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 June 28, 2020, accepted July 13, 2020, date of publication July 21, 2020, date of current version August 5, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3010944

A Pearson Based Feature Compressing Model for SNARE Protein Classification

GUILIN LI
Department of Software Engineering, School of Informatics, Xiamen University, Xiamen 361005, China
e-mail: glli@xmu.edu.cn

ABSTRACT SNARE proteins are a group of proteins that drive the biological fusion of two membranes. It is important to identify them accurately, because malfunction of the SNARE proteins can lead to a lot of diseases. In this paper, a Pearson based feature compressing model is proposed to identify the SNARE proteins accurately and efficiently. First, 188D, CKSAAP, CTDD and CTRIAD feature extraction methods are used to extract features from the SNARE and non-SNARE proteins. As the number of features extracted by the four methods is very large, which means many redundant features are included. It is necessary to filter the original feature set. The Chi-Square, Information Gain and Pearson Correlation Coefficient feature selection methods are used to evaluate the value of each feature in the feature set. The selected features are used to train a random forest classifier and the performance of the selected features is evaluated by cross validation. The experimental results showed that the CTDD based model with the first 70% of features selected by the Pearson feature selection method can achieve the best performance among all kinds of models.

INDEX TERMS
Feature representation, feature selection, protein classification.

I. INTRODUCTION

SNARE proteins are a group of proteins that drive the biological fusion of two membranes. Some research shows that many diseases are associated with the malfunction of the SNARE protein, which means the SNARE proteins are essential to human health, they have attracted many researchers to study it [1]–[12]. And it is necessary to develop some techniques to identify them. The identification of SNARE proteins can be carried out in two ways. The first one is by means of bioinformatics techniques, which are expensive and time-consuming. Considering that the machine learning based methods have been widely used in protein classification [13]–[34], we will use machine learning to recognize SNARE protein in this paper. Traditionally, features are extracted from the SNARE proteins. Then some kinds of machine learning algorithms are trained based on the features. After adjusting the parameters for the machine learning algorithm, a model can be constructed. But the problem is whether the features extracted are all necessary for the protein identification problem? As we all know, most feature extraction methods extract hundreds or even thousands of features from a protein, which can introduce many redundant features. This kind of features brings two problems. First, they increase the complexity of the training algorithm. Second, they can reduce the accuracy of the classifier. To overcome the two problems, the feature compressing or selection methods [13]–[30], [35]–[44] are used to filter the redundant features out.

In this paper, we test the performance of several feature compressing methods to identify the SNARE proteins accurately and efficiently. Four feature extraction methods, which are the 188D [45], the CKSAAP [43], [44], [46], the Composition/Transition/Distribution (CTDD) [47], [48] and the CTRIAD methods [49], are used to extract features from the proteins [50]. As the number of features extracted by the four methods are very large, three kinds of feature compressing methods (Chi-square, Information Gain and Pearson) are used to compress the feature set just extracted. The Chi-square method orders the value of each feature by measuring the correlation degree between the feature and the category. Information gain method orders the value of each feature by calculating how much information the feature can bring to the classification system. Pearson method calculates the degree of coefficient for the feature and category. The feature set is compressed to 10%, 25%, 40%, 55%, 70%, 85% and 100% of their original size by the three feature

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

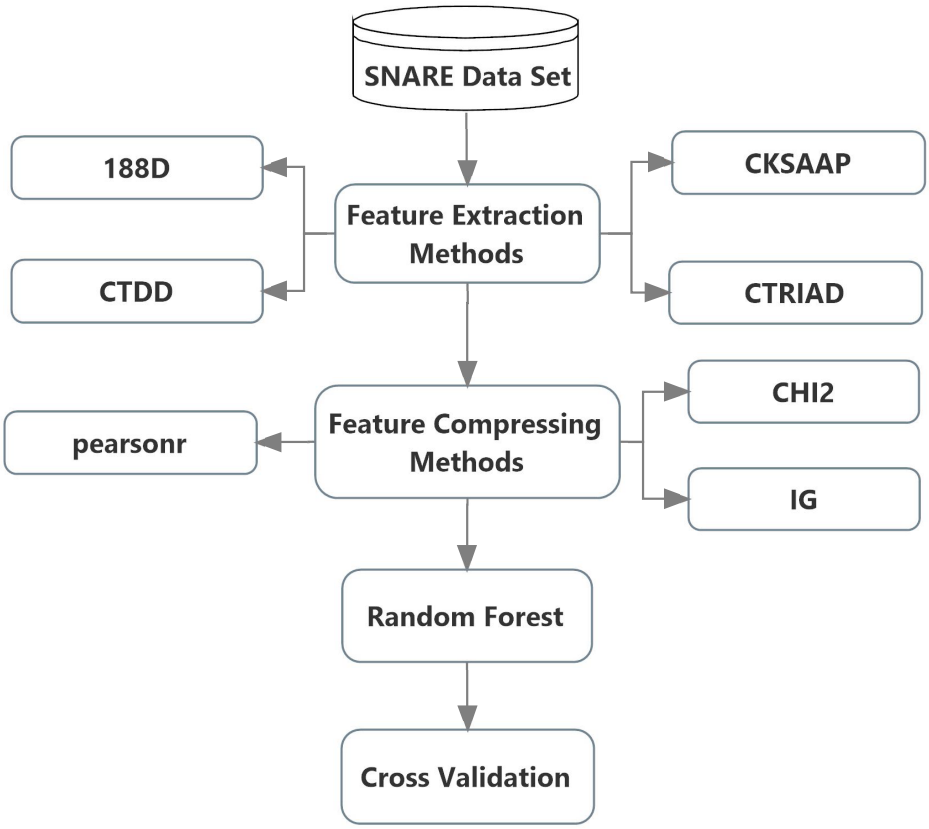


FIGURE 1. Framework of the feature compressing model for the SNARE protein identification.

compressing methods. Finally, the compressed feature set is used to train a random forest classifier. Experiments show that the performance of the Pearson based on the CTDD feature extraction method can achieve the best performance among all models.

The contributions of this work include (1) Three kinds of feature selection methods are applied to four kinds of feature sets extracted by four feature extraction methods from the SNARE proteins. The optimal compressing feature set for each kind of feature extraction method has been found. (2) By comparing the performance of all the optimal compressed feature set, a Pearson based feature compressing model is proposed to identify the SNARE proteins accurately and efficiently.

The rest of the paper is organized as follows. In section 2, we briefly introduce the feature extraction methods, feature compressing methods and machine learning algorithm used in this paper. The experiments are given in Section 3. Finally, we draw the conclusion in Section 4.

II. METHODS

The framework of feature compressing model for the SNARE protein identification is shown in figure 1. The framework is composed of four steps. In the first step, feature extraction methods are used to extract different kinds of features from the dataset, which is composed of SNARE and non-SNARE proteins. Four kinds of feature selection methods are used, which are the CTDD, CKSAAP, 188D and CTRIAD. The 188D method extracts 188 features from the dataset. The feature set for the CKSAAP is composed of 2400 features. The CTDD and CTRIAD extracts 195 and 343 features respectively. It can be seen that the number of features is very large, which contains lots of redundant features. In the second step, several feature compressing or selection methods are used to select the most valuable features from the feature set extracted by each feature extraction method. The feature compressing methods used are the Chi-square (CHI2), Information Gain (IG) and Pearson Correlation Coefficient (Pearson) methods, which are used to select a part of features from the whole feature set, called compressed feature set. In the third step, the compressed feature set is used to train a classifier, which is based on the random forest algorithm. Finally, the performance of each compressed feature set is evaluated by the cross validation method and the best model will be selected.

A. DATASET

First, we downloaded the SNARE data from the UniProt database [51]–[53] as the positive instances. We choose the vesicular transport proteins as the negative instances, because

the structure of vesicular transport proteins are similar to the SNARE proteins and downloaded them. If our classifier can distinguish the SNARE protein from the vesicular transport proteins accurately, it must be a precise classifier. Finally, the dataset with positive and negative instances is used to train and cross validate the feature compressing models.

B. FEATURE EXTRACTION METHODS

1) 188D

188D feature extraction method extracts 188 features from each protein. These features can be further divided into two categories: the first category is the statistical characteristics of amino acids that make up proteins. Because there are altogether 20 kinds of amino acids, the first 20 features belong to the first category and calculated as follows.

Let $FV_1, \ldots, FV_{20}$ denote the first 20 features (1–20):

$$FV_i = \frac{a_i}{L} \quad (i = 1, \ldots, 20)$$

where a_i is the number of the 20 amino acids in the protein sequence and L is its length.

The second category is the statistical characteristics of proteins according to their physicochemical properties, which are mainly from eight kinds of sources, such as hydrophobicity, surface tension, normalized Van der Waals volume etc.. 21 features are calculated based on each physicochemical property to form the remaining 168 features.

2) CKSAAP

CKSAAP features are extracted based on the order of the AACs appearing in the sequence. It computes at most 2400 features from a protein. As we know, a protein is composed of 20 kinds of AACs. There are 400 possible combinations to form a pair of amino acids. Given a protein sequence, the CKSAAP first checks the combination of any two consecutive ACCs in the sequence and counts the number for each combination appearing in the sequence. By dividing the number of each combination with the number of total consecutive pairs of AAC in the sequence, the frequency of the pair of AAC in the sequence are computed, as is shown in the following formula, which is a vector of 400 dimension.

$$\left(\frac{N_{AA}}{N_{total}}, \frac{N_{AC}}{N_{total}}, \frac{N_{AD}}{N_{total}}, \ldots, \frac{N_{YY}}{N_{total}}\right)_{400}$$

where $N_{AA} \ldots N_{YY}$ are the combinations of amino acids in the protein sequence.

Two AACs can be separated by the other k AACs ($k = 1, 2, 3, 4, 5$). In the case of two consecutive AACs, the k equals to 0. For each $k(k = 1, 2, 3, 4, 5)$, 400 features are extracted in the same way as in the case of $k = 0$. The number of features extracted by CKSAAP are $400 \times 6 = 2400$.

3) CTDD

CTDD is a set of features extracted based on the AAC distribution patterns of the physicochemical property in a protein sequence, whose calculation method is as follows: (i) The AACs sequence is converted into a sequence with certain physicochemical or structural property; (ii) Based on the 7 different physicochemical features from the AAC indices of Kanehisa and Tomii, twenty AACs are divided into three groups for each of them.

The Distribution descriptor is composed of five values for each of the three features (neutral, hydrophobic and polar). The descriptor calculates the fraction for an AAC of a given group, where it first located, and where 25, 50, 75 and 100% of occurrences are located at the entire sequence.

For instance, we begin at the first residue and include the residue that marks the occurrence of 25/50/75/100% of any given group of residues, and then we divide the position of the residues by the entire sequence length.

4) CTRIAD

The Conjoint Triad descriptor (CTriad) extracts the 343 features from the proteins. In CTRIAD, all the 20 amino acids are catalogued into 7 classes and three consecutive AACs in a protein sequence are considered as a single unit. The extraction method of CTRIAD is similar to that of the CKSAAP. All the combinations for a unit should be equal to $7 \times 7 \times 7 = 343$. Accordingly, $i = 1, 2, 3, \ldots, 343$. The CTRIAD first calculates the frequency f_i of the ith combination for the protein. In principle, the longer a protein sequence is, the higher probability it has larger values of f_i. Thus, a new parameter d_i is defined, which normalizes f_i by the following formula:

$$d_i = \frac{f_i - \min\{f_1, f_2, \ldots, f_{343}\}}{\max\{f_1, f_2, \ldots, f_{343}\}}$$

C. FEATURE COMPRESSING METHODS

1) CHI-SQUARE FEATURE SELECTION (CHI2)

The basic idea of χ^2 test is to determine whether the hypothesis is correct or not by calculating the deviation between the theoretical value and the actual value. We often assume that the two random variables are indeed independent (called ''original hypothesis''), and then calculate the degree of deviation between the theoretical value and the actual value (called the observation value). If the deviation is small enough, we think that the error is a very natural sample error, which is the measurement If the measurement method is not accurate enough to cause or happen accidentally, the two are indeed independent, then accept the original hypothesis; if the deviation is large enough to a certain extent, so that such error is unlikely to be caused by chance or measurement inaccuracy, we think that the two are actually related, that is, deny the original hypothesis, and accept the alternative hypothesis.

$$\chi^2 = \sum \frac{(A - E)^2}{E}$$

where A is the observed value and E is the expected value.

χ^2 feature selection is a kind of supervised feature selection method. By χ^2 test between the feature and the real category, it can judge the correlation degree between the

feature and the real category, and then determine whether to select the feature or not.

2) INFORMATION GAIN FEATURE SELECTION (IG)

Information gain is a frequently used feature selection method. It is used to measure the amount of information a feature can take to the classification algorithm. The more information a feature takes, the greater the corresponding information gain of the feature will be. For a feature F from the feature set, the information entropy is calculated as:

$$I(F) = -\sum_j P\left(f_j\right) \log_2 \left(P\left(f_j\right)\right)$$

in which f_j represents a set of values of F, and $P(f_i)$ is its prior probability. The conditional entropy of F under the condition of C is defined as:

$$I(F|C) = -\sum_i P\left(c_i\right) \sum_j P\left(f_j|c_i\right) \log_2 \left(P\left(f_j|c_i\right)\right)$$

in which $P\left(f_j|c_i\right)$ denotes the posterior probability of f_j given c_i of C. And the information gain $IG(F|C)$ is calculated by:

$$IG(F|C) = I(F) - I(C)$$

3) PEARSON CORRELATION COEFFICIENT FEATURE SELECTION (PEARSON)

Pearson correlation coefficient, denoted as r, is usually used to measure the degree of correlation between a pair of random variables (P, Q), which is calculated as follows:

$$r = \frac{\sum_i \left(p_i - \bar{p}_i\right)\left(q_i - \bar{q}_i\right)}{\sqrt{\sum_i \left(p_i - \bar{p}_i\right)^2 \sum_i \left(q_i - \bar{q}_i\right)^2}}$$

in which $\bar{p}_i$ and $\bar{q}_i$ are the mean of P and the mean of Q respectively.

The range of r is within the interval from 1 to -1. The greater the absolute value of r, the stronger the correlation between two random variables P and Q.

D. RANDOM FOREST

As is well known, the major shortcoming of the decision tree algorithm is that it is easy to overfit. Random forest algorithm is proposed to overcome the shortcoming of the decision trees. As the name of the algorithm, the random forest algorithm constructs multiple decision trees like a forest. By applying the voting mechanism, the final classification results of random forest is determined by the majority voting mechanism. The bagging strategy is used to train each decision tree in the forest, which means n samples for training a tree is generated from the population by the bootstrapping method.

III. EXPERIMENTS

In this section, five experiments are done to test the performance for different kinds of the feature compressing methods based on the 188D, CKSAAP, CTDD and CTRIAD feature extraction methods. Three kinds of feature compressing methods are used, which are the Cha-square, information gain, the Pearson correlation coefficient based methods. Random forest algorithm is used to identify the SNARE proteins.

SN, SP, ACC, and MCC given by formula (1) to (4) are used to evaluate the performance of different kinds of feature compression methods. 10 fold cross validation methods are used to evaluate the performance of different feature compressing models.

$$SN = \frac{TP}{TP + FN} \tag{1}$$

$$SP = \frac{TN}{TN + FP} \tag{2}$$

$$ACC = \frac{TN + TP}{TN + FP + TP + FN} \tag{3}$$

$$MCC = \frac{1 - \left(\frac{FN}{TP+FN} + \frac{FP}{TN+FP}\right)}{\sqrt{\left(1 + \frac{FP-FN}{TP+FN}\right)\left(1 + \frac{FN-FP}{TN+FP}\right)}} \tag{4}$$

in which TP denotes True Positive, FP denotes False Positive, TN denotes True Negative and FN denotes False Negative.

Weka [54] is used to do the experiments. The parameters are listed in table 1.

TABLE 1. Parameter list.

Parameters for Random Forest	Value
bagSizePercent	100
batchSize	100
maxDepth	0
numDecimalPlaces	2
numFeatures	0
numIterations	100

A. PERFORMANCE OF DIFFERENT KINDS OF COMPRESSING METHODS FOR THE 188D

In this section, the 188D method is used to extract features from the SNARE proteins. The feature set extracted is composed of 188 features. Then, three kinds of compressing methods, which are the CHI2, IG and Pearson, are used to compress the 188D feature set. The number of features in the compressed feature set is the 10%, 25%, 40%, 55%, 70%, 85% and 100% of that in the original feature set. For example, there are 188 features in the original feature set. After CHI2 method is used to compress the feature set into 10% of the original feature set, only 19 features are left in the compressed feature set. Then the data set are filtered based on the compressed feature set, which means only the data for the compressed feature set are left. The filtered data are used to train and cross validate the random forest machine learning algorithm. In the same way, different kinds of compressing algorithms are imposed on the data set for different kinds of feature extraction methods.

The performance for different kinds of compressing methods for the 188D is shown in Figure 2. The performance comparison for SN is shown in Figure 2a. It shows that, for IG and Pearson method, the best SN is achieved when 85% of features are selected from the 188d feature set, which is composed of 160 features. While for the CHI2 method, the

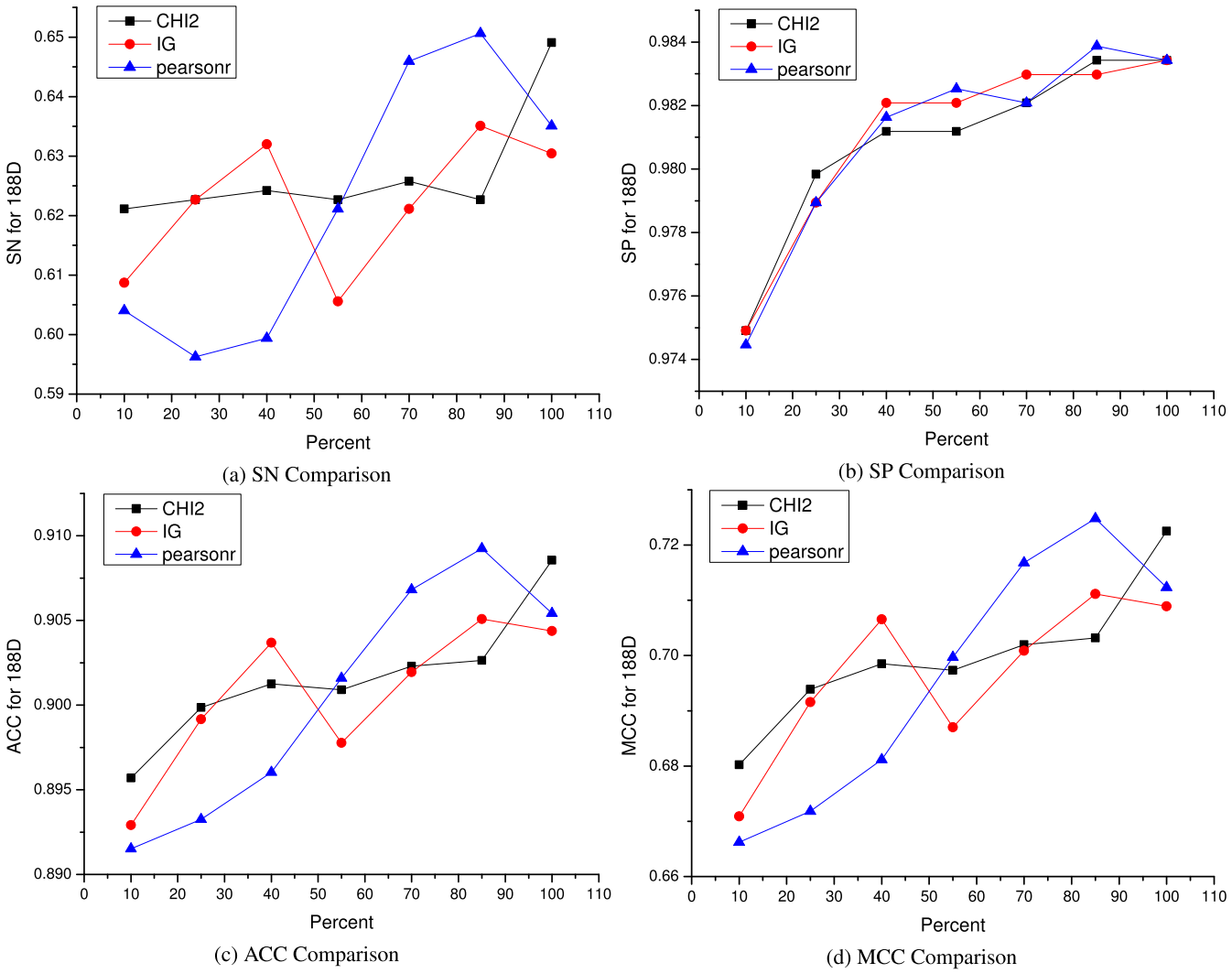


FIGURE 2. Performance comparison of different feature compressing methods for 188D.

best performance for SN is achieved when 188 features are all used. By comparing the best performance of SN achieved by the three comparison methods, the Pearson method wins, which means the best SN is achieved by the personr feature compressing method when 85% features are selected. As the number of selected features increases, the performance of SP becomes better and better. It can be seen from the figure that the best SP value is obtained by the Pearson compression method when 85% of the features are selected. The performance of different compression methods on ACC and MCC is shown in the figure. Similarly, as the number of selected features increases, the performance of ACC and MCC gradually improves. But for the IG and Pearson compression methods, the best performance is also obtained when 85% of the features are selected, rather than when 188 features are used for classification. The experimental results show that the Pearson method is the best compressing method for 188d feature extraction method when 160 features are selected.

B. PERFORMANCE OF DIFFERENT KINDS OF COMPRESSING METHODS FOR THE CKSAAP

Figure 3 shows the performance comparison results of various feature compressing methods on the CKSAAP feature extraction method, and the random forest algorithm is used to classify the SNARE proteins based on the compressed feature set. Figure 3a is the comparison result of SN index of various compression algorithms. As shown in the figure, as the number of compression features increases, the performance of SN continues to deteriorate. This is mainly because the feature set of CKSAAP consists of 2400 features. When the first 10% of features are selected as classification features, the feature set already includes 240 features. As the number of features in the compressed feature set increases, more and more Many features not related to classification are added to the compressed feature set, which results in poor classification performance of the random forest classifier. Furthermore, as can be seen from Figure 3a, the SN index IG

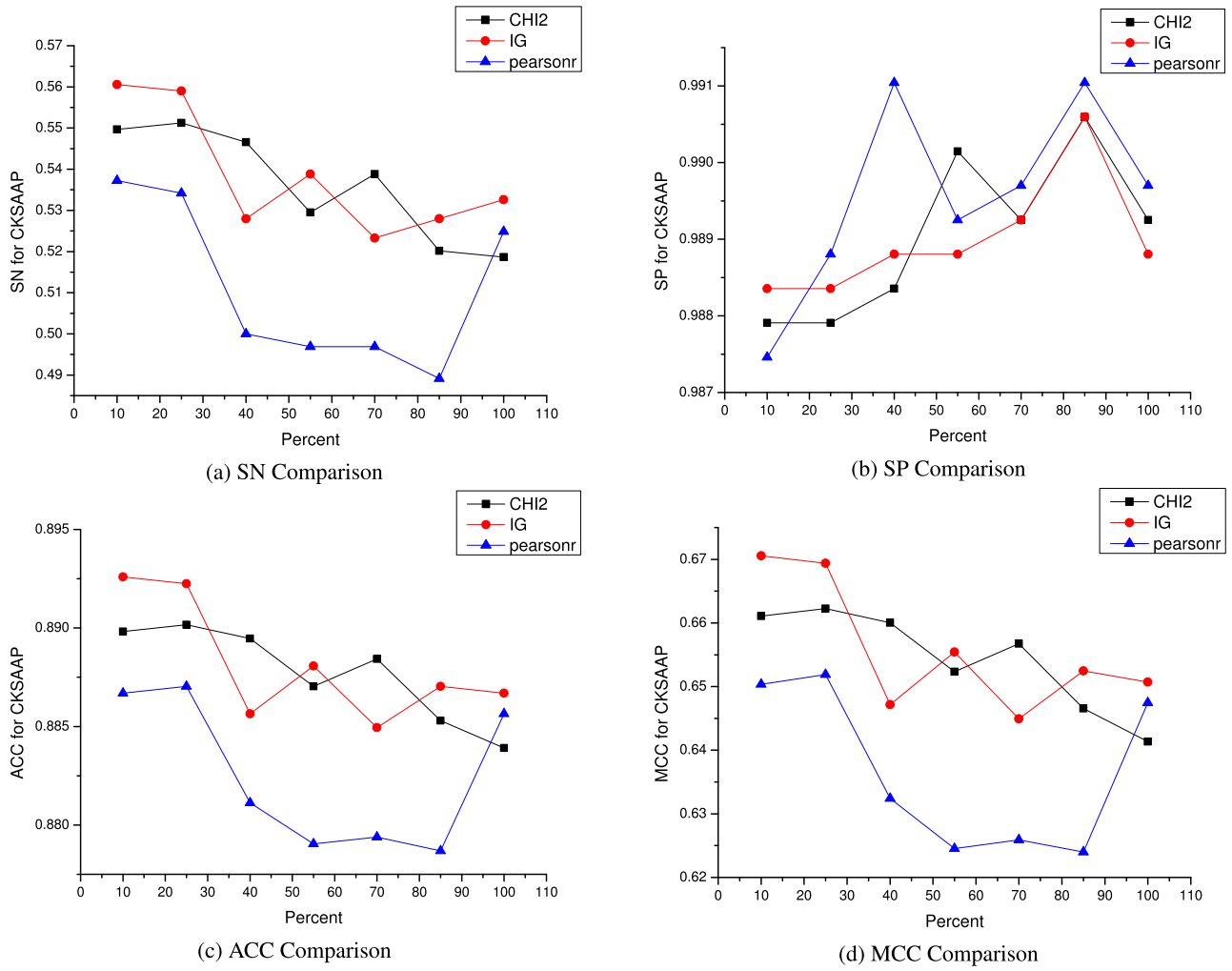


FIGURE 3. Performance comparison of different feature compressing methods for CKSAAP.

compression method has the best performance when selecting the top 10% of features from the CKSAAP feature set as the classification feature set.

Figure 3b is the SP index comparison results of various compression algorithms. For SP indicators, the Pearson feature compression method has the best performance. The best SP value is when using the Pearson method to select the top 40% of features from the CKSAAP feature set as classification features. Figure 3c and Figure 3d show the comparison results of ACC and MCC indicators of various compression algorithms. The results of these two indicators are similar to the situation of SN. As the number of features in the compressed feature set increases, the performance of the ACC and MCC indicators deteriorates. The best performance is achieved when 10% of the features are selected as classification features. In summary, for the CKSAAP feature collection method, the IG feature compression method can obtain the best feature compression effect, and the best effect is achieved when the first 10% of the features are selected.

C. PERFORMANCE OF DIFFERENT KINDS OF COMPRESSING METHODS FOR THE CTDD

Figure 4 shows the performance comparison results of various feature compression methods on the CTDD feature extraction method, using the random forest algorithm to classify SNARE proteins based on the compressed feature set. Figure 4a is the comparison result of various compression algorithm SN indexes. As shown in the figure, when the IG and CHI2 feature compression methods are used to compress the CTDD feature set, as the number of features in the compressed feature set increases, the performance of the SN gradually improves, so these two methods cannot perform the CTDD feature set compression. When using the Pearson method to compress the CTDD feature set, as the number of compressed features increases, the performance of the SN gradually improves. When the number of features in the compressed feature set reaches 70% of the total number of CTDD features, the SN reaches the maximum value. Subsequently, as the number of features in the compressed feature set increases, the performance of SN gradually deteriorates.

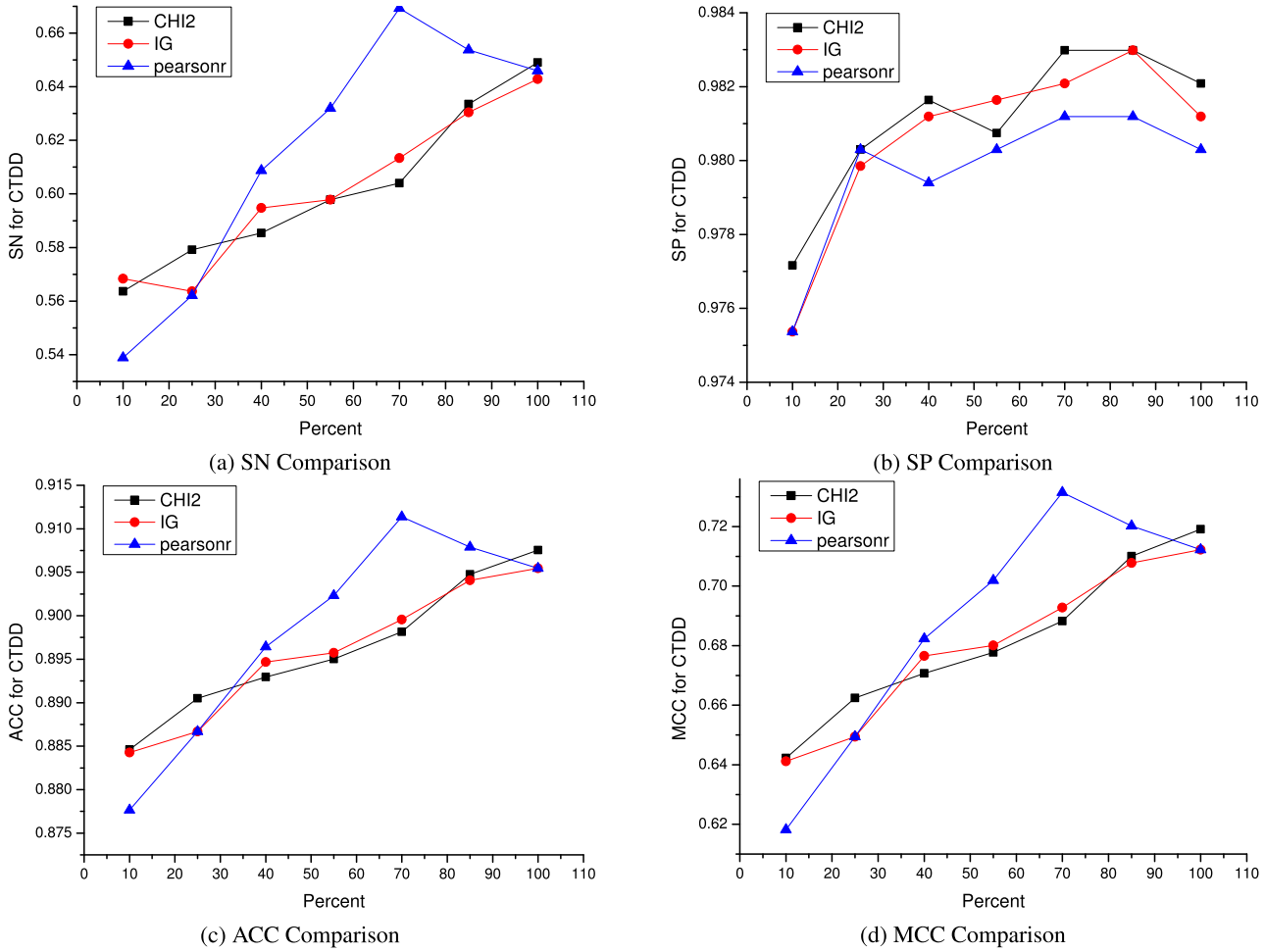


FIGURE 4. Performance comparison of different feature compressing methods for CTDD.

It can be seen that the Pearson compression method has a feature compression effect on the CTDD feature set. The CTDD feature set is composed of 195 features. When the Pearson method takes the top 70% of CTDD features as a compressed feature set for classification (contains 137 features), SN obtains the maximum value. When the number of features in the compressed feature set is small, the classifier does not have enough information to classify, so the SN value is low. As the number of features in the compressed feature set increases, the performance of the classifier gradually improves. When the number of features reaches 137, the best is achieved. Subsequently, more and more irrelevant features are added to the compressed feature set, which leads to the performance degradation of the classifier. As can be seen from Figure 4a, the SN index Pearson compression method has the best performance when selecting the top 70% of features from the CTDD feature set as the classification feature set.

Figure 4b is the SP index comparison results of various compression algorithms. For SP indicators, the CHI2 feature compression method has the best performance. The best SP value is when using the CHI2 method to select the top 70% of features from the CTDD feature set as classification features. Figure 4c and Figure 4d are the comparison results of ACC and MCC indicators of various compression algorithms. The results of these two indicators are similar to the situation of SN. The IG and CHI2 methods have no compression effect on the CTDD feature set. With the increase in the number of features in the compressed feature set of the Pearson method, the performance of the ACC and MCC indicators is good first and then poor. The best performance is achieved when 70% of the features are selected as classification features. In summary, for the CTDD feature collection method, the Pearson feature compression method can obtain the best feature compression effect, and the best effect is achieved when the first 70% of the features are selected.

D. PERFORMANCE OF DIFFERENT KINDS OF COMPRESSING METHODS FOR THE CTRIAD

Figure 5 shows the performance comparison results of various feature compression methods on the CTRIAD feature extraction method, using the random forest algorithm to

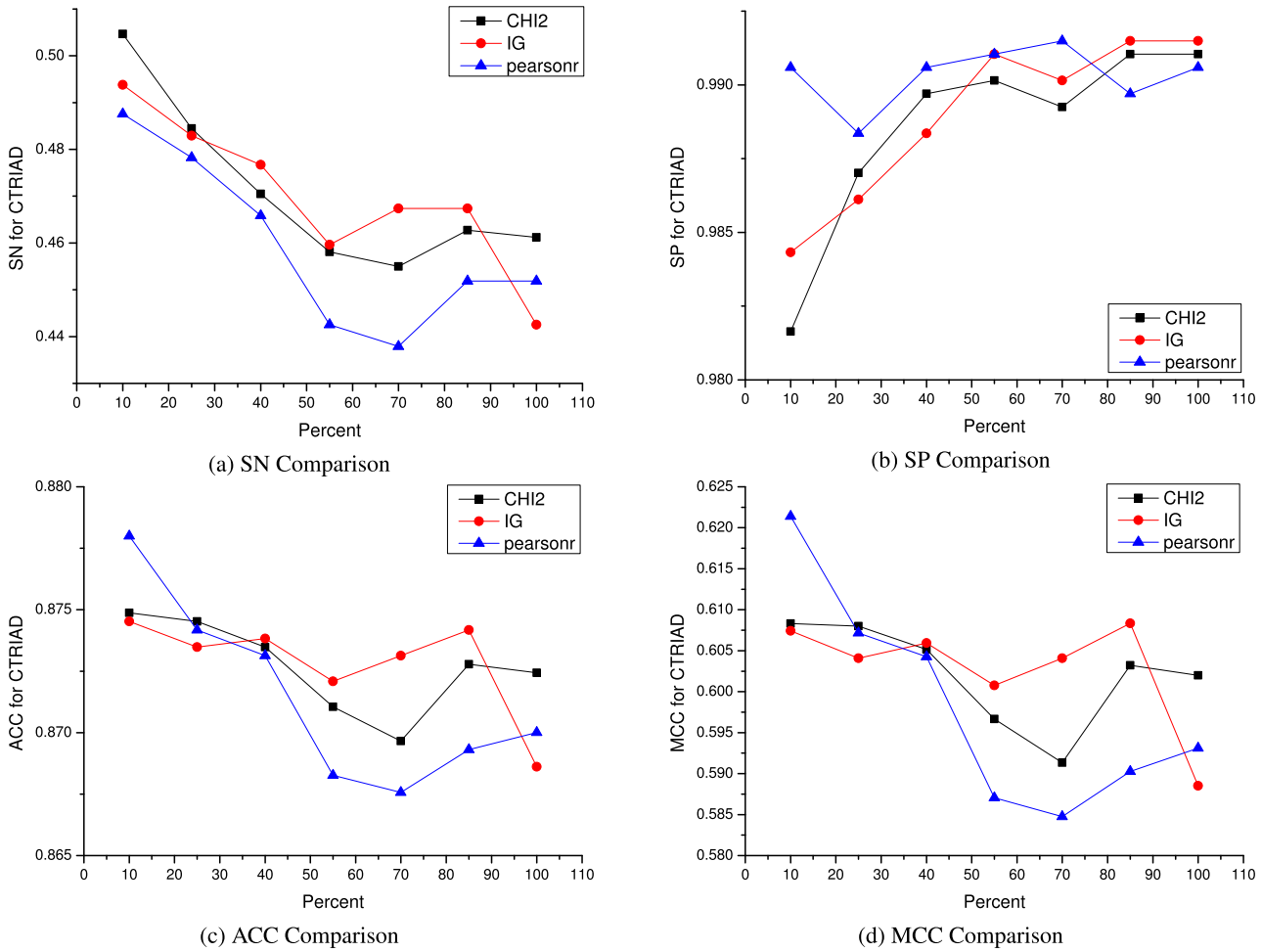


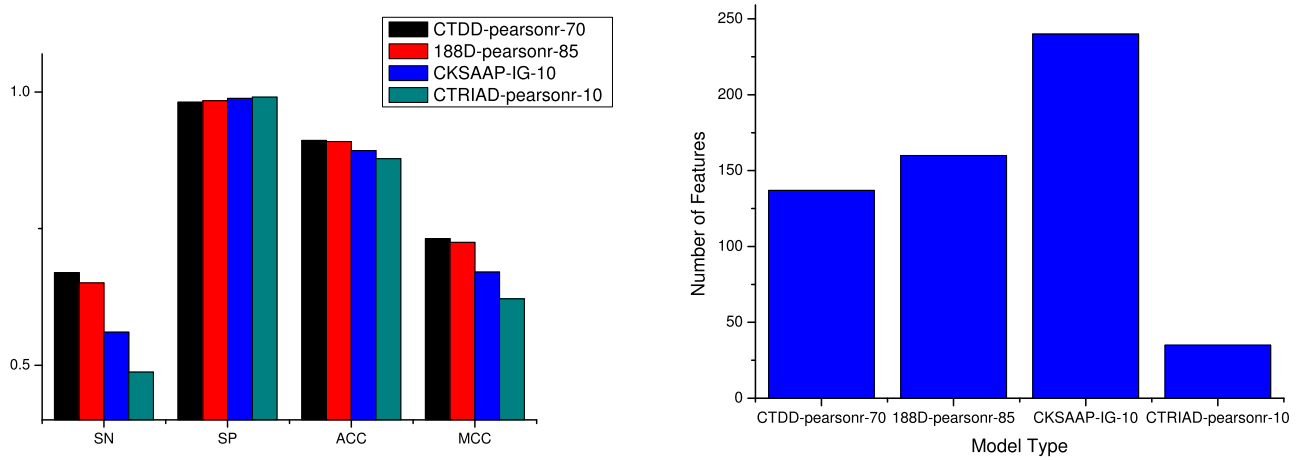
FIGURE 5. Performance comparison of different feature compressing methods for CTRIAD.

classify SNARE proteins based on the compressed feature set. Fig. 5a is the comparison result of SN indexes of various compression algorithms. As shown in the Figure, the compression effect of various feature compression methods on the CTRIAD feature set is similar to the compression effect of the CKSAAP feature set in III-B. When the number of features contained in the compressed feature set is small, the performance of SN is the best. Since the CTRIAD feature set contains a total of 343 features, Figure 5a shows that when the CHI2 method is used to compress the CTRIAD feature set, SN works best when the first 10% of features are taken. As shown in Figure 5b, when using the Pearson method to compress the CTRIAD feature set, the SP achieves the best effect when the feature is taken at 70%. Figure 5c and Figure 5d are the comparison results of ACC and MCC indicators of various compression algorithms. The results of these two indicators are similar to those of SN, and the best performance is achieved when 10% of the features are selected as classification features. In summary, for the CTRIAD feature collection method, the Pearson feature compression method can obtain the best feature compression effect, and the best effect is achieved when the first 10% of the features are selected.

E. COMPARISON OF DIFFERENT KINDS OF COMPRESSING METHODS

In this section, we compare the performance of different combinations of the feature compressing algorithms and feature extraction algorithms. In the last four sections, we find an effective feature compressing methods for each kind of feature extraction method. For the 188D feature extraction method, the Pearson method with 160 features is the best. For the CKSAAP feature extraction method, the IG method with 240 features is the best. For the CTDD feature extraction method, the Pearson method with 137 features achieves the best performance. For the CTRIAD feature extraction method, the Pearson method with 35 features selected achieves the best performance. In this experiment, we compare the performance of the four models. The comparison results are shown in figure 6.

Figure 6a shows that, for SN, the CTDD based model achieves the best performance. For SP, even though the performance of the CTRIAD based model is the best among the four models, the difference among different models is small. For ACC and MCC, the CTDD based model achieves the best performance. The performance of the 188D based



(a) Performance Comparison

(b) Feature Number Comparison

FIGURE 6. Comparison among different kinds of compressing methods.

model is in second place. The CKSAAP based model is the third. Taking the performance of feature compressing into consideration, the number of compressed features for CTDD, 188D, CKSAAP and CTRIAD, shown in figure 6b, are 137, 160, 240 and 35 respectively. We can conclude that the CTDD based model with Pearson compressing method is the best model.

IV. CONCLUSION

In this paper, three kinds of feature selection methods were applied to four kinds of feature sets extracted by four feature extraction methods from the SNARE proteins. The optimal compressing feature set four each kind of feature extraction method has been found. By comparing the performance of all the optimal compressing feature set, a Pearson based feature compressing model is proposed to identify the SNARE proteins accurately and efficiently.

REFERENCES

[1] R. Jahn and R. H. Scheller, "SNAREs—Engines for membrane fusion," *Nature Rev. Mol. Cell Biol.* vol. 7, pp. 631–643, Aug. 2006.
[2] A. D. J. van Dijk, D. Bosch, C. J. F. ter 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.
[3] C. Hou, Y. Wang, J. Liu, C. Wang, and J. Long, "Neurodegenerative disease related proteins have negative effects on SNARE-mediated membrane fusion in pathological confirmation," *Frontiers Mol. Neurosci.*, vol. 10, p. 66, Mar. 2017.
[4] W. G. Honer, P. Falkai, T. A. Bayer, J. Xie, L. Hu, H.-Y. Li, V. Arango, J. J. Mann, A. J. Dwork, and W. S. Trimble "Abnormalities of SNARE mechanism proteins in anterior frontal cortex in severe mental illness," *Cerebral Cortex*, vol. 12, no. 4, pp. 349–356, Apr. 2002.
[5] J. Meng and J. Wang, "Role of SNARE proteins in tumourigenesis and their potential as targets for novel anti-cancer therapeutics," *Biochim. et Biophys. Acta (BBA)-Rev. Cancer*, vol. 1856, no. 1, pp. 1–12, Aug. 2015.
[6] Q. Sun, X. Huang, Q. Zhang, J. Qu, Y. Shen, X. Wang, H. Sun, J. Wang, L. Xu, X. Chen, and B. Ren, "SNAP23 promotes the malignant process of ovarian cancer," *J. Ovarian Res.*, vol. 9, no. 1, p. 80, Dec. 2016.
[7] T. Weimbs, S. H. Low, S. J. Chapin, K. E. Mostov, P. Bucher, and K. Hofmann, "A conserved domain is present in different families of vesicular fusion proteins: A new superfamily," *Proc. Nat. Acad. Sci. USA*, vol. 94, no. 7, pp. 3046–3051, Apr. 1997.
[8] A. C. Yoshizawa, S. Kawashima, S. Okuda, M. Fujita, M. Itoh, Y. Moriya, M. Hattori, and M. Kanehisa, "Extracting sequence motifs and the phylogenetic features of SNARE-dependent membrane traffic," *Traffic*, vol. 7, no. 8, pp. 1104–1118, Aug. 2006.
[9] T. H. Kloepper, C. N. Kienle, and D. Fasshauer, "An elaborate classification of SNARE proteins sheds light on the conservation of the eukaryotic endomembrane system," *Mol. Biol. Cell*, vol. 18, no. 9, pp. 3463–3471, Sep. 2007.
[10] T. H. Kloepper, C. N. Kienle, and D. Fasshauer, "SNAREing the basis of multicellularity: Consequences of protein family expansion during evolution," *Mol. Biol. Evol.*, vol. 25, no. 9, pp. 2055–2068, Jun. 2008.
[11] X. Shi, P. Halder, H. Yavuz, R. Jahn, and H. A. Shuman, "Direct targeting of membrane fusion by SNARE mimicry: Convergent evolution of Legionella effectors," *Proc. Nat. Acad. Sci. USA*, vol. 113, no. 31, pp. 8807–8812, Aug. 2016.
[12] B. Lu, "The destructive effect of botulinum neurotoxins on the SNARE protein: SNAP-25 and synaptic membrane fusion," *PeerJ*, vol. 3, p. e1065, Jun. 2015.
[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. 971–980, 2018.
[14] L. Yu and L. Gao, "Human pathway-based disease network," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 16, no. 4, pp. 1240–1249, Jul. 2019.
[15] 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.
[16] 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.
[17] L. Wei, P. Xing, G. Shi, Z. Ji, and Q. Zou, "Fast prediction of protein methylation sites using a sequence-based feature selection technique," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 16, no. 4, pp. 1264–1273, Jul. 2019.
[18] 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.
[19] 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.
[20] J. He, T. Fang, Z. Zhang, B. Huang, X. Zhu, and Y. Xiong, "PseUI: Pseudouridine sites identification based on RNA sequence information," *BMC Bioinf.*, vol. 19, no. 1, p. 306, Dec. 2018.
[21] Q. Xu, Y. Xiong, H. Dai, K. M. Kumari, Q. Xu, H.-Y. Ou, and D.-Q. Wei, "PDC-SGB: Prediction of effective drug combinations using a stochastic gradient boosting algorithm," *J. Theor. Biol.*, vol. 417, pp. 1–7, Mar. 2017.
[22] J. Zhang and B. Liu, "A review on the recent developments of sequence-based protein feature extraction methods," *Current Bioinf.*, vol. 14, no. 3, pp. 190–199, Mar. 2019.

[23] 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, 2019.
[24] Y. Wang, S. Yang, J. Zhao, W. Du, Y. Liang, C. Wang, F. Zhou, Y. Tian, and Q. Ma, "Using machine learning to measure relatedness between genes: A multi-features model," *Sci. Rep.*, vol. 9, no. 1, Dec. 2019, Art. no. 4192.
[25] X. Zeng, S. Zhu, X. Liu, Y. Zhou, R. Nussinov, and F. Cheng, "DeepDR: A network-based deep learning approach to in silico drug repositioning," *Bioinformatics*, vol. 35, no. 24, pp. 5191–5198, Dec. 2019, doi: 10.1093/bioinformatics/btz418.
[26] P. Zhu, Q. Hu, Q. Hu, C. Zhang, and Z. Feng, "Multi-view label embedding," *Pattern Recognit.*, vol. 84, pp. 126–135, Dec. 2018.
[27] P. Zhu, Q. Hu, Y. Han, C. Zhang, and Y. Du, "Combining neighborhood separable subspaces for classification via sparsity regularized optimization," *Inf. Sci.*, vols. 370–371, pp. 270–287, Nov. 2016.
[28] P. Zhu, Q. Xu, Q. Hu, and C. Zhang, "Co-regularized unsupervised feature selection," *Neurocomputing*, vol. 275, pp. 2855–2863, Jan. 2018.
[29] P. Zhu, Q. Xu, Q. Hu, C. Zhang, and H. Zhao, "Multi-label feature selection with missing labels," *Pattern Recognit.*, vol. 74, pp. 488–502, Feb. 2018.
[30] P. Zhu, W. Zhu, Q. Hu, C. Zhang, and W. Zuo, "Subspace clustering guided unsupervised feature selection," *Pattern Recognit.*, vol. 66, pp. 364–374, Jun. 2017.
[31] B. Małysiak-Mrozek, T. Baron, and D. Mrozek, "Spark-IDPP: High-throughput and scalable prediction of intrinsically disordered protein regions with spark clusters on the cloud," *Cluster Comput.*, vol. 22, no. 2, pp. 487–508, Jun. 2019, doi: 10.1007/s10586-018-2857-9.
[32] Q. Zou, D. Mrozek, Q. Ma, and Y. Xu, "Scalable data mining algorithms in computational biology and biomedicine," *BioMed Res. Int.*, vol. 2017, pp. 1–3, Feb. 2017, doi: 10.1155/2017/5652041.
[33] B. Malysiak-Mrozek and D. Mrozek, "An improved method for protein similarity searching by alignment of fuzzy energy signatures," *Int. J. Comput. Intell. Syst.*, vol. 4, no. 1, pp. 75–88, Feb. 2011, doi: 10.1080/18756891.2011.9727765.
[34] D. Mrozek, B. Socha, S. Kozielski, and B. Małysiak-Mrozek, "An efficient and flexible scanning of databases of protein secondary structures: With the segment index and multithreaded alignment," *J. Intell. Inf. Syst.*, vol. 46, no. 1, pp. 213–233, Feb. 2016, doi: 10.1007/s10844-014-0353-0.
[35] M. Dash and H. Liu, "Feature selection for classification," *Intell. Data Anal.*, vol. 1, nos. 1–4, pp. 131–156, 1997.
[36] G. Wang, Y. Wang, W. Feng, X. Wang, J. Y. Yang, Y. Zhao, Y. Wang, and Y. Liu, "Transcription factor and microRNA regulation in androgen-dependent and-independent prostate cancer cells," *BMC Genomics*, vol. 9, no. 2, p. S22, 2008.
[37] Q. Jiang, G. Wang, S. Jin, Y. Li, and Y. Wang, "Predicting human microRNA-disease associations based on support vector machine," *Int. J. Data Mining Bioinf.*, vol. 8, no. 3, pp. 282–293, 2013.
[38] L. Xu, G. Liang, S. Shi, and C. Liao, "SeqSVM: A sequence-based support vector machine method for identifying antioxidant proteins," *Int. J. Mol. Sci.*, vol. 19, no. 6, p. 1773, Jun. 2018.
[39] L. Xu, G. Liang, L. Wang, and C. Liao, "A novel hybrid sequence-based model for identifying anticancer peptides," *Genes*, vol. 9, no. 3, p. 158, Mar. 2018.
[40] L. Dou, X. Li, H. Ding, L. Xu, and H. Xiang, "Is there any sequence feature in the RNA pseudouridine modification prediction problem?" *Mol. Therapy-Nucleic Acids*, vol. 19, pp. 293–303, Mar. 2020.
[41] 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," *Frontiers Genet.*, vol. 9, p. 745, Jan. 2019.
[42] B. Liu, Y. Zhu, and K. Yan, "Fold-LTR-TCP: protein fold recognition based on triadic closure principle," *Briefings Bioinf.*, Dec. 2019, doi: 10.1093/bib/bbz139.
[43] K. Chen, Y. Jiang, L. Du, and L. Kurgan, "Prediction of integral membrane protein type by collocated hydrophobic amino acid pairs," *J. Comput. Chem.*, vol. 30, no. 1, pp. 163–172, Jan. 2009.
[44] K. Chen, L. A. Kurgan, and J. Ruan, "Prediction of flexible/rigid regions from protein sequences using k-spaced amino acid pairs," *BMC Struct. Biol.*, vol. 7, no. 1, p. 25, 2007.
[45] C. Z. Cai, L. Y. Han, Z. L. Ji, X. Chen, and Y. Z. Chen, "SVM-prot: Web-based support vector machine software for functional classification of a protein from its primary sequence," *Nucleic Acids Res.*, vol. 31, no. 13, pp. 3692–3697, Jul. 2003.
[46] K. Chen, L. Kurgan, and M. Rahbari, "Prediction of protein crystallization using collocation of amino acid pairs," *Biochem. Biophys. Res. Commun.*, vol. 355, no. 3, pp. 764–769, Apr. 2007.
[47] I. Dubchak, I. Muchnik, S. R. Holbrook, and S. H. Kim "Prediction of protein folding class using global description of amino acid sequence," *Proc. Nat. Acad. Sci. USA*, vol. 92, no. 19, pp. 8700–8704, 1995.
[48] I. Dubchak, I. Muchnik, C. Mayor, I. Dralyuk, and S.-H. Kim, "Recognition of a protein fold in the context of the SCOP classification," *Proteins*, vol. 35, no. 4, pp. 401–407, 1999.
[49] J. Shen, J. Zhang, X. Luo, W. Zhu, K. Yu, K. Chen, Y. Li, and H. Jiang, "Predicting protein-protein interactions based only on sequences information," *Proc. Nat. Acad. Sci. USA*, vol. 104, no. 11, pp. 4337–4341, Mar. 2007.
[50] Z. Chen, P. Zhao, F. Li, A. Leier, T. T. Marquez-Lago, Y. Wang, G. I. Webb, A. I. Smith, R. J. Daly, K.-C. Chou, and J. Song, "iFeature: A Python package and Web server for features extraction and selection from protein and peptide sequences," *Bioinformatics*, vol. 34, no. 14, pp. 2499–2502, 2018.
[51] A. Bairoch, R. Apweiler, C. H. Wu, W. C. Barker, B. Boeckmann, S. Ferro, E. Gasteiger, H. Huang, R. Lopez, M. Magrane, M. J. Martin, D. A. Natale, C. O'Donovan, N. Redaschi, and L.-S. L. Yeh, "The universal protein resource (UniProt)," *Nucleic Acids Res.*, vol. 33, no. 1, pp. 154–159, 2005.
[52] R. Apweiler, A. Bairoch, C. H. Wu, W. C. Barker, B. Boeckmann, S. Ferro, E. Gasteiger, H. Huang, R. Lopez, M. Magrane, M. J. Martin, D. A. Natale, C. O'Donovan, N. Redaschi, and L.-S. L. Yeh, "UniProt: The universal protein knowledgebase," *Nucleic Acids Res.*, vol. 32, no. 1, pp. 115–119, 2004.
[53] C. H. Wu, R. Apweiler, A. Bairoch, D. A. Natale, W. C. Barker, B. Boeckmann, S. Ferro, E. Gasteiger, H. Huang, R. Lopez, M. Magrane, M. J. Martin, R. Mazumder, C. O'Donovan, N. Redaschi, and B. Suzek, "The universal protein resource (UniProt): An expanding universe of protein information," *Nucleic Acids Res.*, vol. 34, no. 1, pp. 187–191, 2006.
[54] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: An update," *ACM SIGKDD Explor. Newslett.*, vol. 1, pp. 10–18, Nov. 2009.

GUILIN LI was born in Harbin, Heilongjiang, China, in 1979. He received the B.S. and M.S. degrees in computer science and technology and the Ph.D. degree in computer software and theory from the Harbin Institute of Technology, Harbin, in 2001, 2003, and 2009, respectively.

From 2009 to 2013, he was an Assistant Professor with the Software Department, Xiamen University, Fujian, China, where he has been an Associate Professor with the School of Informatics, since 2013. He is the author of more than 30 articles. His research interests include bioinformatics, feature engineering, machine learning, and deep learning.