IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON FEATURE REPRESENTATION AND LEARNING METHODS WITH APPLICATIONS IN LARGE-SCALE BIOLOGICAL SEQUENCE ANALYSIS

Received December 2, 2019, accepted December 24, 2019, date of publication December 30, 2019, date of current version January 8, 2020.

Digital Object Identifier 10.1109/ACCESS.2019.2962821

Predicting Sub-Golgi Apparatus Resident Protein With Primary Sequence Hybrid Features

CHUNYU WANG[1], JIALIN LI[1], XIAOYAN LIU[1], AND MAOZU GUO[2,3]

[1]School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China
[2]School of Electrical and Information Engineering, Beijing University of Civil Engineering and Architecture, Beijing 100044, China
[3]Beijing Key Laboratory of Intelligent Processing for Building Big Data, Beijing 100044, China

Corresponding authors: Chunyu Wang (chunyu@hit.edu.cn) and Maozu Guo (guomaozu@bucea.edu.cn)

This work was supported in part by the Natural Science Foundation of China under Grant 61872114, Grant 61571163, Grant 61532014, and Grant 61871020, and in part by the National Key Research and Development Plan Task of China under Grant 2016YFC0901902.

ABSTRACT The Golgi apparatus is a significant membrane-bound organelle of eukaryotic cells that is made up of a series of flattened, stacked pouches (called cisternae). The Golgi apparatus packages proteins into membrane-bound vesicles, and so it is responsible for transporting, modifying, and packaging proteins and lipids into vesicles for delivery to targeted destinations. It belongs to the central organelle mediating system of eukaryotic cells. Functional defects of the Golgi apparatus are associated with many kinds of neurodegenerative diseases, such as Parkinson's and Alzheimer's diseases. Golgi-resident proteins play an important role in the Golgi apparatus' processing, which includes storing, packaging, and dispatching proteins. Identifying sub-Golgi protein types can help researchers to develop more effective therapies and drugs for diseases that result from disorders of Golgi-resident proteins. In this paper, we propose a computational model to discriminate cis-Golgi proteins from trans-Golgi proteins using a machine learning method. First, we use PseKNC, K-separated Bigrams, and PsePSSM as feature extraction techniques, and then we select the optimal features among those identified by PseKNC with the AdaBoost classifier. To create a balanced dataset out of the imbalanced set of Golgi proteins, we used the Random-SMOTE oversampling approach. Finally, we employed the SVM algorithm to distinguish cis-Golgi proteins from trans-Golgi proteins. The proposed method achieves promising performance, with accuracy of 96.5%, 96.5%, and 96.9% in the experiments with jackknife cross-validation, independent testing, and 10-fold cross-validation, respectively, which exceeds the performance of previous related work.

INDEX TERMS Golgi apparatus, feature extraction, hybrid sequence features, protein classification, SVM.

I. INTRODUCTION

The Golgi apparatus, also known as the Golgi complex or Golgi body, is the central organelle that mediates protein and lipid transport within eukaryotic cells [1]. It is located very near the rough endoplasmic reticulum (ER) and hence very near the nucleus. The number of Golgi apparatus bodies within a single eukaryotic cell varies. Typical animal cells may have fewer and larger Golgi apparatus units, while plant cells may contain as many as hundreds of smaller ones. The Golgi apparatus receives proteins and lipids from the rough ER and then modifies, sorts, concentrates, and packs them into sealed droplets called vesicles before sending them out to the cytoplasm. The Golgi apparatus is composed of a series of compartments called cisternae, which are fused and flattened membrane-enclosed disks. A single Golgi apparatus can be roughly divided into two parts: the cis-Golgi network (CGN) and the trans-Golgi network (TGN). Proteins enter the Golgi on the side facing the ER (cis side) and exit on the opposite side of the stack, facing the plasma membrane of the cell (trans side). Cargo proteins processed by the Golgi apparatus must make their way through the network of intervening cisternae, and along the way, they become modified and packaged for transport to various locations within the cell. Both the CGN and TGN have variable structures, including both cisterna-like and vesiculated regions. Each cisterna or region contains different Golgi-resident protein modification enzymes to help the above process, namely cis-Golgi proteins and trans-Golgi proteins. These Golgi-resident proteins play important roles in the Golgi apparatus' processing, which includes storing, packaging, and dispatching cargo proteins. Existing studies have shown that the biological

The associate editor coordinating the review of this manuscript and approving it for publication was Dariusz Mrozek.

function of the Golgi apparatus is closely related to many diseases, such as diabetes [2], cancer [3], Parkinson's disease [4], and Alzheimer's disease [5]. Accurately identifying the types of sub-Golgi proteins could help researchers to understand the contribution of Golgi dysfunction to diseases and develop more effective therapies and drugs for these diseases.

Unfortunately, identifying Golgi-resident protein types by traditional experiments is very costly and time-consuming [6]. It is relatively sensitive to the stability of the experimental environment, equipment, and procedure, which results in poor replicability. With the rapid development of bioinformatics and machine learning techniques, emerging algorithms and models have been devised for protein sequence classification problems. Machine learning-based computational models have been used for many subcellular and sub-subcellular localization problems involving proteins [7]–[13]. Over the past few years, several models for predicting sub-Golgi protein types have been constructed [14]. Ding proposed a method that combines a special mode of pseudo amino acid composition with the Modified Mahalanobis discriminant to identify protein types and obtained an accuracy of 74.7% using jackknife cross-validation [15], [16]. Later, Van Dijk et al. built a model to predict the location of type-ii membrane proteins by using amino acid grouping, string-based triads, and 3D structure-based triads as feature representations in an SVM classifier [17]. Ding et al. continued to improve their previous work by using an ANOVA to filter 2-gap dipeptide features with an accuracy of 85.4% in jackknife cross-validation, and an online server was also deployed for researchers [18]. Jiao et al. proposed a novel protein sequence representation method, namely position specific physico-chemical properties (PSPCP), which integrates Position Specific Scoring Matrix (PSSM) information with artificially created physicochemical property values. They used ANOVA for feature selection and SVM with an RBF kernel for prediction and obtained an accuracy of 86.9% [19]. Further, Jiao used minimum redundancy maximum relevance (mRMR) as the feature selection algorithm with the same feature extraction technique and improved the accuracy to 91% [20]. Later, Yang et al. collected an updated dataset containing 304 sub-Golgi proteins from UniProt. They used this newly constructed dataset as a training set and continued to use the 64 sub-Golgi proteins as an independent test set. They extracted CSP-based features and g-gap dipeptide composition to capture protein sequence characteristics. To balance the datasets, the SMOTE algorithm was adopted and random forest-recursive feature elimination was employed to search for optimal features. Their model achieved accuracy of 88.5%, 93.8%, and 90.1% with jackknife cross-validation, independent testing, and 10-fold cross-validation, respectively [21]. Ahmad et al. used split Amino acid compositions and a bigram positional-specific scoring matrix as the feature extraction method and K-Nearest Neighbor (KNN) as the learner to exceed previous methods with an accuracy of 94.9%, 94.8%, 94.9% in jackknife cross-validation, independent testing, and 10-fold cross-validation, respectively [22]. Rahman et al. adopted both position-specific feature extraction such as n-grams and n-gapped dipeptides and position-independent feature extraction. An RF filter and an SVM wrapper were applied to select the optimal feature subsets. This is the state-of-art method has so far achieved accuracy of 95.4%, 95.5%, and 95.3% for 10-fold cross-validation, jackknife testing, and independent testing, respectively [23].

However, we think there is still room to improve. To develop a useful statistical predictor for a protein classification problem, people often obey Chou's 5-step rule [24]–[31]: (1) Collect protein sequences to construct a benchmark dataset. (2) Extract features from natural protein sequences that can reveal the intrinsic relationships between peptide sequences and targets. (3) Develop a powerful predictor to finish the prediction tasks. (4) Evaluate the predictor using cross-validation tests. (5) Establish a user-friendly web server for the predictor.

The proposed model's construction workflow is shown in Figure 1. We use three feature extraction methods: PseKNC, PsePSSM, and k-separated bigrams. The AdaBoost classifier was employed to select features from PseKNC, which is a high-dimensional vector of size 8,420. We concatenated the three feature descriptors and then used Random-SMOTE to address dataset imbalance. Finally, we used the SVM algorithm to distinguish cis-Golgi from trans-Golgi proteins.

II. MATERIALS AND METHODS

A. DATASETS

We used the training and test benchmark datasets from Yang *et al.* [21]. The training set contains 304 protein sequences, among which 217 are trans-Golgi and 87 are cis-Golgi proteins, while the test set (collected by Ding) contains 13 cis-Golgi proteins and 51 trans-Golgi proteins (total: 64 sequences) [18].

None of the protein sequences in the training set have more than 40% pairwise identity with any other protein sequence in the training dataset, and none of the protein sequences in the test dataset have more than 25% pairwise identity with any other protein sequence in the test dataset. This is because a redundancy cutoff was performed on them to avoid homology bias and redundancy.

Both the training set and the independent test set were extracted according to the following criteria [18]:

1. Only proteins annotated as cis-Golgi or trans-Golgi are selected.

2. Only proteins with experimentally verified annotations are included. Proteins annotated with 'PROBABLE,' 'PO-TENTIAL,' or 'BY SIMILARITY' are excluded.

3. Protein sequences with ambiguous amino acid notations (X, B, or Z) are discarded, as are fragments of other proteins.

4. The sequence identity level should be lower than a CD-HIT threshold, such as 40% or 25%.

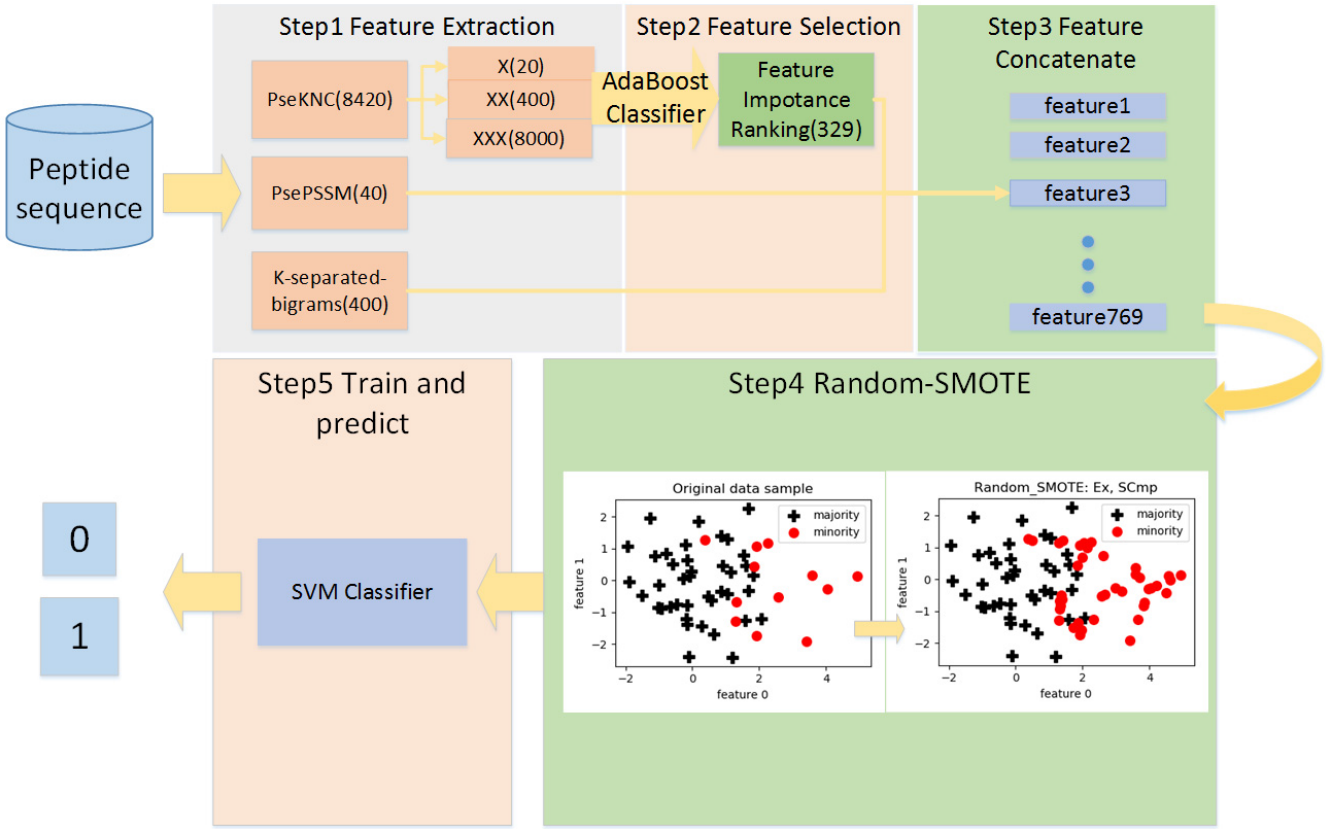


FIGURE 1. Framework of the proposed model. In step 1 we use three extraction techniques PseKNC, PsePSSM, k-separated-bigrams. And the character 'X' represents one type of 20 amino acids while 'XX' represents one type of 400 dipeptides and so forth. In step 2, Adaboost classifier was used to select optimal features among PseKNC according to their importanceranking and eventually we got 329 features. In step 3 we simply concatenate these three sets of features and obtained a 769-dimension feature vector. In step 4, we used Random-SMOTE to balance the datasets. Finally, in step 5, we used the SVM algorithm to distinguish the cis-Golgi proteins from trans-Golgi proteins.

B. FEATURE EXTRACTION METHODS

Generally, one key step in protein prediction is to convert the protein sequence into an effective mathematical expression using a reasonable formula [32]. In sequence-based problems, the method of extracting features from proteins' primary sequences is significant. One of the primary steps in the development of a powerful computational model is conversion of the protein sequences to a set of numerical features that intrinsically reveal the sequences' characteristics [33]. The process of feature exaction directly influences the model's precision. In general, a combination of different feature descriptors contains as much information about a protein sequence as possible because different descriptors can complement information that the others do not have [34]. In this paper, we use three feature extraction methods: PseKNC, PsePSSM, and k-separated bigrams.

1) PseKNC FEATURES

The PseKNC is a novel nucleotide sequence representation that have been applied to predict the attributes of DNA sequences [35]. It can also be applied to protein sequences [36]. A protein sequence P can be represented as below:

$$P = R_1 R_2 R_3 \ldots R_L \tag{1}$$

where R_1 represents the amino acid at chain position 1, R_2 represents the amino acid at chain position 2, and so forth. The k-tuple nucleotide composition is a vector with 4^k components that represents a DNA sequence.

$$D = \left[f_1^{K-\text{tuple}}, f_2^{K-\text{tuple}}, \ldots, f_i^{K-\text{tuple}}, \ldots, f_{4k}^{K-\text{tuple}} \right]^T \tag{2}$$

In equation (2), $f_i^{K-\text{tuple}}$ is the normalized occurrence of the i-th k-tuple nucleotide in the DNA sequence. When $k = n$, $\Sigma_{i=1}^{n} 4^i$, and $\Sigma_{i=1}^{n} 20^i$ features will be generated for DNA and protein, respectively. The PseKNC we used was that from PyFeature [37]. For example, when $k = 3$, the feature structure will be X, XX, and XXX, where X represents one type of amino acid. It produces an 8,420-dimensional vector composed of the number of occurrences of single peptide, di-peptides, and tri-peptides in the protein sequence.

2) PsePSSM FEATURES

The position specific scoring matrix (PSSM) can be used to describe evolutionary information about a protein sequence. Evolutionary conservation can reflect import biological functions [38], [39]. The PSSM can be generated from the PSI-BLAST by searching for homogenous sequences to each query protein in the Swiss-Prot database for three iterations

with 0.01 as the E-value cutoff [40]. The PSSM of sequence P is represented by equation (3).

$$P_{\mathrm{PSSM}} = \begin{bmatrix} E_{1\to1} & E_{1\to2} & \cdots & E_{1\to20} \\ E_{2\to1} & E_{2\to2} & \cdots & E_{2\to20} \\ \vdots & \vdots & \ddots & \vdots \\ E_{L\to1} & E_{L\to2} & \cdots & E_{L\to20} \end{bmatrix} \tag{3}$$

where in equation (3) $E_{i\to j}$ represents the score of the amino acid residue at the i-th position of the protein sequence being changed to amino acid residue type j during the evolutionary process, where L is the length of the query sequence and numerical codes 1, 2, ..., 20 represent the 20 native amino acid residues in alphabetical order. Because $E_{i\to j}$ varies in a wide range, the following standardization is performed:

$$E_{i\to j} = \frac{E^0_{i\to j} - \frac{1}{20}\Sigma^{20}_{k=1}E^0_{i\to k}}{\sqrt{\frac{1}{20}\Sigma^{20}_{u=1}\left(E^0_{i\to j} - \frac{1}{20}\Sigma^{20}_{k=1}E^0_{i\to k}\right)^2}} \tag{4}$$

where $i = 1, 2, \ldots, L$, $j = 1, 2, \ldots, 20$, and $E^0_{i\to j}$ represents the original scores calculated by PSI-BLAST [41]. However, proteins with different lengths generate different matrices, which cannot be handled by prediction models. Thus, to obtain a uniform dimensional matrix, the following transformation is performed:

$$\overline{P_{\mathrm{PSSM}}} = \left[\overline{\mathrm{E}_1}, \overline{\mathrm{E}_2}, \ldots, \overline{\mathrm{E}_{20}}\right]^T \tag{5}$$

$$\overline{E}_j = \frac{1}{L}\sum_{i=1}^{L} E_{i\to j} \tag{6}$$

where $\bar{E}_j (j = 1, 2, \ldots, 20)$ represents the average score of the j-th type of amino acid in protein P during evolution. However, this would lead to the loss of information about sequence order. To avoid this, and prompted by the creation of PseAAC, the PsePSSM was proposed by Shen and Chou [42]. The PseAAC is widely used and has been popular since it was introduced because it not only calculates amino acid frequency but also considers the long-range correlations of physicochemical properties between two residues along the sequence, avoiding the complete loss of sequence-order information [43]–[45]. This pseudo position-specific scoring matrix can be depicted as shown below:

$$P^{\xi}_{\mathrm{Pse-PSSM}} = \left[\overline{\mathrm{E}_1}, \overline{\mathrm{E}_2}, \ldots, \overline{\mathrm{E}_{20}}, G^{\xi}_1, G^{\xi}_2, \ldots, G^{\xi}_{20}\right]^T \tag{7}$$

$$G^{\xi}_j = \frac{1}{L-\xi}\sum_{i=1}^{L-\xi}\left[E_{i\to j} - E_{(i+\xi)\to j}\right]^2 \tag{8}$$

3) k-SEPARATED BIGRAMS

The k-separated bigrams were extracted from PSSM proposed by Saini *et al.* [46]. It describes the relationships between non-adjacent amino acids along the protein sequence. The two amino acids are separated by k amino acids and the bigram probabilities are extracted from the sequential evolution probabilities in PSSM. The algorithm can be described mathematically as below:

$$T_{m,n}(k) = \sum_{i=1}^{L-K} N_{i,m}N_{i+k,n}, \tag{9}$$

$$T(k) = \left[T_{1,1}(k), \ldots, T_{1,20}(k), T_{2,1}(k), \ldots, T_{20,20}(k)\right]. \tag{10}$$

where $1 \le m \le 20$, $1 \le n \le 20$, $k \le K$, and N is the PSSM matrix representation for a protein sequence, and it has L rows and 20 columns as equation (3) and L is the length of the protein sequence. The k represents the distance between the amino acid positions which are used to compute the transition probabilities, for $k = 1$, the amino acids used to calculate the transition probabilities are separated by 0 amino acid. For $k = 2$, the amino acids used to calculate the transition probabilities are separated by 1 amino acid and so forth. For each k, $T(k)$ generates 400 features.

C. FEATURES SELECTION

We combine different assortments of feature descriptors as much as possible to avoid losing sequential information. However, there may be overlap among different features and noise, which can result in over-fitting [47]. To reduce the impact of redundancy and decrease time and space complexity, we use the AdaBoost classifier implemented in the scikit-learn package in Python to find the optimal feature sets with its default the parameter settings (learning rate=1, C4.5 as the default base learner, and 500 as the number of base learners) to find the optimal feature sets. AdaBoost is an ensemble boosting classifier that was proposed by Freund and Schapire [48]. It is an iterative ensemble method that works by combining multiple poorly performing classifiers. The basic concept behind AdaBoost is to set the classifiers' weights for each iteration according to the accuracy of the classifier, and it assigns greater weights to incorrectly classified observations so that these observations will have a high probability of classification in the next iteration. It calculates the average impurity curtailment achieved by splitting based on each of the features in each tree trained on different sample weight distributions. Every feature is given an importance score using the scikit-learn package's feature importance function. More important features have higher scores. In this paper, we select 329 features whose importance is larger than 0.

D. RANDOM-SMOTE

The main concept of SMOTE is to create new minority class samples by interpolating between several minority class examples that lie close together [49]–[51]. Specifically, for every minority sample, its k nearest neighbors from the minority class are first selected. Then, according to the oversampling rate, N neighbors should be chosen randomly from the k nearest neighbors. Finally, synthetic examples P_j are

created in the following way:

$$P_j = x + \text{rand}\,(0, 1) * \left(y_j - x\right). \tag{11}$$

where $y_j (j = 1, 2, \ldots, N)$ is one of the k randomly selected nearest neighbors of x, and rand(0,1) generates a random number between 0 and 1.

SMOTE creates a new sample along the line between the minority class sample and the selected nearest neighbors. After SMOTE, the dataset maintains its intensive or sparse characteristics, thus leading to poor performance when the sample is located in the sparsely populated space. To solve the problem, Random-SMOTE was proposed by Dong and Wang [52]. In Random-SMOTE, two examples (y_1 and y_2) are randomly selected from the minority class. Consequently, a triangle is formed by the sample (x), y_1 and y_2. Then, based on the oversampling rate N, several examples are created randomly within the triangular area.

The detailed procedure for generating synthetic examples is depicted below:

1. Generate a temporary example v_1 on the line between the two selected minority examples y_1 and y_2

$$v_1 = y_1 + \text{rand}\,(0, 1) * (y_2 - y_1); \tag{12}$$

2. Generate synthetic minority class examples p_j $(j = 1, 2, \ldots, N)$ on the line between sample x and the temporary sample v_j

$$P_j = x + \text{rand}\,(0, 1) * \left(v_j - x\right). \tag{13}$$

However, there are three cases for the relative locations among x and the two temporary examples y_1 and y_2.

1. When the three points' locations coincide, a copy of x is obtained; This degenerates to random oversampling.
2. When two of the three points coincide, it is the same as SMOTE.
3. When none of the points coincide with each other, synthetic samples are generated in the triangular area (this is the usual case).

In conclusion, Random-SMOTE is a more general method. Random oversampling or SMOTE is a special case of Random-SMOTE.

E. PREDICTION ALGORITHM

Support Vector Machine (SVM) [53] has been successfully applied in protein sequence classification projects [54], [55], in which a decision boundary that maximizes the margin between positive and negative samples is found. [56]–[62]. The basic idea of SVM is to map the original data into a higher-dimensional feature space using a kernel function, and then perform classification in this feature space by finding the optimal separating hyperplane. We used grid search to find the optimal parameters, and eventually, the best parameter combination of kernel='linear', C=0.01, and gamma=0.01 was found by 10-fold cross-validation.

F. PERFORMANCE EVALUATION

It is an important step to choose good performance metrics to measure whether the model works well. In this paper, we use accuracy, sensitivity, specificity, and Matthew's correlation coefficients, which are calculated using a confusion matrix obtained according to true and predicted classes [61], [63]–[71]. The chosen metrics are defined as below:

$$Sn = \frac{TP}{TP + FN} \tag{14}$$

$$Sp = \frac{TP}{TN + FP} \tag{15}$$

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{16}$$

$$MCC = \frac{TP \times TN - FN \times FP}{\sqrt{(TP+FP)\,(TP+FN)\,(TN+FP)\,(TN+FN)}} \tag{17}$$

We categorized our dataset into two classes: the positive class and negative class. TP are defined as the positive samples that are classified as positive instances, TN are the negative samples that are categorized as negative, FP are the negative samples that are categorized as positive, and FN are the positive samples that are categorized as negative. Sn measures the true positive rate, while Sp measures the true negative rates, and these values are equally important for evaluating the model. ACC reflects the predictor's overall accuracy, but when the dataset is imbalanced, ACC may be misleading [72], while MCC will still be informative to measure the model's overall quality. MCC ranges -1 to 1, where -1 represents that the predictor always predicts the wrong result, 0 indicates a random guess, and 1 denotes that the predictor predicts all samples accurately. Thus, MCC can be seen as a correlation coefficient between the true and predicted classes.

III. EVALUATION METRICS

There are many methods to evaluate the performance of a computational model. Three cross-validation methods are widely used in statistical prediction [28], [73]–[80]. These are the jackknife cross-validation, 10-fold cross-validation, and independent testing [81]. In this paper, we also use those methods to evaluate our model.

In the jackknife test (also called leave-one-out cross-validation), each protein sequence in the training set is held out as an independent test sample. This is the most objective and rigorous method, and it can yield unbiased results with small variance values [82], but this method takes more time to run because its execution time is equal to the number of samples.

In 10-fold cross-validation, the dataset is divided into 10 parts, each of which are used for both training and testing (9 are used for training and the other for testing during each iteration). The average accuracy of the 10 results is seen as an estimate of the algorithm's overall accuracy. Usually, 10-fold cross-validation is applied multiple times.

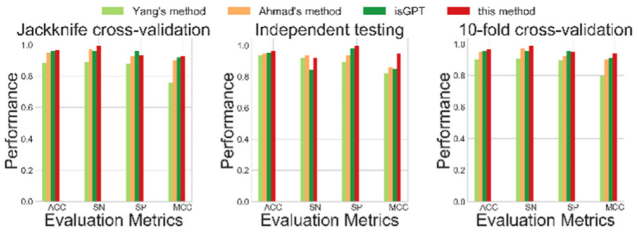


FIGURE 2. Comparison histogram between the proposed model and three previous models in three different evaluation metrics.

FIGURE 3. Comparison histogram between models without Random-SMOTE and with Random-SMOTE in three different evaluation metrics.

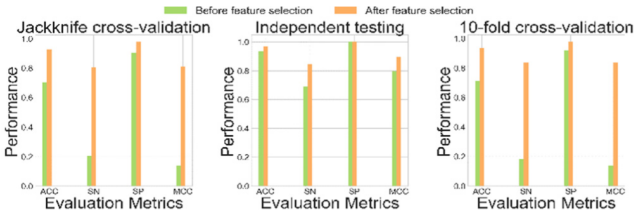


FIGURE 4. Comparison histogram between models before feature selection and after feature selection in three different evaluation metrics.

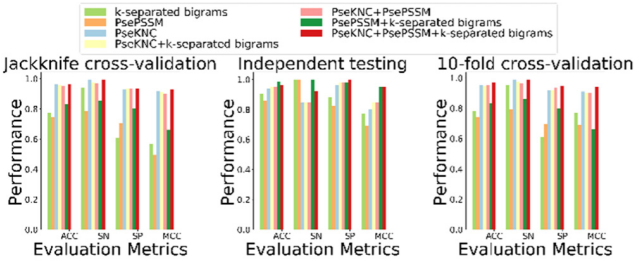


FIGURE 5. Comparison histogram between models with different feature combinations in three different evaluation metrics.

Independent testing, also known as holdout testing, in which there is no overlap between the training and test sets. In other words, the training set is completely different from the testing set. When the testing set is divided from the training set, the distribution of the testing set should be similar to that of the training set; otherwise, the results of this testing strategy may be misleading [83]. In this paper, we used two different datasets collected independently: one for training and the other for testing.

IV. RESULTS AND DISCUSSIONS

A. COMPARISON WITH EXISTING METHODS

To determine whether our method is more powerful or at least comparable to state-of-the-art method, we performed a jackknife cross-validation, independent testing, 10-fold cross-validation on the same datasets. The results are showed in Figure 2. Our method gives the highest overall accuracy and MCC using Jackknife cross-validation, independent testing, and 10-fold cross-validation. Our accuracy was 96.5%, 96.5%, and 96.9% on the jackknife cross-validation, independent testing, and 10-fold cross-validation, respectively, and the MCC values were 0.93, 0.95, and 0.94, respectively. Therefore, we conclude that the proposed method is a powerful classifier of sub-Golgi proteins.

B. EFFECTS OF RANDOM-SMOTE

To investigate the effectiveness of Random-SMOTE, we show the experimental results of the models with and without Random-SMOTE in Figure 3. We use MCC to evaluate their performance because it is more informative than ACC when the data are imbalanced, as in this experiment. The results verify the effectiveness of Random-SMOTE, with MCC values of 0.93, 0.95, and 0.94 for the three types of validation, respectively.

C. EFFECTS OF FEATURE SELECTION ON PseKNC

Figure 4 shows the experimental results before and after feature selection for Jackknife cross-validation, independent testing, and 10-fold cross-validation. Before feature selection, our accuracy was 70.3%, 93.7% and 71.1% and the MCC values was 0.14, 0.80 and 0.14 for the three types of validation, respectively. After feature selection, our accuracy was 92.7%,96.8% and 93.7% and the MCC values was 0.81,0.90 and 0.84 for the three types of validation, respectively. The results indicate that the model that uses the AdaBoost classifier after feature selection is much more effective because it has removed some redundant and irrelevant information.

D. EFFECTS OF DIFFERENT FEATURE COMBINATIONS

An analysis of different hybrid features is presented in Figure 5. This set of experiments is performed to select the best combination of different feature descriptors. These hybrid feature sets are PseKNC, k-separated-bigrams, PsePSSM, PseKNC + k-separated-bigrams, PseKNC + PsePSSM, k-separated-bigrams + PsePSSM and PseKNC + PsePSSM + k-separated-bigrams. For the parameters in the feature extraction technique of k-separated-bigrams and PsePSSM, we set $k = 1$ and $\xi = 1$ respectively. The best hybrid feature set is PseKNC + PsePSSM + k-separated-bigrams, with accuracy of 96.5%, 96.5%, and 96.9% in jackknife cross-validation, independent testing and 10-fold cross-validation, respectively. The best hybrid feature set contains 769 features, among which 329, 40, and 400 were from PseKNC, PsePSSM, and k-separated-bigrams, respectively.

V. CONCLUSION

In this paper, we developed a robust and powerful computational model for classification of sub-Golgi proteins. In this model, we extracted PseKNC, k-separated-bigrams, and PsePSSM to represent protein sequences. The Adaboost classifier was used to remove the redundant information contained in the PseKNC descriptor, and the reduced PseKNC features achieved a higher score than the full PseKNC. Comparative experiments showed that the combination of PseKNC, k-separated-bigrams, and Pse-PSSM was the most effective combination. The random-SMOTE technique was adopted to balance the datasets, and the prediction performance of Random-SMOTE based models is much better than that of those models that did not use Random-SMOTE. Finally, we used SVM as our predictor. By comparing our method with previous work, we conclude that our method is much more powerful, with accuracy of 96.5%, 96.5%, and 96.9% in jackknife cross-validation, independent testing, and 10-fold cross-validation, respectively.

AUTHOR CONTRIBUTIONS

C. Wang initiated the idea, conceived the whole process and drafted the manuscript. J. Li implemented the experiments, designed the figures and also drafted the manuscript. X. Liu helped with data analysis and revised the manuscript. M. Guo finalized the article. All authors have read and approved the final manuscript.

ACKNOWLEDGMENT

The funders had no role in study design, data collection and analysis decision to publish, or preparation of the manuscript. The authors would like to thank R. Lipkin, Ph.D., from Liwen Bianji (www.liwenbianji.cn/ac), Edanz Group China, for editing the English text of a draft of this manuscript.

REFERENCES

[1] Y. Wang, "Golgi apparatus inheritance," in *The Golgi Apparatus: State of the Art 110 Years After Camillo Golgi's Discovery*, A. A. Mironov and M. Pavelka, Eds. Vienna, Austria: Springer, 2008, pp. 580–607, doi: 10.1007/978-3-211-76310-0_34.

[2] S. Hoyer, "Is sporadic Alzheimer disease the brain type of non-insulin dependent diabetes mellitus? A challenging hypothesis," *J. Neural Transmiss.*, vol. 105, no. 4, pp. 415–422, 1998.

[3] D. R. Rose, "Structure, mechanism and inhibition of Golgi α-mannosidase II," *Current Opinion Struct. Biol.*, vol. 22, no. 5, pp. 558–562, Oct. 2012.

[4] Y. Fujita, E. Ohama, M. Takatama, S. Al-Sarraj, and K. Okamoto, "Fragmentation of Golgi apparatus of nigral neurons with α-synuclein-positive inclusions in patients with Parkinson's disease," *Acta Neuropatholo.*, vol. 112, no. 3, pp. 261–265, Sep. 2006.

[5] N. K. Gonatas, J. O. Gonatas, and A. Stieber, "The involvement of the Golgi apparatus in the pathogenesis of amyotrophic lateral sclerosis, Alzheimer's disease, and ricin intoxication," *Histochem. Cell Biol.*, vol. 109, nos. 5–6, pp. 591–600, Jun. 1998.

[6] W. Yang, X.-J. Zhu, J. Huang, H. Ding, and H. Lin, "A brief survey of machine learning methods in protein sub–Golgi localization," *Current Bioinf.*, vol. 14, no. 3, pp. 234–240, Mar. 2019.

[7] P.-F. Du, "Predicting protein submitochondrial locations: The 10th anniversary," *Current Genomics*, vol. 18, no. 4, pp. 316–321, Jul. 2017.

[8] B. Manavalan, S. Basith, T. H. Shin, L. Wei, and G. Lee, "MAHTPred: A sequence-based meta-predictor for improving the prediction of anti-hypertensive peptides using effective feature representation," *Bioinformatics*, vol. 35, no. 16, pp. 2757–2765, Aug. 2019.

[9] W. Qiu, S. Li, X. Cui, Z. Yu, M. Wang, J. Du, Y. Peng, and B. Yu, "Predicting protein submitochondrial locations by incorporating the pseudo-position specific scoring matrix into the general Chou's pseudo-amino acid composition," *J. Theor. Biol.*, vol. 450, pp. 86–103, Aug. 2018.

[10] K. Ahmad, M. Waris, and M. Hayat, "Prediction of protein submitochondrial locations by incorporating dipeptide composition into Chou's general pseudo amino acid composition," *J. Membrane Biol.*, vol. 249, no. 3, pp. 293–304, Jun. 2016.

[11] Y. Shen, J. Tang, and F. Guo, "Identification of protein subcellular localization via integrating evolutionary and physicochemical information into Chou's general PseAAC," *J. Theor. Biol.*, vol. 462, pp. 230–239, Feb. 2019.

[12] L. Yu, S. Yao, L. Gao, and Y. Zha, "Conserved disease modules extracted from multilayer heterogeneous disease and gene networks for understanding disease mechanisms and predicting disease treatments," *Front. Genet.*, vol. 9, p. 745, Jan. 2019.

[13] L. Yu, J. Zhao, and L. Gao, "Predicting potential drugs for breast cancer based on miRNA and tissue specificity," *Int. J. Biol. Sci.*, vol. 14, no. 8, pp. 971–982, 2018.

[14] Z. Lv, S. Jin, H. Ding, and Q. Zou, "A random forest sub–Golgi protein classifier optimized via dipeptide and amino acid composition features," *Front. Bioeng. Biotechnol.*, vol. 7, p. 215, Sep. 2019.

[15] Y. Ding, J. Tang, and F. Guo, "Identification of drug-target interactions via multiple information integration," *Inf. Sci.*, vols. 418–419, pp. 546–560, Dec. 2017.

[16] H. Ding, L. Liu, F.-B. Guo, J. Huang, and H. Lin, "Identify Golgi protein types with modified mahalanobis discriminant algorithm and pseudo amino acid composition," *Protein Peptide Lett.*, vol. 18, no. 1, pp. 58–63, Jan. 2011.

[17] A. D. J. Van Dijk, D. Bosch, C. J. F. T. Braak, A. R. van der Krol, and R. C. H. J. van Ham, "Predicting sub-Golgi localization of type II membrane proteins," *Bioinformatics*, vol. 24, no. 16, pp. 1779–1786, Aug. 2008.

[18] H. Ding, S.-H. Guo, E.-Z. Deng, L.-F. Yuan, F.-B. Guo, J. Huang, N. Rao, W. Chen, and H. Lin, "Prediction of Golgi-resident protein types by using feature selection technique," *Chemometrics Intell. Lab. Syst.*, vol. 124, pp. 9–13, May 2013.

[19] Y.-S. Jiao and P.-F. Du, "Predicting Golgi-resident protein types using pseudo amino acid compositions: Approaches with positional specific physicochemical properties," *J. Theor. Biol.*, vol. 391, pp. 35–42, Feb. 2016.

[20] Y.-S. Jiao and P.-F. Du, "Prediction of Golgi-resident protein types using general form of Chou's pseudo-amino acid compositions: Approaches with minimal redundancy maximal relevance feature selection," *J. Theor. Biol.*, vol. 402, pp. 38–44, Aug. 2016.

[21] R. Yang, C. Zhang, R. Gao, and L. Zhang, "A novel feature extraction method with feature selection to identify Golgi–resident protein types from imbalanced data," *Int. J. Mol. Sci.*, vol. 17, no. 2, p. 218, Feb. 2016.

[22] J. Ahmad, F. Javed, and M. Hayat, "Intelligent computational model for classification of sub-Golgi protein using oversampling and Fisher feature selection methods," *Artif. Intell. Med.*, vol. 78, pp. 14–22, May 2017.

[23] M. S. Rahman, M. K. Rahman, M. Kaykobad, and M. S. Rahman, "IsGPT: An optimized model to identify sub-Golgi protein types using SVM and Random Forest based feature selection," *Artif. Intell. Med.*, vol. 84, pp. 90–100, Jan. 2018.

[24] J. Jia, Z. Liu, X. Xiao, B. Liu, and K.-C. Chou, "IPPBS-Opt: A sequence–based ensemble classifier for identifying protein–protein binding sites by optimizing imbalanced training datasets," *Molecules*, vol. 21, no. 1, p. 95, Jan. 2016.

[25] J. Jia, Z. Liu, X. Xiao, B. Liu, and K.-C. Chou, "IPPI-Esml: An ensemble classifier for identifying the interactions of proteins by incorporating their physicochemical properties and wavelet transforms into PseAAC," *J. Theor. Biol.*, vol. 377, pp. 47–56, Jul. 2015.

[26] B. Liu, L. Fang, S. Wang, X. Wang, H. Li, and K.-C. Chou, "Identification of microRNA precursor with the degenerate K-tuple or Kmer strategy," *J. Theor. Biol.*, vol. 385, pp. 153–159, Nov. 2015.

[27] B. Liu, R. Long, and K.-C. Chou, "IDHS-EL: Identifying DNase I hypersensitive sites by fusing three different modes of pseudo nucleotide composition into an ensemble learning framework," *Bioinformatics*, vol. 32, no. 16, pp. 2411–2418, Aug. 2016.

[28] H. Ding, E.-Z. Deng, L.-F. Yuan, L. Liu, H. Lin, W. Chen, and K.-C. Chou, "ICTX-type: A sequence–based predictor for identifying the types of conotoxins in targeting ion channels," *Biomed. Res. Int.*, vol. 2014, pp. 1–10, May 2014.

[29] L. Yu, J. Zhao, and L. Gao, "Drug repositioning based on triangularly balanced structure for tissue-specific diseases in incomplete interactome," *Artif. Intell. Med.*, vol. 77, pp. 53–63, Mar. 2017.

[30] B. Liu, X. Gao, and H. Zhang, "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.*, vol. 47, no. 20, p. e127, Nov. 2019.

[31] W. Chen, P. Feng, T. Liu, and D. Jin, "Recent advances in machine learning methods for predicting heat shock proteins," *Current Drug Metabolism*, vol. 20, no. 3, pp. 224–228, May 2019.

[32] J.-X. Tan, S.-H. Li, Z.-M. Zhang, C.-X. Chen, W. Chen, H. Tang, and H. Lin, "Identification of hormone binding proteins based on machine learning methods," *Math. Biosci. Eng.*, vol. 16, no. 4, pp. 2466–2480, 2019.

[33] H. Ding and D. Li, "Identification of mitochondrial proteins of malaria parasite using analysis of variance," *Amino Acids*, vol. 47, no. 2, pp. 329–333, Feb. 2015.

[34] X.-J. Zhu, C.-Q. Feng, H.-Y. Lai, W. Chen, and L. Hao, "Predicting protein structural classes for low-similarity sequences by evaluating different features," *Knowl.-Based Syst.*, vol. 163, pp. 787–793, Jan. 2019.

[35] L. Cheng, J. Li, Y. Hu, Y. Jiang, Y. Liu, Y. Chu, Z. Wang, and Y. Wang, "Using semantic association to extend and infer literature–oriented relativity between terms," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 12, no. 6, pp. 1219–1226, Nov. 2015.

[36] H. Tang, Y.-W. Zhao, P. Zou, C.-M. Zhang, R. Chen, P. Huang, and H. Lin, "HBPred: A tool to identify growth hormone-binding proteins," *Int. J. Biol. Sci.*, vol. 14, no. 8, pp. 957–964, 2018.

[37] R. Muhammod, S. Ahmed, D. Md Farid, S. Shatabda, A. Sharma, and A. Dehzangi, "PyFeat: A Python-based effective feature generation tool for DNA, RNA and protein sequences," *Bioinformatics*, vol. 35, no. 19, pp. 3831–3833, Oct. 2019.

[38] B. Liu and Y. Zhu, "ProtDec–LTR3.0: Protein remote homology detection by incorporating profile–based features into learning to rank," *IEEE Access*, vol. 7, pp. 102499–102507, 2019.

[39] L. Wei, Q. Zou, M. Liao, H. Lu, and Y. Zhao, "A novel machine learning method for cytokine-receptor interaction prediction," *Combinat. Chem. High Throughput Screening*, vol. 19, no. 2, pp. 144–152, Jan. 2016.

[40] A. A. Schäffer, "Improving the accuracy of PSI–BLAST protein database searches with composition-based statistics and other refinements," *Nucleic Acids Res.*, vol. 29, no. 14, pp. 2994–3005, Jul. 2001.

[41] S. Altschul, "Gapped BLAST and PSI–BLAST: A new generation of protein database search programs," *Nucleic Acids Res.*, vol. 25, no. 17, pp. 3389–3402, Sep. 1997.

[42] H.-B. Shen and K.-C. Chou, "Nuc–PLoc: A new Web-server for predicting protein subnuclear localization by fusing PseAA composition and PsePSSM," *Protein Eng. Des. Selection*, vol. 20, no. 11, pp. 561–567, Nov. 2007.

[43] B. Yu, S. Li, W.-Y. Qiu, C. Chen, R.-X. Chen, L. Wang, M.-H. Wang, and Y. Zhang, "Accurate prediction of subcellular location of apoptosis proteins combining Chou’s PseAAC and PsePSSM based on wavelet denoising," *Oncotarget*, vol. 8, no. 64, Dec. 2017, Art. no. 107640.

[44] M. Hayat and A. Khan, "Discriminating outer membrane proteins with fuzzy K–nearest neighbor algorithms based on the general form of Chou's PseAAC," *Protein Peptide Lett.*, vol. 19, no. 4, pp. 411–421, Apr. 2012.

[45] B. Liu, "BioSeq–Analysis: A platform for DNA, RNA and protein sequence analysis based on machine learning approaches," *Briefings Bioinf.*, vol. 20, no. 4, pp. 1280–1294, Jul. 2019, doi: 10.1093/bib/bbx165.

[46] H. Saini, G. Raicar, S. Lal, A. Dehzangi, J. Lyons, K. K. Paliwal, S. Imoto, S. Miyano, and A. Sharma, "Genetic algorithm for an optimized weighted voting scheme incorporating k-separated bigram transition probabilities to improve protein fold recognition," in *Proc. Asia–Pacific World Congr. Comput. Sci. Eng.*, Nov. 2014, pp. 1–7.

[47] H. Tang, W. Chen, and H. Lin, "Identification of immunoglobulins using Chou's pseudo amino acid composition with feature selection technique," *Mol. Biosyst.*, vol. 12, pp. 1269–1275, Apr. 2016.

[48] Y. Freund and R. E. Schapire, "A decision–theoretic generalization of on–line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, Aug. 1997.

[49] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, Jul. 2018.

[50] S. Wan, Y. Duan, and Q. Zou, "HPSLPred: An ensemble multi–label classifier for human protein subcellular location prediction with imbalanced source," *Proteomics*, vol. 17, nos. 17–18, Sep. 2017, Art. no. 1700262.

[51] L. Chao, L. Wei, and Q. Zou, "SecProMTB: A SVM-based classifier for secretory proteins of mycobacterium tuberculosis with imbalanced data set," *Proteomics*, vol. 19, Sep. 2019, Art. no. e1900007.

[52] Y. Dong and X. Wang, "A new over-sampling approach: Random-SMOTE for learning from imbalanced data sets," in *Knowledge Science, Engineering and Management* (Lecture Notes in Computer Science). 2011, pp. 343–352.

[53] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[54] L. Wei, C. Zhou, H. Chen, J. Song, and R. Su, "ACPred–FL: A sequence-based predictor using effective feature representation to improve the prediction of anti-cancer peptides," *Bioinformatics*, vol. 34, no. 23, pp. 4007–4016, 2018.

[55] H. Yang, H. Tang, X. X. Chen, C. J. Zhang, P. P. Zhu, and H. Ding, "Identification of secretory proteins in mycobacterium tuberculosis using pseudo amino acid composition," *Biomed. Res. Int.*, vol. 2016, Jul. 2016, Art. no. 5413903.

[56] M. Bhasin and G. P. S. Raghava, "ESLpred: SVM-based method for subcellular localization of eukaryotic proteins using dipeptide composition and PSI–BLAST," *Nucleic Acids Res.*, vol. 32, pp. W414–W419, Jul. 2004.

[57] S. Wan, M.-W. Mak, and S.-Y. Kung, "GOASVM: A subcellular location predictor by incorporating term-frequency gene ontology into the general form of Chou's pseudo-amino acid composition," *J. Theor. Biol.*, vol. 323, pp. 40–48, Apr. 2013.

[58] C. Jia, Y. Zuo, and Q. Zou, "O-GlcNAcPRED-II: An integrated classification algorithm for identifying O–GlcNAcylation sites based on fuzzy undersampling and a K-means PCA oversampling technique," *Bioinformatics*, vol. 34, no. 12, pp. 2029–2036, Jun. 2018.

[59] B. Liu, C. C. Li, and K. Yan, "DeepSVM-fold: Protein fold recognition by combining support vector machines and pairwise sequence similarity scores generated by deep learning networks," *Briefings Bioinf.*, 2019, Art. no. bbz098, doi: 10.1093/bib/bbz098.

[60] X. Zhang, Q. Zou, A. Rodriguez-Paton, and X. Zeng, "Meta–path methods for prioritizing candidate disease miRNAs," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 16, no. 1, pp. 283–291, Jan. 2019.

[61] X. Zeng, Y. Liao, Y. Liu, and Q. Zou, "Prediction and validation of disease genes using HeteSim scores," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 14, no. 3, pp. 687–695, May 2017.

[62] Q. Zou, J. Li, L. Song, X. Zeng, and G. Wang, "Similarity computation strategies in the microRNA-disease network: A survey," *Briefings Funct. Genomics*, vol. 15, no. 1, pp. 55–64, Jul. 2015.

[63] X. Zeng, W. Lin, M. Guo, and Q. Zou, "A comprehensive overview and evaluation of circular RNA detection tools," *PLoS Comput. Biol.*, vol. 13, no. 6, Jun. 2017, Art. no. e1005420.

[64] L. Wei, P. Xing, R. Su, G. Shi, Z. S. Ma, and Q. Zou, "CPPred–RF: A sequence-based predictor for identifying cell–penetrating peptides and their uptake efficiency," *J. Proteome Res.*, vol. 16, no. 5, pp. 2044–2053, May 2017.

[65] L. Wei, P. Xing, J. Zeng, J. Chen, R. Su, and F. Guo, "Improved prediction of protein–protein interactions using novel negative samples, features, and an ensemble classifier," *Artif. Intell. Med.*, vol. 83, pp. 67–74, Nov. 2017.

[66] L. Cheng, C. Qi, H. Zhuang, T. Fu, and X. Zhang, "gutMDisorder: A comprehensive database for dysbiosis of the gut microbiota in disorders and interventions," *Nucleic Acids Res.*, 2019, Art. no. gkz843, doi: 10.1093/nar/gkz843.

[67] Y. Hu, T. Zhao, N. Zhang, T. Zang, J. Zhang, and L. Cheng, "Identifying diseases-related metabolites using random walk," *BMC Bioinf.*, vol. 19, no. S5, p. 116, Apr. 2018.

[68] M. Zhang, F. Li, T. T. Marquez-Lago, A. Leier, C. Fan, C. K. Kwoh, K.-C. Chou, J. Song, and C. Jia, "MULTiPly: A novel multi-layer predictor for discovering general and specific types of promoters," *Bioinformatics*, vol. 35, no. 17, pp. 2957–2965, Sep. 2019.

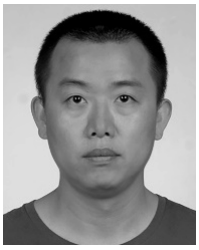
[69] T. Song, A. Rodriguez-Paton, P. Zheng, and X. Zeng, "Spiking neural P systems with colored spikes," *IEEE Trans. Cogn. Devel. Syst.*, vol. 10, no. 4, pp. 1106–1115, Dec. 2018.

[70] X. Lin, Z. Quan, Z.-J. Wang, H. Huang, and X. Zeng, "A novel molecular representation with BiGRU neural networks for learning atom," *Briefings Bioinf.*, 2019, Art. no. bbz125, doi: 10.1093/bib/bbz125.

[71] X. Zeng, W. Wang, C. Chen, and G. G. Yen, "A consensus community-based particle swarm optimization for dynamic community detection," *IEEE Trans. Cybern.*, to be published, doi: 10.1109/tcyb.2019.2938895.

[72] Y. Jiao and P. Du, "Performance measures in evaluating machine learning based bioinformatics predictors for classifications," *Quant. Biol.*, vol. 4, no. 4, pp. 320–330, Dec. 2016.

[73] M. Hayat, M. Tahir, and S. A. Khan, "Prediction of protein structure classes using hybrid space of multi-profile Bayes and bi-gram probability feature spaces," *J. Theor. Biol.*, vol. 346, pp. 8–15, Apr. 2014.
[74] L. Wei, H. Chen, and R. Su, "M6APred-EL: A sequence–based predictor for identifying N6-methyladenosine sites using ensemble learning," *Mol. Therapy-Nucleic Acids*, vol. 12, pp. 635–644, Sep. 2018.
[75] L. Cheng, Y. Jiang, H. Ju, J. Sun, J. Peng, M. Zhou, and Y. Hu, "InfAcrOnt: Calculating cross-ontology term similarities using information flow by a random walk," *BMC Genomics*, vol. 19, no. S1, p. 919, Jan. 2018.
[76] L. Cheng, Y. Hu, J. Sun, M. Zhou, and Q. Jiang, "DincRNA: A comprehensive Web-based bioinformatics toolkit for exploring disease associations and ncRNA function," *Bioinformatics*, vol. 34, no. 11, pp. 1953–1956, Jun. 2018.
[77] N. Dyubankova, E. Sochacka, K. Kraszewska, B. Nawrot, P. Herdewijn, and E. Lescrinier, "Contribution of dihydrouridine in folding of the D-arm in tRNA," *Organic Biomol. Chem.*, vol. 13, no. 17, pp. 4960–4966, Mar. 2015.
[78] B. Liu, Y. Zhu, and K. Yan, "Fold-LTR-TCP: Protein fold recognition based on triadic closure principle," *Briefings Bioinf.*, 2019, Art. no. bbz139, doi: 10.1093/bib/bbz139.
[79] Y. Liu, X. Zeng, Z. He, and Q. Zou, "Inferring MicroRNA–disease associations by random walk on a heterogeneous network with multiple data sources," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 14, no. 4, pp. 905–915, Jul./Aug. 2017.
[80] X. Chen, M. J. Pérez-Jiménez, L. Valencia-Cabrera, B. Wang, and X. Zeng, "Computing with viruses," *Theor. Comput. Sci.*, vol. 623, pp. 146–159, Apr. 2016.
[81] P.-M. Feng, H. Ding, W. Chen, and H. Lin, "Naïve Bayes classifier with feature selection to identify phage virion proteins," *Comput. Math. Methods Med.*, vol. 2013, Apr. 2013, Art. no. 530696.
[82] B. Efron, "Bootstrap methods: Another look at the jackknife," in *Breakthroughs in statistics*. New York, NY, USA: Springer, 1992, pp. 569–593.
[83] R. Kohavi, D. Sommerfield, and J. Dougherty, "Data mining using a machine learning library in C++," *Int. J. Artif. Intell. Tools*, vol. 6, no. 4, pp. 537–566, 1997.



CHUNYU WANG received the B.S., M.S., and Ph.D. degrees in computer science and technology from the Harbin Institute of Technology. From 2016 to 2017, he was with the Department of Electrical Engineering and Computer Science, University of Missouri, as a Visiting Scholar. He is currently an Associate Professor with the Department of Computer Science and Technology. His research interests include bioinformatics and machine learning.

JIALIN LI received the B.S. degree in computer science and technology from the Harbin Institute of Technology, Weihai. She is currently pursuing the M.S. degree with the School of Computer Science and Technology, Harbin Institute of Technology. Her main field of research is machine learning and computational biology.

XIAOYAN LIU received the B.S., M.S., and Ph.D. degrees from the Harbin Institute of Technology. She is currently an Associate Professor with the Harbin Institute of Technology. Her research interests include computational biology, engineering database, and knowledge-based systems.

MAOZU GUO received the Ph.D. degree. He is currently a Professor and the Dean of the School of Electrical and Information Engineering, Beijing University of Civil Engineering and Architecture. His research fields mainly include machine learning and data mining, computational biology and bioinformatics, and image understanding. He is currently one of the guiding panel members in some Major Research Plan of National Natural Science Foundation Committee (NSFC), a member of the Artificial Intelligence and Pattern Recognition Society of China Computer Federation (CCF) and the Systems Biology Society of Operations Research Society of China (ORSC), and a Standing Committee Member of the Machine Learning Society of Chinese Association for Artificial Intelligence (CAAI).