.001543	0.000616	0.003682	0.004962

1	0.004378	0.001044	0.004909	0.005010	0.001711	0.003728
1	0.004637	0.001488	0.003186	0.005015	0.002286	0.004740
0	0.002997	0.001442	0.002855	0.003831	0.002151	0.002535
0	0.004800	0.001369	0.002088	0.006203	0.000915	0.005732
0	0.003484	0.001164	0.002626	0.002853	0.001692	0.003894
0	0.004020	0.001232	0.001976	0.003086	0.001760	0.003804
0	0.002653	0.000822	0.002058	0.002619	0.001427	0.003786
1	0.004413	0.000922	0.004935	0.004692	0.002399	0.003388
1	0.005996	0.001315	0.002512	0.011560	0.001318	0.004138
1	0.013962	0.003622	0.005360	0.005739	0.005505	0.006753

FIGURE 1 | The research flow diagram of SNARE protein identification using a decision tree model.

Considering the serious imbalance between positive and negative samples in this dataset, only one unbalanced strategy may not work well; it needs to be sampled up and down simultaneously. This article uses a combination of sampling partial negative samples and the synthetic minority oversampling technique (SMOTE) to generate new positive samples to address sample imbalance. (Chawla et al., 2002; Riaz and Li, 2019; Zhang C H et al., 2020; Zhao et al., 2020)

SMOTE is an oversampling technique that balances the quantity gap between two categories by finding the nearest neighbor of certain data in a positive example and then using the K-nearest neighbor algorithm to generate new positive samples. For each sample x in the positive sample, calculate the K positive samples x_k {k = 1, 2, K} closest to x, and determine the sampling ratio n according to the unbalanced proportion of samples. For the k nearest neighbor samples of each sample x, n samples are randomly selected, and the newly constructed sample x_{new} can be obtained through the following formula:

$$x_{new} = x + rand(0,1)*|x - x_n| \quad (1)$$

In the experiment, part of the negative sample is treated with simple undersampling at first. SMOTE is used to generate positive samples to ensure that the number of positive and negative samples is consistent. Then, a balanced dataset of sample size can be obtained, which will be used in subsequent model training experiments.

RESULT AND DISCUSSION

Evaluation Indexes

To objectively evaluate the performance of various algorithms, some convincing indicators of these algorithms need to be compared after the experiment (Wei et al., 2017; Wei et al., 2018; Wei et al., 2019; Wang et al., 2020; Ding et al., 2021; Shang et al., 2021; Wu and Yu, 2021; Yang et al., 2021). Next, the algorithm with the best performance is selected for subsequent research according to these indices. Similarly, common metrics are used to compare the performance of each algorithm. The four values of TP, FP, TN, and FN (representing true positive, false-positive, true negative, and false negative values, respectively) can be obtained for the classifier test (Jiang et al., 2013; Cheng et al., 2016; Xiao et al., 2019; Zhang L et al., 2020; Huang et al., 2020; Li and Liu, 2020; Liu et al., 2020; Mo et al., 2020; Tang et al., 2020; Han et al., 2021; Wang et al., 2021; Xu et al., 2021). Accuracy, MCC, sensitivity, and specificity can then be calculated based on these values.

$$Sensitivity = \frac{TP}{TP + FN} \quad (2)$$

$$Specificity = \frac{TN}{TN + FP} \quad (3)$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (4)$$

TABLE 1 | Feature dimensions of partial feature extraction algorithms and AUROC performance under multiple classifiers.

	Feature dimension	RandomForest (Breiman, 2001)	LightGBM (Ke et al., 2017)	XGBoost (Chen and Guestrin, 2016)
ASDC (Wei et al., 2018)	400	**0.8599**	**0.8829**	**0.8839**
QSOrder (Chou, 2000)	44	0.8401	0.864	0.8604
DDE (Saravanan and Gautham, 2015)	400	0.824	0.8604	0.849
CKSAAP (Chen et al., 2007)	1,600	0.8337	0.8664	0.8588
AAC (Bhasin and Raghava, 2004)	20	0.8467	0.8514	0.8428

The meaning of the bold values is the feature extraction algorithm that performs best under a particular classification algorithm.

TABLE 2 | Model performance under different n values.

n	Cross-validation				Independent			
	Sens	Spec	Acc	MCC	Sens	Spec	Acc	MCC
628	82.97	82.954	82.486	0.6522	94.2	60.84	63.69	0.3102
1,256	95.148	89.886	92.516	0.8523	73.91	76.56	76.33	0.3152
2,510	98.486	91.832	95.158	0.9055	76.81	86.34	85.5	0.4492
3,764	99.07	94.394	96.73	0.936	65.22	93.22	90.83	0.5071
5,019	99.302	94.682	96.99	0.941	62.32	94.04	91.33	0.5081
6,640	99.292	94.414	96.852	0.9384	59.42	94.99	91.95	0.5149

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{5}$$

Selection of the Descriptors

In this paper, the iLearnPlus platform (Chen Z et al., 2021) was used to compare the performance of various extraction algorithms: multiple descriptors were applied to obtain the feature vectors of the source FASTA file, followed by training and testing the obtained features using several classification algorithms and analyzing the performance of different feature extraction algorithms. To eliminate the influence of other subjective factors, the area under the receiver operating characteristic curve (AUROC) index was adopted to evaluate the performance of the algorithm.

Accuracy and MCC are widely adopted to measure model performance in classification problems. These two values can be regulated by artificially setting thresholds so that the specific performance of each algorithm cannot be truly reflected. The AUROC index takes TPR [TP/(TP + FN)] and FPR[FP/(FP + TN)] as the horizontal and vertical coordinates to obtain the area under the curve. The larger the area is, the higher the coincidence degree between the prediction label of the model and the source label is. It is necessary to take the AUROC as the evaluation standard so that the algorithm with the best overall performance can be selected.

According to the experiment, several feature extraction algorithms and classifiers with better performance can be obtained. Some experimental results are shown in **Table 1**.

The experimental results show that the performance of adaptive skip dipeptide composition (ASDC), CKSAAP, and QSOrder feature extraction algorithms outperform other algorithms. Among them, the optimal algorithm is the ASDC, and the subsequent multiple numbers also use ASDC to extract features.

ASDC is a feature extraction algorithm based on GDC (G-gap dipeptide composition) algorithm. Dipeptide composition is the fraction of any two adjacent residues as a dipeptide pair, and it measures the correlation of any two adjacent residues in the peptide sequence. GDC encapsulates the composition and local order information of any two spacer residues in the peptide sequence, it has a hyperparameter *g* to determine the gap between two adjacent residues. And ASDC calculates all values of *g* and accumulates them. For a given protein read R with L length, the feature vector for ASDC is represented by:

$$ASDC = \left(fv_1, fv_2..., fv_{400}\right) \tag{6}$$

where fv_i is calculated by

$$fv_i = \frac{\sum_{g=1}^{L-1} O_i^g}{\sum_{i=1}^{400} \sum_{g=1}^{L-1} O_i^g} \tag{7}$$

where *g* represents the g-gap (g = 1, 2, L-1) dipeptide and fv_i is the occurrence frequency of the ith (i = 1, 2, 400) adaptive skip dipeptide. It is worth mentioning that if the cumulative term with g is removed from **Eq. 7**, it becomes the formula for the GDC features.

Since there are approximately 8,000 samples in the dataset, the 400 dimension is relatively moderate. Another is that ASDC considers the frequency of any two unconnected amino acids in the whole protein and can capture all the information of dipeptide composition. It also shows that the SNARE proteins have a high correlation with their dipeptide composition. This information may bring biological assistance to the final SNARE protein recognition.

TABLE 3 | The performance of the three classifiers on the independent test set (n = 2,510).

n = 2,510	Sensitivity (%)	Specificity (%)	Accuracy (%)	MCC
RandomForest	63.77	**90.92**	**88.6**	0.444
LightGBM	**76.81**	86.31	85.5	**0.4492**
XGBoost	73.91	86.99	85.87	0.4412

The meaning of the bold values is the feature extraction algorithm that performs best under a particular classification algorithm.

TABLE 4 | The performance of the three classifiers on the independent test set (n = 5,019).

n = 5,019	Sensitivity (%)	Specificity (%)	Accuracy (%)	MCC
RandomForest	46.38	95.66	91.45	0.435
LightGBM	**60.87**	**95.39**	**92.44**	**0.5386**
XGBoost	**60.87**	94.58	91.7	0.5132

The meaning of the bold values is the feature extraction algorithm that performs best under a particular classification algorithm.

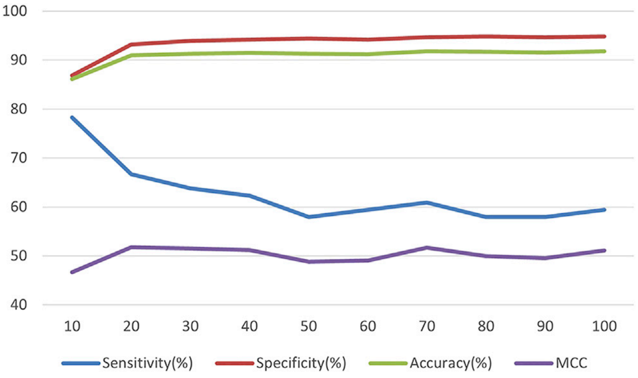


FIGURE 2 | The relationship between the number of leaves and model performance.

However, the results also showed that several other algorithms performed only slightly worse than ASDC, so it was considered that features stitched together after using multiple feature extraction algorithms could be used to train the model. After experimental verification, when the feature data extracted by algorithms such as ASDC and QSOrder were spliced together and then used to train the model, it was found that instead of improving the results, there was a slight decrease. In response to this result, it is believed that the data dimensionality is too large, and the resulting redundant data will not only have a positive effect on the training of the model but also degrade the model performance. Therefore, the spliced features were subsequently selected again, and relevant experiments were conducted. However, the model trained with these data still performed poorly on the independent set. After comparing the feature vectors extracted by the feature extraction algorithms used, it was concluded that the main reason was that the feature values obtained by each algorithm did not fall within the same range of values. For example, the feature matrix extracted by the QSOrder algorithm is a sparse matrix containing a large number of 0 or very close to 0 values, and there are some negative numbers in the DDE features, which when mixed together may affect the direction of the model iteration and thus the final results.

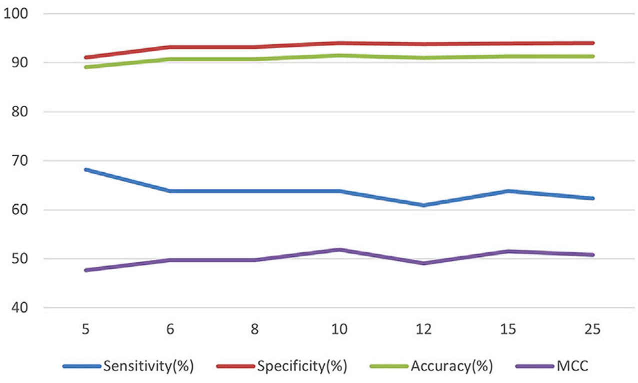


FIGURE 3 | The relationship between the number of maxdepth and model performance.

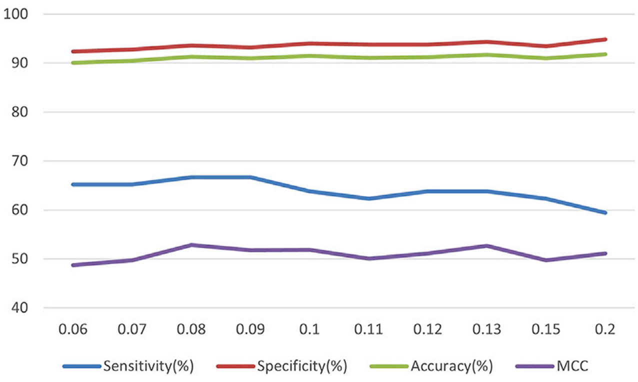


FIGURE 4 | The relationship between learning rate and model performance.

Unbalanced Processing

In the step of dealing with the data imbalance problem, n negative samples are first downsampled from the original dataset to ensure that n is greater than the number of positive samples 628. Then, the SMOTE algorithm is used to expand the number of positive samples to n to build a balanced dataset. When n = 628, the strategy is equivalent to complete downsampling, and when n = 6,640 (the total number of negative samples), the strategy is equivalent to complete oversampling, so the value of n is in the range (628, 6,640). After sampling the negative samples, all data were tested with the same independent test set to determine their generalization ability.

To analyze the effect of the number of down samples n on the classification performance, several sets of parameters were set for experiments in this paper, and the best performing n value was selected based on the results. n values were set, and the related performance is shown in **Table 2**. To partially eliminate the error caused by the randomness of the data, no put-back sampling was

TABLE 5 | Comparison with the experimental results of 2D CNN in the same setting.

Classifier	Cross-validation				Independent			
	Sens	Spec	Acc	MCC	Sens	Spec	Acc	MCC
2D CNN (Tao et al., 2020)	76.6	93.5	89.7	0.7	65.8	90.3	87.9	0.46
This Methods	98.168	90.736	94.718	0.8974	81.58	94.84	93.54	0.6839

performed in downsampling, and the set of negative samples sampled was denoted by S (n). Then, there was S(n)⊂S(m), where n < m.

Parameter Optimization

In the recognition problem, it is also very important to select the appropriate classifier. There are also multiple classifiers in the field, each with a different focus, so their performance in a particular task may be different. Therefore, to select a classifier that best fits the task, we follow the same approach as in the selection of the descriptors subsection, where different classifiers are used to train and classify the same feature data, and the best performing algorithm is selected for subsequent experiments. After using three mainstream classifiers, the model performance corresponding to the parameters of Part n is shown in **Tables 3** and **4**. It can be concluded that LightGBM with n = 5,019 is the best performer and most in line with this task. LightGBM (Light Gradient Boosting Machine) is a framework for implementing the GBDT (Gradient Boosting Decision Tree) algorithm, which is an iterative decision tree algorithm consisting of multiple decision trees. LightGBM improves on the traditional GBDT algorithm in many ways, such as using a Histogram-based decision tree algorithm and using a leaf-wise strategy instead of level-wise.

In this experiment, the number of leaf nodes, the maximum depth of the tree and the learning rate of the LightGBM algorithm were adjusted (Ao et al., 2021b). First, we compared the impact of the number of leaf nodes of the tree on the performance of the algorithm when the maximum depth of the tree was not limited. The result is shown in **Figure 2** (The MCC values in the figure have been normalized with the other three indicators for plotting purposes, and the following similar charts have been followed in the same way). Through a series of comparative experiments, the number of leaf nodes can be set to 31 while considering the efficiency of the algorithm operation.

This is followed by choosing the depth of the tree given the number of leaf nodes, as there is a maxdepth>2leaves-1 constraint, and the leaf value has been set to 31; the maximum depth of the tree cannot be less than 5 (log2 (31 + 1). The result is shown in **Figure 3**. Similarly, the optimal maxdepth can be chosen as 10.

Then, it is time to adjust the learning rate and compare the impact of changes in the learning rate on performance, and the results are shown in **Figure 4**. In the end, the optimal parameters are leaves = 31, maxdepth = 10, and learning rate = 0.08.

Comparison With the Other Method

In comparison with 2D CNN, the data of this paper needed to be modified because the data allocation differed. It used a cross-validation set of 644 positive and 2,234 counterexamples and an independent dataset of 38 positive and 349 counterexamples. Similar experiments were conducted using this setup in this paper. In this sequence classification task, the focus is on the classification performance of the SNARE protein, which in the model performance evaluation is the size of the specificity. The experimental results are shown in **Table 5**. It can be found that all the metrics performed better except for the specificity on the cross-validation set, which was slightly weaker than 2D CNN, and the method had an AUROC value of 0.9671 under the independent set, which further proves that the algorithm in this paper has a high generalization capability. The main reason for this result is that the original paper used more positive samples for training the model, with fewer positive examples remaining to evaluate the applicability of the model. However, a set partitioning ratio of 9:1 (cross validation dataset: independent dataset) was applied in this experiment, and although this may lead to some performance loss, the best results obtained in the independent dataset were still good: sensitivity of 66.67%, specificity of 93.63%, accuracy of 91.33%, and MCC of 0.528.

CONCLUSION

In this paper, we used the SMOTE algorithm with different parameters to address the sample imbalance of the dataset. The results show that this strategy can obtain a better result in terms of managing sample imbalance. In this process, ASDC as the feature extraction algorithm and LightGBM as the classification algorithm by comparing the results of various algorithms and descriptors. The combination obtained the best performance, and compared to other advanced neural networks, it achieved a significant improvement in all the typical measurement indexes. Under the same experimental setup, the method in this paper improves the accuracy by 5.64% in the independent test set and 0.2239 in the MCC metric relative to 2D-CNN. For the future research, graph neural networks (Zeng et al., 2020; Chen Y et al., 2021) and unsupervised learning (Xu et al., 2019a; Xu et al., 2019b) can be considered for performance improvement.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: https://github.com/khanhlee/snare-cnn.

AUTHOR CONTRIBUTIONS

DM conceived and designed the experiments, performed the experiments, analyzed the data, prepared figures and tables, performed the computation work, authored or reviewed drafts of the paper. ZC conceived and designed the experiments, authored or reviewed drafts of the paper. ZH and XH performed the experiments, analyzed the data, revised the manuscript. All authors read and approved the final manuscript.

FUNDING

This work was supported by the Chinese National Natural Science Foundation under Grant 61876047.

REFERENCES

Ao, C., Yu, L., and Zou, Q. (2021). Prediction of Bio-Sequence Modifications and the Associations with Diseases. *Brief. Funct. Genomics* 20 (1), 1–18. doi:10.1093/bfgp/elaa023

Ao, C., Zou, Q., and Yu, L. (2021). RFhy-m2G: Identification of RNA N2-Methylguanosine Modification Sites Based on Random forest and Hybrid Features. *Methods (San Diego, Calif.)* S1046-2023 (21), 00142. doi:10.1016/j.ymeth.2021.05.016

Azad, M. T. A., Qulsum, U., and Tsukahara, T. (2019). Comparative Activity of Adenosine Deaminase Acting on RNA (ADARs) Isoforms for Correction of Genetic Code in Gene Therapy. *Cgt* 19 (1), 31–39. doi:10.2174/1566523218666181114122116

Bhasin, M., and Raghava, G. P. S. (2004). Classification of Nuclear Receptors Based on Amino Acid Composition and Dipeptide Composition. *J. Biol. Chem.* 279 (22), 23262–23266. doi:10.1074/jbc.m401932200

Breiman, L. (2001). Random Forests. *Machine Learn.* 45 (1), 5–32. doi:10.1023/a:1010933404324

Chao, L., Wei, L., and Zou, Q. (2019). SecProMTB: A SVM-Based Classifier for Secretory Proteins of *Mycobacterium tuberculosis* with Imbalanced Data Set. *Proteomics* 19, e1900007.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *jair* 16, 321–357. doi:10.1613/jair.953

Chen, K., Kurgan, L. A., and Ruan, J. (2007). Prediction of Flexible/rigid Regions from Protein Sequences Using K-Spaced Amino Acid Pairs. *BMC Struct. Biol.* 7 (1), 25–13. doi:10.1186/1472-6807-7-25

Chen, T., and Guestrin, C. (2016). in Xgboost: A scalable tree boosting system, Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, California, San Francisco, USA, August 13 - 17, 2016 Association for Computing Machinery. 785–794.

Chen, Y., Ma, T., Yang, X., Wang, J., Song, B., and Zeng, X. (2021). MUFFIN: Multi-Scale Feature Fusion for Drug-Drug Interaction Prediction. *Bioinformatics* 37, 2651–2658. doi:10.1093/bioinformatics/btab169

Chen Z, Z., Zhao, P., Li, C., Li, F., Xiang, D., Chen, Y.-Z., et al. (2021). iLearnPlus: a Comprehensive and Automated Machine-Learning Platform for Nucleic Acid and Protein Sequence Analysis, Prediction and Visualization. *Nucleic Acids Res.* 49 (10), e60. doi:10.1093/nar/gkab122

Cheng, L., Hu, Y., Sun, J., Zhou, M., and Jiang, Q. (2018). DincRNA: a Comprehensive Web-Based Bioinformatics Toolkit for Exploring Disease Associations and ncRNA Function. *Bioinformatics* 34 (11), 1953–1956. doi:10.1093/bioinformatics/bty002

Cheng, L., Shi, H., Wang, Z., Hu, Y., Yang, H., Zhou, C., et al. (2016). IntNetLncSim: an Integrative Network Analysis Method to Infer Human lncRNA Functional Similarity. *Oncotarget* 7 (30), 47864–47874. doi:10.18632/oncotarget.10012

Chou, K.-C. (2000). Prediction of Protein Subcellular Locations by Incorporating Quasi-Sequence-Order Effect. *Biochem. biophysical Res. Commun.* 278 (2), 477–483. doi:10.1006/bbrc.2000.3815

Ding, Y., Yang, C., Tang, J., and Guo, F. (2021). Identification of Protein-Nucleotide Binding Residues via Graph Regularized K-Local Hyperplane Distance Nearest Neighbor Model. *Applied Intelligence*, 1–15. doi:10.1007/s10489-021-02737-0

Guo, Z., Wang, P., Liu, Z., and Zhao, Y. (2020). Discrimination of Thermophilic Proteins and Non-thermophilic Proteins Using Feature Dimension Reduction. *Front. Bioeng. Biotechnol.* 8, 584807. doi:10.3389/fbioe.2020.584807

Han, X., Kong, Q., Liu, C., Cheng, L., and Han, J. (2021). SubtypeDrug: a Software Package for Prioritization of Candidate Cancer Subtype-specific Drugs. *Bioinformatics*. 1. btab011. doi:10.1093/bioinformatics/btab011

Hu, L., Huang, T., Shi, X., Lu, W.-C., Cai, Y.-D., and Chou, K.-C. (2011). Predicting Functions of Proteins in Mouse Based on Weighted Protein-Protein Interaction Network and Protein Hybrid Properties. *PloS one* 6 (1), e14556. doi:10.1371/journal.pone.0014556

Huang, Y., Zhou, D., Wang, Y., Zhang, X., Su, M., Wang, C., et al. (2020). Prediction of Transcription Factors Binding Events Based on Epigenetic Modifications in Different Human Cells. *Epigenomics* 12 (16), 1443–1456. doi:10.2217/epi-2019-0321

Jahn, R., and Scheller, R. H. (2006). SNAREs - Engines for Membrane Fusion. *Nat. Rev. Mol. Cel Biol* 7 (9), 631–643. doi:10.1038/nrm2002

Jiang, Q., Wang, G., Jin, S., Li, Y., and Wang, Y. (2013). Predicting Human microRNA-Disease Associations Based on Support Vector Machine. *Ijdmb* 8 (3), 282–293. doi:10.1504/ijdmb.2013.056078

Jin, S., Zeng, X., Xia, F., Huang, W., and Liu, X. (2021). Application of Deep Learning Methods in Biological Networks. *Brief. Bioinform.* 22 (2), 1902–1917. doi:10.1093/bib/bbaa043

Kaur, H., Pannu, H. S., and Malhi, A. K. (2019). A Systematic Review on Imbalanced Data Challenges in Machine Learning. *ACM Comput. Surv.* 52 (4), 1–36. doi:10.1145/3343440

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., et al. (2017). Lightgbm: A Highly Efficient Gradient Boosting Decision Tree. *Adv. Neural Inf. Process. Syst.* 30, 3146–3154.

Kinjo, A. R., and Nakamura, H. (2012). Composite Structural Motifs of Binding Sites for Delineating Biological Functions of Proteins. *PloS one* 7 (2), e31437. doi:10.1371/journal.pone.0031437

Le, N. Q. K., and Nguyen, V.-N. (2019). SNARE-CNN: a 2D Convolutional Neural Network Architecture to Identify SNARE Proteins from High-Throughput Sequencing Data. *PeerJ Comp. Sci.* 5, e177. doi:10.7717/peerj-cs.177

Le, N. Q. K., Yapp, E. K. Y., Nagasundaram, N., Chua, M. C. H., and Yeh, H.-Y. (2019). Computational Identification of Vesicular Transport Proteins from Sequences Using Deep Gated Recurrent Units Architecture. *Comput. Struct. Biotechnol. J.* 17, 1245–1254. doi:10.1016/j.csbj.2019.09.005

Li, C.-C., and Liu, B. (2020). MotifCNN-fold: Protein Fold Recognition Based on Fold-specific Features Extracted by Motif-Based Convolutional Neural Networks. *Brief. Bioinform.* 21 (6), 2133–2141. doi:10.1093/bib/bbz133

Li, H.-L., Pang, Y.-H., and Liu, B. (2021). BioSeq-BLM: a Platform for Analyzing DNA, RNA and Protein Sequences Based on Biological Language Models. *Nucleic Acids Res.* 1. gkab829. doi:10.1093/nar/gkab829

Liu, B., Gao, X., and Zhang, H. (2019). BioSeq-Analysis2.0: an Updated Platform for Analyzing DNA, RNA and Protein Sequences at Sequence Level and Residue Level Based on Machine Learning Approaches. *Nucleic Acids Res.* 47 (20), e127. doi:10.1093/nar/gkz740

Liu, B., Li, C.-C., and Yan, K. (2020). DeepSVM-fold: Protein Fold Recognition by Combining Support Vector Machines and Pairwise Sequence Similarity Scores Generated by Deep Learning Networks. *Brief. Bioinform.* 21 (5), 1733–1741. doi:10.1093/bib/bbz098

Mo, F., Luo, Y., Fan, D., Zeng, H., Zhao, Y., Luo, M., et al. (2020). Integrated Analysis of mRNA-Seq and miRNA-Seq to Identify C-MYC, YAP1 and miR-3960 as Major Players in the Anticancer Effects of Caffeic Acid Phenethyl Ester in Human Small Cell Lung Cancer Cell Line. *Cgt* 20 (1), 15–24. doi:10.2174/1566523220666200523165159

Priya, R. D., and Sivaraj, R. (2021). Gene Selection in Multi-Class Imbalanced Microarray Datasets Using Dynamic Length Particle Swarm Optimization. *Cbio* 16 (5), 734–748. doi:10.2174/1574893615999201002093834

Riaz, F., and Li, D. (2019). Non-coding RNA Associated Competitive Endogenous RNA Regulatory Network: Novel Therapeutic Approach in Liver Fibrosis. *Cgt* 19 (5), 305–317. doi:10.2174/1566523219666191107113046

Saravanan, V., and Gautham, N. (2015). Harnessing Computational Biology for Exact Linear B-Cell Epitope Prediction: a Novel Amino Acid Composition-Based Feature Descriptor. *Omics: a J. Integr. Biol.* 19 (10), 648–658. doi:10.1089/omi.2015.0095

Schoch, S., Dea´k, F., Königstorfer, A., Mozhayeva, M., Sara, Y., Südhof, T. C., et al. (2001). SNARE Function Analyzed in Synaptobrevin/VAMP Knockout Mice. *Science* 294 (5544), 1117–1122. doi:10.1126/science.1064335

Shang, Y., Gao, L., Zou, Q., and Yu, L. (2021). Prediction of Drug-Target Interactions Based on Multi-Layer Network Representation Learning. *Neurocomputing* 434, 80–89. doi:10.1016/j.neucom.2020.12.068

Shao, J., and Liu, B. (2021). ProtFold-DFG: Protein Fold Recognition by Combining Directed Fusion Graph and PageRank Algorithm. *Brief Bioinform* 22 (3), bbaa192. doi:10.1093/bib/bbaa192

Shao, J., Yan, K., and Liu, B. (2021). FoldRec-C2C: Protein Fold Recognition by Combining Cluster-To-Cluster Model and Protein Similarity Network. *Brief Bioinform* 22 (3), bbaa144. doi:10.1093/bib/bbaa144

Sharma, A. K., and Srivastava, R. (2021). Protein Secondary Structure Prediction Using Character Bi-gram Embedding and Bi-LSTM. *Cbio* 16 (2), 333–338. doi:10.2174/1574893615999200601122840

Su, X., Xu, J., Yin, Y., Quan, X., and Zhang, H. (2019). Antimicrobial Peptide Identification Using Multi-Scale Convolutional Network. *BMC bioinformatics* 20 (1), 730–810. doi:10.1186/s12859-019-3327-y

Sundar, G. N., and Narmadha, D. (2021). An Automated Model for Target Protein Prediction in PPI. *Cbio* 16 (4), 601–609. doi:10.2174/1574893615999200831142241

Tang, Y.-J., Pang, Y.-H., Liu, B., and Idp-Seq2Seq (2020). IDP-Seq2Seq: Identification of Intrinsically Disordered Regions Based on Sequence to Sequence Learning. *Bioinformaitcs* 36 (21), 5177–5186. doi:10.1093/bioinformatics/btaa667

Tao, Z., Li, Y., Teng, Z., and Zhao, Y. (2020). A Method for Identifying Vesicle Transport Proteins Based on LibSVM and MRMD. *Comput. Math. Methods Med.* 2020, 8926750. doi:10.1155/2020/8926750

Wang, H., Ding, Y., Tang, J., and Guo, F. (2020). Identification of Membrane Protein Types via Multivariate Information Fusion with Hilbert-Schmidt Independence Criterion. *Neurocomputing* 383, 257–269. doi:10.1016/j.neucom.2019.11.103

Wang, X., Yang, Y., Liu, J., and Wang, G. (2021). The Stacking Strategy-Based Hybrid Framework for Identifying Non-coding RNAs. *Brief Bioinform* 22 (5), bbab023. doi:10.1093/bib/bbab023

Wang, Y.-N., Figueiredo, D., Sun, X.-D., Dong, Z.-Q., Chen, W.-B., Cui, W.-P., et al. (2018). Controlling of Glutamate Release by Neuregulin3 via Inhibiting the Assembly of the SNARE Complex. *Proc. Natl. Acad. Sci. USA* 115 (10), 2508–2513. doi:10.1073/pnas.1716322115

Wei, L., Zhou, C., Chen, H., Song, J., and Su, R. (2018). ACPred-FL: a Sequence-Based Predictor Using Effective Feature Representation to Improve the Prediction of Anti-cancer Peptides. *Bioinformatics* 34 (23), 4007–4016. doi:10.1093/bioinformatics/bty451

Wei, L., Xing, P., Shi, G., Ji, Z., and Zou, Q. (2019). Fast Prediction of Protein Methylation Sites Using a Sequence-Based Feature Selection Technique. *Ieee/acm Trans. Comput. Biol. Bioinf.* 16 (4), 1264–1273. doi:10.1109/tcbb.2017.2670558

Wei, L., Xing, P., Zeng, J., Chen, J., Su, R., and Guo, F. (2017). Improved Prediction of Protein-Protein Interactions Using Novel Negative Samples, Features, and an Ensemble Classifier. *Artif. Intelligence Med.* 83, 67–74. doi:10.1016/j.artmed.2017.03.001

Wu, X., and Yu, L. (2021). *EPSOL: Sequence-Based Protein Solubility Prediction Using Multidimensional Embedding*. Oxford, England: Bioinformatics.

Xiao, X., Xu, Z.-C., Qiu, W.-R., Wang, P., Ge, H.-T., and Chou, K.-C. (2019). iPSW(2L)-PseKNC: A Two-Layer Predictor for Identifying Promoters and Their Strength by Hybrid Features via Pseudo K-Tuple Nucleotide Composition. *Genomics* 111 (6), 1785–1793. doi:10.1016/j.ygeno.2018.12.001

Xu, H., Zeng, W., Zeng, X., and Yen, G. G. (2019). An Evolutionary Algorithm Based on Minkowski Distance for many-objective Optimization. *IEEE Trans. Cybern.* 49 (11), 3968–3979. doi:10.1109/tcyb.2018.2856208

Xu, H., Zeng, W., Zhang, D., and Zeng, X. (2019). MOEA/HD: A Multiobjective Evolutionary Algorithm Based on Hierarchical Decomposition. *IEEE Trans. Cybern.* 49 (2), 517–526. doi:10.1109/tcyb.2017.2779450

Xu, Z., Luo, M., Lin, W., Xue, G., Wang, P., Jin, X., et al. (2021). DLpTCR: an Ensemble Deep Learning Framework for Predicting Immunogenic Peptide Recognized by T Cell Receptor. *Brief Bioinform* 22, bbab335. doi:10.1093/bib/bbab335

Yang, C., Ding, Y., Meng, Q., Tang, J., and Guo, F. (2021). Granular Multiple Kernel Learning for Identifying RNA-Binding Protein Residues via Integrating Sequence and Structure Information. *Neural Comput. Appl.* 33 (17), 11387–11399. doi:10.1007/s00521-020-05573-4

Yang, X.-F., Zhou, Y.-K., Zhang, L., Gao, Y., and Du, P.-F. (2020). Predicting LncRNA Subcellular Localization Using Unbalanced Pseudo-k Nucleotide Compositions. *Cbio* 15 (6), 554–562. doi:10.2174/1574893614666190902151038

Zeng, X., Zhu, S., Lu, W., Liu, Z., Huang, J., Zhou, Y., et al. (2020). Target Identification Among Known Drugs by Deep Learning from Heterogeneous Networks. *Chem. Sci.* 11 (7), 1775–1797. doi:10.1039/c9sc04336e

Zhai, Y., Chen, Y., Teng, Z., and Zhao, Y. (2020). Identifying Antioxidant Proteins by Using Amino Acid Composition and Protein-Protein Interactions. *Front. Cel Dev. Biol.* 8, 591487. doi:10.3389/fcell.2020.591487

Zhang, C.-H., Li, M., Lin, Y.-P., and Gao, Q. (2020). Systemic Therapy for Hepatocellular Carcinoma: Advances and Hopes. *Cgt* 20 (2), 84–99. doi:10.2174/1566523220666200628014530

Zhang, L., Xiao, X., and Xu, Z.-C. (2020). iPromoter-5mC: A Novel Fusion Decision Predictor for the Identification of 5-Methylcytosine Sites in Genome-wide DNA Promoters. *Front. Cel Dev. Biol.* 8, 614. doi:10.3389/fcell.2020.00614

Zhao, T., Hu, Y., Peng, J., and Cheng, L. (2020). DeepLGP: a Novel Deep Learning Method for Prioritizing lncRNA Target Genes. *Bioinformatics* 36 (16), 4466–4472. doi:10.1093/bioinformatics/btaa428

Zou, Q., Xie, S., Lin, Z., Wu, M., and Ju, Y. (2016). Finding the Best Classification Threshold in Imbalanced Classification. *Big Data Res.* 5, 2–8. doi:10.1016/j.bdr.2015.12.001

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.