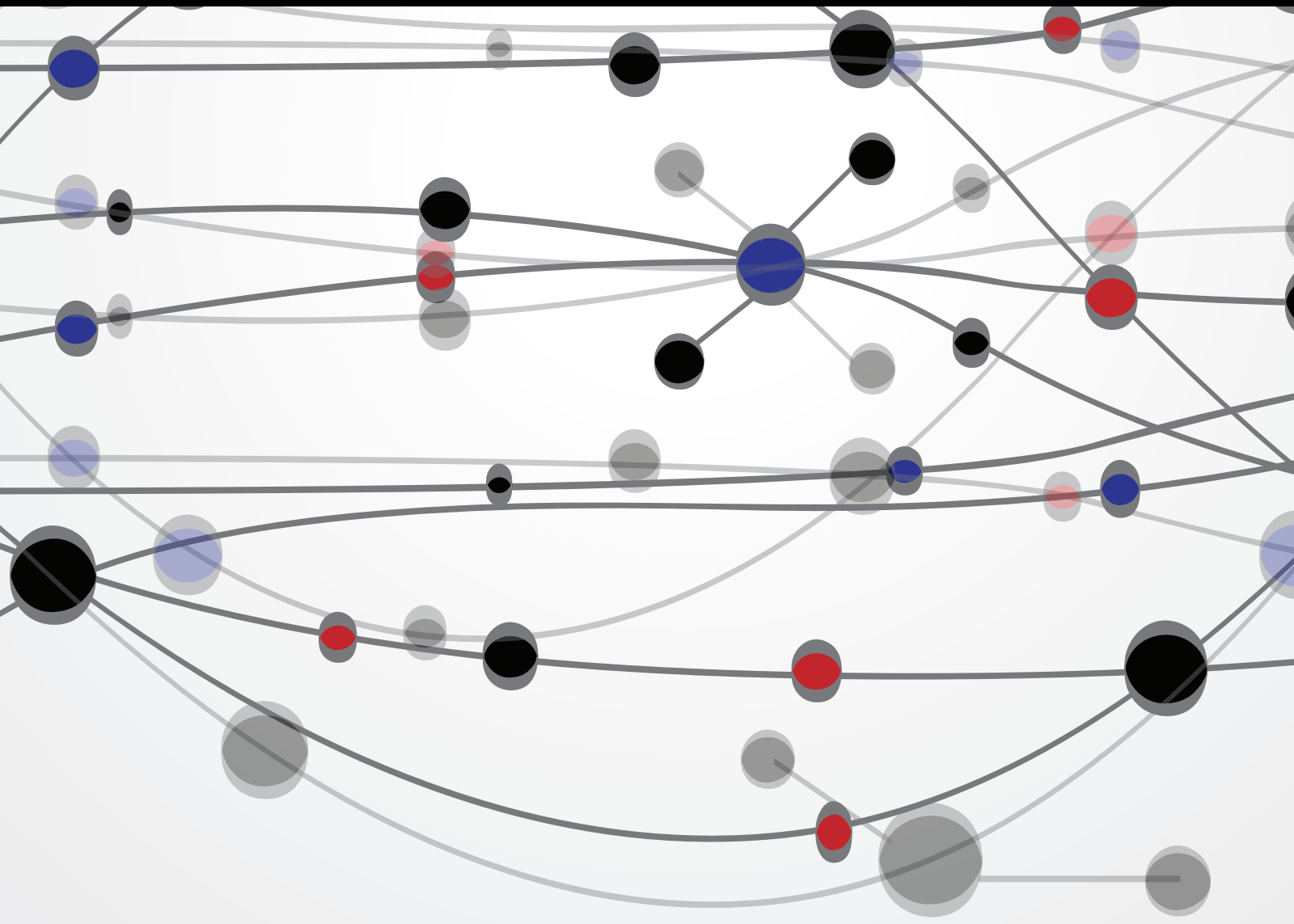
The Scientific World Journal

Application of Machine Learning Method in Genomics and Proteomics

Guest Editors: Hao Lin, Wei Chen, Ramu Anandakrishnan, and Dariusz Plewczynski



Application of Machine Learning Method in Genomics and Proteomics

The Scientific World Journal

Application of Machine Learning Method in Genomics and Proteomics

Guest Editors: Hao Lin, Wei Chen, Ramu Anandakrishnan, and Dariusz Plewczynski

Contents

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2015, Article ID 914780, 2 pages
http://dx.doi.org/10.1155/2015/914780

Editorial

Application of Machine Learning Method in Genomics and Proteomics

Hao Lin,[1] Wei Chen,[2] Ramu Anandakrishnan,[3] and Dariusz Plewczynski[4]

[1]*Center of Bioinformatics, School of Life Science and Technology, University of Electronic Science and Technology of China, Chengdu 610054, China*
[2]*Department of Physics, Center for Genomics and Computational Biology, College of Sciences, Hebei United University, Tangshan 063000, China*
[3]*Department of Computer Science, Laboratory for Theoretical and Computational Molecular Biophysics, Virginia Tech, Blacksburg, VA 24060, USA*
[4]*Center of New Technologies, University of Warsaw, 02097 Warszawa, Poland*

Correspondence should be addressed to Hao Lin; hlin@uestc.edu.cn

Received 19 January 2015; Accepted 19 January 2015

With the avalanche of genomic and proteomic data generated in the postgenomic age, it is highly desirable to develop automated methods for rapidly and effectively analyzing and predicting the structure, function, and other properties of DNA and protein. The machine learning methods have become an important strategy for the discovery of potential knowledge in genomics and proteomics. Researches in recent years have shown tremendous advances in the properties prediction of DNA fragments and protein sequences by various pattern recognition methods. These techniques provide economical and timesaving solutions for identifying the properties of DNA and protein. This special issue was hosted for the recent development of the application of machine learning methods in genomics and proteomics.

In this special issue, five works focused on the protein classification. How to extract key features from a protein was a key step in the discrimination of protein class. B. Liu et al. proposed to use Position-Specific Score Matrix (PSSM) and Accessible Surface Area (ASA) to formulate protein samples. The hidden Markov support vector machine (HM-SVM) was employed to predict protein binding site. Simulation in five-fold cross-validation on a benchmark dataset including 1124 protein chains showed that their method is more accurate for protein binding site prediction than some state-of-the art methods. This method can also be applied in DNA binding site, vitamin binding site, and posttranslational modification of proteins.

Based on chemical shift (CS) information derived from nuclear magnetic resonance (NMR), F. Yonge proposed a novel feature to predict protein supersecondary structures. The quadratic discriminant (QD) analysis was selected as the prediction algorithm. Overall accuracy in threefold cross-validation is 77.3% for predicting four types of supersecondary structures. According to the concept of pseudo amino acids, G.-L. Fan et al. proposed the average chemical shifts (ACS) composition and established an online webserver called acACS which was calculated from average chemical shift information and protein secondary structure. By using SVM as the classification algorithm, the acACS was used in the discrimination between acidic and alkaline enzymes and between bioluminescent and nonbioluminescent proteins. Encouraging results were achieved. The protein secondary structure, structure class, and disorder region can be predicted using the AC-based method.

L. Nanni et al. proposed to combine different features to improve protein prediction. These features include amino acids composition, PSSM, and substitution matrix representation (SMR). Each feature is used to train a separate SVM. Total of 15 benchmark datasets were used to evaluate the performance of their proposed method. Comparative

results show that the PSSM always produces good accuracies. However, no single descriptor is superior to all others across all test datasets. The major contribution in this paper is to propose an ensemble of classifiers for sequence-based protein classification.

H. Lin et al. briefly reviewed the development of ion channel prediction using machine learning method. They initially introduced how to construct a valid and objective benchmark dataset to train and test the predictor. Subsequently, the mathematical descriptors were presented to formulate the ion channel sequences. Moreover, two feature selection techniques on how to optimize feature set were described. Finally, the support vector machine was suggested performing classification. The methods introduced in that review can be generalized into other protein prediction fields as well.

The paper from P. Feng et al. was the unique work focused on DNA prediction using machine learning method. They proposed a novel descriptor called pseudo K-tuple nucleotide composition (PseKNC) to formulate the DNA sequences. The feature is calculated from K-tuple nucleotide composition and the structural correlation of DNA dinucleotides. Subsequently, the SVM was used to predict DNase I hypersensitive sites. The jackknife cross-validated accuracy is 83%, which is competitive with that of the existing method. This new descriptor can also be widely used in DNA regulatory elements prediction.

Hao Lin
Wei Chen
Ramu Anandakrishnan
Dariusz Plewczynski

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2015, Article ID 945927, 7 pages
http://dx.doi.org/10.1155/2015/945927

Review Article

Briefing in Application of Machine Learning Methods in Ion Channel Prediction

Hao Lin[1] and Wei Chen[2]

[1] *Key Laboratory for Neuro-Information of Ministry of Education, Center of Bioinformatics, School of Life Science and Technology, University of Electronic Science and Technology of China, Chengdu 610054, China*
[2] *Department of Physics, School of Sciences and Center for Genomics and Computational Biology, Hebei United University, Tangshan 063000, China*

Correspondence should be addressed to Hao Lin; hlin@uestc.edu.cn and Wei Chen; greatchen@heuu.edu.cn

Received 31 July 2014; Accepted 11 September 2014

Academic Editor: Ramu Anandakrishnan

In cells, ion channels are one of the most important classes of membrane proteins which allow inorganic ions to move across the membrane. A wide range of biological processes are involved and regulated by the opening and closing of ion channels. Ion channels can be classified into numerous classes and different types of ion channels exhibit different functions. Thus, the correct identification of ion channels and their types using computational methods will provide in-depth insights into their function in various biological processes. In this review, we will briefly introduce and discuss the recent progress in ion channel prediction using machine learning methods.

1. Background

Ion channels are a diverse group of proteins that extend across the lipid membrane of cells and form channel pores [1]. They allow ions to move into and out of the cell to establish and control the voltage gradient across the cell membrane in response to stimuli, such as ligand, voltage, and pressure changes. Many biological processes including muscle contraction, neuronal excitability, epithelial transport of nutrients and ions, hormone secretion, T-cell activation, and pancreatic beta-cell insulin release are all controlled and regulated by ion channels [2].

It has been reported that the normal function of ion channels can be disrupted by chemicals and genetics, which would result in negative impact on the organism [2]. For example, channelopathies are caused by mutations in ion channel-encoding genes [3]. Moreover, various neurotoxins bind to ion channels to modulate the nervous systems of animals. Since ion channels have such important biological function in various biological processes, scientists have developed drugs to target them for disease therapy. Ion channels have been demonstrated as valuable targets for the treatment of epilepsy, chronic pain, and other diseases [4].

Over 300 types of ion channels have been found in living cells [5]. Most channels are ion-selective and ion-specific. For example, most of potassium channels have a permeability ratio for potassium over sodium of 1000 : 1 [6]. Based on their biological properties, ion channels can be clustered into numerous types. The ion channels activated by the binding of ligand molecules (such as a neurotransmitter) are called ligand-gated ion channels (LGIC) that can be further classified into three superfamilies, namely, Cys-loop receptors, ionotropic glutamate receptors, and ATP-gated channels. Voltage-gated ion channels (VGIC) are another kind of ion channels which open to allow ions to pass through the membrane in response to the changes in electrical potential difference. According to ion type permeability, the VGICs can be further classified into potassium (K), sodium (Na), calcium (Ca), and anion VGICs. Moreover, some ion channels can also be opened and closed by mechanical forces, temperature, and pressure. However, the number of these ion channels is too few to have statistical significance. Thus, this review focuses on the prediction of ligand-gated and voltage-gated ion channels.

Different ion channel types perform different biological functions and regulate different biological possesses. To

Table 1: List of databases related with ion channels.

Name	URL
PDB	http://www.rcsb.org/pdb/home/home.do
Uniprot	http://www.uniprot.org
IUPHAR	http://www.iuphar-db.org
LGIC	http://www.ebi.ac.uk/compneur-srv/LGICdb/LGICdb.php
VKCDB	http://vkcdb.biology.ualberta.ca/index.php

Table 2: List of three programs.

Name	URL
BLASTClust	http://www.ncbi.nlm.nih.gov/Web/Newsltr/Spring04/blastlab.html
CD-HIT	http://weizhong-lab.ucsd.edu/cd-hit
PISCES	http://dunbrack.fccc.edu/PISCES.php

identify the types of ion channels, traditional biochemical experimental methods are time-consuming and costly, whereas computational methods are cost-effective. Therefore, in this paper, we review the development of machine learning methods in the prediction of ion channel and their types. To predict ion channels using machine learning method, the following issues should be considered. (i) How to construct a valid and objective benchmark data set to train and test the predictor? (ii) How to formulate the ion channel sequences using an effective mathematical descriptor which can truly reflect the properties of samples? (iii) How to develop or use a machine learning algorithm to perform the prediction? (iv) What kind of cross-validation tests should be used to evaluate the anticipated accuracy of the predictor? We will discuss each issue in turn.

2. Published Databases

The first essential requirement in developing computational methods for the prediction of ion channels is to obtain a benchmark database. At present, many public databases are available online. Some of these original databases, such as protein data bank (PDB) [7] and universal protein resource (UniProt) [8], have deposited many ion channel data. Based on these databases and related publications, some special databases such as IUPHAR (International Union of Basic and Clinical Pharmacology) database [9], ligand-gated ion channel database [10], and VKCDB [11] have been built. The web addresses of these databases are listed in Table 1.

However, the databases listed above are not suitable for ion channel prediction using machine learning methods, because the data deposited are redundant and are of low quality. A reliable and objective benchmark dataset should be constructed by the following strategies: (i) if the protein sequence of an ion channel contains ambiguous residues (such as "B," "X," and "Z"), the ion channel must be excluded; (ii) if sequences are fragments of other proteins, the sequences must be excluded; (iii) if an ion channel is inferred from homology or prediction, the ion channel must also be excluded; and (iv) the highly similar sequences must be excluded for objectivity, because the high similarity data will lead to overestimating the performance of the proposed predictors.

In order to exclude highly similar sequences from these datasets, BLASTClust, CD-HIT [12], and PISCES [13] have been developed and could be freely obtained at the addresses listed in Table 2. BLASTClust is a program that can be used to cluster either protein or nucleotide sequences. However, since it requires all against all comparisons of sequences for optimal results [14], the efficiency of this program is relatively low. Owing to the clustering efficiency and capability to handle extremely large databases, CD-HIT has been widely employed to remove redundant sequences. However, CD-HIT cannot deal with sequences with sequence identity below 40%. To overcome this shortcoming, PISCES was proposed in 2003, which can exclude proteins with the sequence identity of 25% [13].

According to the above mentioned public databases and sequence culling programs, four benchmark datasets of ion channels have been proposed in previous studies [15–19].

The first benchmark dataset S1 [19] contains 1574 nonion channels and 473 ion channels, of which 164 are potassium, 27 sodium, 27 calcium, and 18 chloride VGICs. The sequence identity between any two sequences in S1 is less than 90%. These data are derived from the Swiss-Prot database.

The second nonredundant benchmark dataset S2 [17] contains 37 Kv1, 16 Kv2, 18 Kv3, 15 Kv4, and 14 Kv7 subfamilies of voltage-gated K+ channels. These data are derived from the VKCDB database.

The third benchmark dataset S3 [16] contains 300 nonion channel membrane proteins and 298 ion channel proteins. The ion channel dataset contains 148 VGICs (81 potassium, 29 calcium, 12 sodium, and 26 anion VGICs) and 150 LGICs. The sequence identity of this dataset is less than 40%. These data are derived from the Uniprot and LGIC databases.

The fourth benchmark dataset S4 [15] contains 217 voltage-gated K+ channels, composed of 82 Kv1, 16 Kv2, 37 Kv3, 32 Kv4, 10 Kv6, and 40 Kv7 families, respectively. The sequence identity of this dataset is less than 60%. These data are derived from the VKCDB database.

3. Methods

3.1. Protein Description. Use of informative parameters to represent the ion channel samples is the second essential requirement for bioinformatics prediction. Here, three kinds of features, amino acid compositions, dipeptide compositions, and tripeptide compositions, were used to represent ion channels and expressed as follows:

$$f_{20}(i) = \frac{x_{20}(i)}{\sum_i x_{20}(i)}, \tag{1}$$

$$f_{400}(j) = \frac{y_{400}(j)}{\sum_j y_{400}(j)}, \tag{2}$$

$$f_{8000}(k) = \frac{z_{8000}(k)}{\sum_k z_{8000}(k)}, \tag{3}$$

where $x_{20}(i)$, $y_{400}(j)$, and $z_{8000}(k)$ are the occurrence number of residues i, the number of occurrences of dipeptide j, and the number of occurrences of tripeptide k in the protein sequence of an ion channel, respectively. 20, 400, and 8000 are the number of the standard amino acids, the number of combination of dipeptides, and the number of combination of tripeptides, respectively.

3.2. Feature Selection. Theoretically, high dimension features will lead to three serious issues, that is, overfitting, information redundancy, or noise and dimension disaster [20]. These issues would result in low generalization ability of the predictor, poor prediction accuracy, and time-consuming computations. Thus, it is necessary to use feature selection techniques to optimize feature set for economizing the time for computation and building robust prediction models. In the following section, we will discuss how to use three feature selection techniques, that is, analysis of variance, correlation-based feature selection, and binomial distribution, to select optimal features.

3.2.1. Analysis of Variance (ANOVA). To evaluate the contribution of features to the classification, the F value ($F(\lambda)$) of the λth feature can be defined as

$$F(\lambda) = \frac{s_B^2(\lambda)}{s_W^2(\lambda)}, \tag{4}$$

where $s_B^2(\lambda)$ is called means square between (MSB) and denotes the sample variance between classes and $s_W^2(\lambda)$ is called mean square within (MSW) and denotes sample variance within classes. They can be calculated by [16]

$$s_B^2(\lambda) = \frac{\sum_{i=1}^{K} n_i \left(\left(\sum_{j=1}^{n_i} f_{ij}(\lambda)\right)/n_i - \left(\sum_{i=1}^{K}\sum_{j=1}^{n_i} f_{ij}(\lambda)\right)/\sum_{i=1}^{K} n_i\right)^2}{df_B}, \tag{5}$$

$$s_W^2(\lambda) = \frac{\sum_{i=1}^{K}\sum_{j=1}^{n_i}\left(f_{ij}(\lambda) - \left(\sum_{j=1}^{n_i} f_{ij}(\lambda)\right)/n_i\right)^2}{df_W}, \tag{6}$$

where K and N represent the number of classes and total number of samples, respectively. $f_{ij}(\lambda)$ represents the frequency of the λth feature of the jth sample in the ith class. n_i is the number of samples in the ith class. $df_B = K - 1$ is the degrees of freedom for MSB and $df_W = N - K$ the degrees of freedom for MSW.

Based on the theory of statistics, the $F(\lambda)$ in (4) obeys F sampling distribution with degrees of freedom df_B and df_W. The $F(\lambda)$ measures the contribution of the λth feature related to the class variables. In the absence of differences between groups, the $F(\lambda)$ will be close to 1. In other words, the feature with a larger $F(\lambda)$ indicates that it is a more highly relevant one for the target to be predicted. Thus, features can be initially ranked according to F value.

3.2.2. Correlation-Based Feature Selection (CFS). The heart of the correlation-based feature selection algorithm is to evaluate the merit of a feature subset and exclude the redundant features which are highly correlated with one or more of the other features. The merit of a feature subset S containing k features is defined by the following equation [15]:

$$\text{Merit}(S, k) = \frac{k\overline{r_{cf}}}{\sqrt{k + k(k-1)\overline{r_{ff}}}}, \tag{7}$$

where $\overline{r_{cf}}$ is the average feature-class correlation expressed as in the following equation and $\overline{r_{ff}}$ the average feature-feature intercorrelation, which can be defined as

$$\overline{r_{cf}} = \frac{1}{k}\sum_{f_z \in \theta} r_{cf_z}, \qquad \overline{r_{ff}} = \frac{2}{k(k-1)}\sum_{f_i, f_j \in \theta} r_{f_i f_j}, \tag{8}$$

where c is the class. The numerator in (8) indicates predictive ability of subset θ and the denominator stands for redundancy among the features. In fact, (7) is the Pearson's correlation where all variables have been normalized. The numerator gives an indication of how predictive a group of features are, whereas the denominator describes how much redundancy there is among them.

3.2.3. Binomial Distribution (BD). For a stochastic event, two possible cases, namely, occurrence and nonoccurrence, will happen when one observes the ith feature occurring in the kth type set [18]. Each outcome has a fixed probability when benchmark dataset has been fixed. This probability is called prior probability p_k.

The total occurrence number of the ith feature in benchmark dataset is expressed as N_i. That is to say, under the condition of the prior probability p_k, one performs trial or observation with N_i times. The posterior probability P_{ik} of the ith feature occurring n_{ik} or more times in the kth type set can be calculated as follows

$$P_{ik} = 1 - CL_{ik} = \sum_{m=n_{ik}}^{N_i} \frac{N_i}{m!\,(N_i - m)!} p_k^m (1 - p_k)^{N_i - m}, \tag{9}$$

where CL_{ik} is the confidence level (CL) of the ith feature in the kth dataset. Based on small probability event principle, if P_{ik} is a small value, it means the feature i appearing in dataset k is not random. The feature with a small P_{ik} indicates that it is a more highly relevant one for the target to be predicted. Thus, features can be initially ranked according to P_{ik} value.

The incremental feature selection (IFS) can be used to determine the optimal number of features. The IFS procedure includes the following steps: starting with one feature with the first score in the feature set, adding the second feature with the second score, adding the third feature with the third score, and repeating this process until all candidate features are added. Finally, the proposed machine learning methods are used to investigate the performance of each feature subset. The feature subset which can yield the maximum accuracy is the optimal feature subset.

3.3. Support Vector Machine (SVM). The third essential key for bioinformatics is to select an efficient and accurate machine learning method to make a predictive decision. SVM is a kind of machine learning method which has been successfully used in wide fields of ion channel prediction. Many researchers have developed free and convenient software packages for the implementation of SVM, such as LibSVM [21] and SVM_Light [22].

The basic idea of the SVM is described as follows. For a two-class classification problem, a series of training vectors $\vec{X}_i \in R^d$ $(i = 1, 2, \ldots, N)$ with corresponding labels $y_i \in \{+1, -1\}$ $(i = 1, 2, \ldots, N)$ can be generated. Here, +1 and −1, respectively, indicate the two classes. SVM maps the input vectors $\vec{X}_i \in R^d$ into a high dimensional feature space in order to construct an optimal separating hyperplane with the largest distance between the two classes. The decision function implemented by SVM is written as

$$f\left(\vec{X}\right) = \operatorname{sgn}\left(\sum_{i=1}^{N} y_i \alpha_i \cdot K\left(\vec{X}, \vec{X}_i\right) + b\right), \tag{10}$$

where $K(\vec{X}, \vec{X}_i)$ is a kernel function which defines an inner product in a high dimensional feature space. There are three kinds of kernel functions for the nonlinear classification problems defined as follows.

Polynomial function

$$K\left(\vec{X}_i, \vec{X}_j\right) = \left(\vec{X}_i \cdot \vec{X}_j + 1\right)^d. \tag{11}$$

Radial basis function (RBF)

$$K\left(\vec{X}_i, \vec{X}_j\right) = \exp\left(-\gamma \left\| \vec{X}_i - \vec{X}_j \right\|^2\right). \tag{12}$$

Sigmoid function

$$K\left(\vec{X}_i, \vec{X}_j\right) = \tanh\left[b\left(\vec{X}_i \cdot \vec{X}_j\right) + c\right]. \tag{13}$$

The coefficients α_i can be solved by the following convex quadratic programming (QP) problem:

$$\begin{aligned} &\text{Maximize} \quad \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j \cdot y_i y_j \cdot K\left(\vec{X}_i, \vec{X}_j\right) \\ &\text{subject to} \quad 0 \le \alpha_i \le C, \end{aligned} \tag{14}$$

where $\sum_{i=1}^{N} \alpha_i y_i = 0$, $i = 1, 2, \ldots, N$. The regularization parameter C can control the trade-off between margin and misclassification error.

For multiclass problems, several strategies such as one-versus-rest (OVR), one-versus-one (OVO), and DAGSVM are applied to extend the traditional SVM. Because the RBF usually outperforms polynomial function and sigmoid function, the RBF is widely used in bioinformatics. The regularization parameter C and kernel parameter γ were tuned to optimize the classification performance using grid search with cross-validation.

3.4. Criteria for Performance Evaluation. In developing a useful statistical predictor, it is very important to objectively evaluate its performance or anticipated success rate. Here, a set of more intuitive and easier-to-understand metrics is introduced. Those are sensitivity (Sn), specificity (Sp), accuracy (Acc), and Matthew's correlation coefficient (MCC) defined as [23]

$$\begin{aligned} \mathrm{Sn} &= 1 - \frac{N_-^+}{N^+}, \\ \mathrm{Sp} &= 1 - \frac{N_+^-}{N^-}, \\ \mathrm{Acc} &= 1 - \frac{\left(N_-^+ + N_+^-\right)}{\left(N^+ + N^-\right)}, \\ \mathrm{MCC} &= \frac{1 - \left(N_-^+/N^+ + N_+^-/N^-\right)}{\sqrt{\left(1 + \left(N_+^- - N_-^+\right)/N^+\right)\left(1 + \left(N_-^+ - N_+^-\right)/N^-\right)}}, \end{aligned} \tag{15}$$

where N^+ is the total number of the positives while N_-^+ is the number of the positives incorrectly predicted as the negatives; N^- is the total number of the negatives while N_+^- is the number of the negatives incorrectly predicted as positives. These four metrics are generally used in statistical prediction for quantitatively measuring the performance of a predictor from four different angles.

Three cross-validation tests, that is, independent dataset test, subsampling (or K-fold cross-validation) test, and jackknife test, are often used to evaluate the anticipated success rate of a predictor [24]. The K-fold cross-validation is a kind of rigorous and objective method for evaluating the predictive performance of predictors. For K-fold cross-validation, the dataset is divided into K equal parts. Of these K parts, $K - 1$ parts are used for training and the Kth part is used for testing. This process is repeated K times for all K parts and the success rate is the average of the K times tests. The jackknife test is

deemed the least arbitrary one and hence has been widely used in the realm of bioinformatics. In the jackknife test, each sequence in the training dataset is in turn singled out as an independent test sample and all the rule-parameters (Sn, Sp, Acc, and MCC) are calculated without including the one being identified.

4. Published Results

Although many works have investigated the dynamics of ion channel, only few pattern recognition methods focused on the prediction of ion channels. The pioneering works for the prediction of ion channels were carried out independently by two groups in 2006.

Based on the benchmark dataset S1, a SVM-based method (SVM_light package) was proposed to discriminate ion channels from nonion channels [19]. In five-fold cross-validation, the Accs of 82.89% and 85.56% were achieved by using amino acid composition (1) and dipeptide composition (2), respectively. Authors also investigated the performance of position-specific scoring matrix (PSSM) generated from PSI-BLAST (Position-Specific Iterative Basic Local Alignment Search Tool) which can provide the distant relationships between proteins. Three iterations of PSI-BLAST were carried out at a cut-off E-value of 0.01. Then the accuracy of 84.22% was obtained by using the five-fold cross-validation. By combining dipeptide composition with position-specific scoring matrix, the five-fold cross-validated accuracy increased to 89.11%. Subsequently, these methods were used to predict potassium, sodium, calcium, and chloride VGICs. For further improving the accuracy, the Hidden Markov model (HMM) profiles of the four types of VGICs were constructed using the HMMER software package. Each protein sequence was aligned in a multiple sequence alignment using ClustalW. The E-value threshold (E-value) was set to 0.01. As a result, the five-fold cross-validated Acc reached 97.78% by using the hybrid method that combines dipeptide-based SVM and hidden Markov model methods. The Sns (MCC) of potassium, sodium, calcium, and chloride VGIC predictions are 99.38% (0.96), 96.00% (0.93), 96.00% (0.98), and 86.67% (0.92). Based on these approaches, a web server VGIchan (http://www.imtech.res.in/raghava/vgichan/) was developed for predicting and classifying voltage-gated ion channels. This is the first online server for ion channel prediction using a machine learning method.

Based on the benchmark dataset S2, Liu et al. [17] predicted the five subfamilies of potassium VGICs by using SVM combined with dipeptide composition (2). In the jackknife cross-validation, the average Acc of 98.0% was achieved with the average Sn of 89.9%, Sp of 100%, and MCC of 0.94.

Although these two studies have achieved good results, the high sequence similarity of the two datasets might result in overestimating the performance and reducing the generalization ability of the proposed predictive models.

Recently, based on the benchmark dataset S3 and by using dipeptide composition (2) as parameters, Lin and Ding [16] successfully predicted ion channels and their types using Libsvm package. In jackknife cross-validation, the Accs of 85.0%, 89.9%, and 82.4% are obtained for the classification of ion channels and nonion channels, VGICs and LGICs, and the subclasses of VGICs, respectively. For further improving predictive performance of SVM model, the ANOVA (3)–(5) was firstly proposed to select the optimal dipeptide compositions (2). Then, the Accs increase from 85.0%, 89.9%, and 82.4% to 86.6%, 92.6%, and 87.8%, respectively, when using the 140, 159, and 232 optimal dipeptides according to the F values, respectively. These results demonstrate that the ANOVA is a powerful and efficient feature selection technique which can improve the predictive accuracy by excluding noise and redundant parameters. Based on this proposed method, an effective tool for predicting ion channels and their types, called IonchanPred, was constructed and can be freely downloaded from http://cobi.uestc.edu.cn/people/hlin/tools/IonchanPred/. By using the IonchanPred, the KCMA1 can be correctly identified, which is a potassium channel activated by either membrane depolarization or increase in cytosolic Ca2+ and plays a key role in controlling excitability in a number of systems. For comparison, this feature selection technique was also used to investigate the performance of SVM on the benchmark dataset S1. In five-fold cross-validation, the Acc and average accuracy are 97.97% and 95.55%, respectively. Comparison demonstrates again that the ANOVA is a powerful technique for feature selection.

Based on the benchmark dataset S4, Chen and Lin presented a SVM-based method (LibSVM package) to predict six subfamilies of potassium VGICs using amino acid composition and dipeptide composition [15]. The Acc of 87.39% was achieved in jackknife cross-validation. Furthermore, the CFS was proposed to find the best feature set. As a result, the maximum Acc of 93.09% was obtained in jackknife cross-validation when 118 features were used. For the convenience of the vast majority of experimental scientists, a predictive tool, called VKCPred, was constructed and can be freely downloaded from http://cobi.uestc.edu.cn/people/hlin/tools/VKCPred/. For further improving the accuracy, Liu et al. [18] proposed BD-based feature selection technique to pick out optimal tripeptides. The LibSVM was used to execute the SVM algorithm. The overall accuracy improved to 96.77% in jackknife cross-validation when 648 tripeptides were selected as optimal features. A user-friendly web-server called iVKC-OTC was established and can be freely accessible at http://lin.uestc.edu.cn/server/iVKC-OTC.

The four tools, VGIchan, IonchanPred, VKCPred, and iVKC-OTC, are listed in Table 3 for use by experimental researches.

5. Prospect

Ion channels are important drug targets. Using computational methods can provide valuable information for narrowing the scope of drug targets discovery. However, few methods have been applied in this realm and the accuracy is still far from that required for successful application.

Many machine learning methods such as neural network (NN) [25], K nearest neighbor (KNN) [26], extreme learning

Table 3: Summary of the ion channel prediction tools.

Name	URL
VGIchan	http://www.imtech.res.in/raghava/vgichan/
IonchanPred	http://cobi.uestc.edu.cn/people/hlin/tools/IonchanPred/
VKCPred	http://cobi.uestc.edu.cn/people/hlin/tools/VKCPred/
iVKC-OTC	http://lin.uestc.edu.cn/server/iVKC-OTC/

machine (ELM) [27], and deep learning (DL) [28] have been widely applied in computational proteomics. Some feature selection techniques such as minimum redundancy maximum relevance feature selection (mRMR) [29], manifold learning (ML) [30], principal component analysis (PCA) [31], and regularized trees [32] have also been developed and were gradually used to obtain optimal features that produce the highest predictive accuracy.

Developing a set of informative parameters to formulate the ion channel samples is also necessary for ion channel prediction. In this paper, only the amino acid, the dipeptide, and tripeptide composition were used to represent ion channels. The physiochemical characteristics [33], overrepresented motifs [34], and functional domains [35] can also be utilized in the field.

Of course, to construct better benchmark dataset which not only contains more sequences but also obeys more objective and strict standards can benefit the study ion channels. Now, with the avalanche of genome and proteome sequences generated in the postgenomic age, many ion channels are available in various sequence, structure, and reference database. Collecting and building these data is the key role in ion channel study.

In the future, we hope that researchers can focus on the three aspects discussed above for developing powerful and efficient predictors of ion channels.

6. Summary

This review focused on the development of prediction methods for ion channels in terms of the following issues:

(i) datasets of ion channel proteins,

(ii) machine learning methods to predict ion channels,

(iii) feature selection techniques to obtain optimal features for ion channel predictions,

(iv) prospect of ion channel predictions by using bioinformatics methods.

Conflict of Interests

The authors declare that there is no conflict of interests.

Acknowledgments

The authors would like to thank the anonymous reviewers for their constructive comments. This work was supported by the National Nature Scientific Foundation of China (nos. 61202256, 61301260, and 61100092), the Nature Scientific Foundation of Hebei Province (no. C2013209105), and the Fundamental Research Funds for the Central Universities (nos. ZYGX2012J113 and ZYGX2013J102).

References

[1] B. Hille, C. M. Armstrong, and R. MacKinnon, "Ion channels: from idea to reality," *Nature Medicine*, vol. 5, no. 10, pp. 1105–1109, 1999.

[2] D. C. Camerino, J.-F. Desaphy, D. Tricarico, S. Pierno, and A. Liantonio, "Therapeutic approaches to ion channel diseases," *Advances in Genetics*, vol. 64, pp. 81–145, 2008.

[3] R. S. Kass, "The channelopathies: novel insights into molecular and genetic mechanisms of human disease," *The Journal of Clinical Investigation*, vol. 115, no. 8, pp. 1986–1989, 2005.

[4] J. J. Clare, "Targeting ion channels for drug discovery," *Discovery Medicine*, vol. 9, no. 46, pp. 253–260, 2010.

[5] I. S. Gabashvili, B. H. A. Sokolowski, C. C. Morton, and A. B. S. Giersch, "Ion channel gene expression in the inner ear," *JARO: Journal of the Association for Research in Otolaryngology*, vol. 8, no. 3, pp. 305–328, 2007.

[6] T. Dudev and C. Lim, "Determinants of K^+ vs Na^+ selectivity in potassium channels," *Journal of the American Chemical Society*, vol. 131, no. 23, pp. 8092–8101, 2009.

[7] P. W. Rose, C. Bi, W. F. Bluhm et al., "The RCSB protein data bank: new resources for research and education," *Nucleic Acids Research*, vol. 41, pp. D475–D482, 2013.

[8] UniProt Consortium, "Update on activities at the Universal Protein Resource (UniProt) in 2013," *Nucleic Acids Research*, vol. 41, pp. D43–D47, 2013.

[9] T. Kenakin, "New concepts in pharmacological efficacy at 7TM receptors: IUPHAR review 2," *The British Journal of Pharmacology*, vol. 168, no. 3, pp. 554–575, 2013.

[10] M. Donizelli, M.-A. Djite, and N. le Novère, "LGICdb: a manually curated sequence database after the genomes," *Nucleic Acids Research*, vol. 34, pp. D267–D269, 2006.

[11] W. J. Gallin and P. A. Boutet, "VKCDB: voltage-gated K+ channel database updated and upgraded," *Nucleic Acids Research*, vol. 39, no. 1, pp. D362–D366, 2011.

[12] L. Fu, B. Niu, Z. Zhu, S. Wu, and W. Li, "CD-HIT: accelerated for clustering the next-generation sequencing data," *Bioinformatics*, vol. 28, no. 23, pp. 3150–3152, 2012.

[13] G. Wang and R. L. Dunbrack Jr., "PISCES: a protein sequence culling server," *Bioinformatics*, vol. 19, no. 12, pp. 1589–1591, 2003.

[14] W. Li, L. Fu, B. Niu, S. Wu, and J. Wooley, "Ultrafast clustering algorithms for metagenomic sequence analysis," *Briefings in Bioinformatics*, vol. 13, no. 6, Article ID bbs035, pp. 656–668, 2012.

[15] W. Chen and H. Lin, "Identification of voltage-gated potassium channel subfamilies from sequence information using support

vector machine," *Computers in Biology and Medicine*, vol. 42, no. 4, pp. 504–507, 2012.

[16] H. Lin and H. Ding, "Predicting ion channels and their types by the dipeptide mode of pseudo amino acid composition," *Journal of Theoretical Biology*, vol. 269, pp. 64–69, 2011.

[17] L.-X. Liu, M.-L. Li, F.-Y. Tan et al., "Local sequence information-based support vector machine to classify voltage-gated potassium channels," *Acta Biochimica et Biophysica Sinica*, vol. 38, no. 6, pp. 363–371, 2006.

[18] W. X. Liu, E. Z. Deng, W. Chen, and H. Lin, "Identifying the subfamilies of voltage-gated potassium channels using feature selection technique," *International Journal of Molecular Sciences*, vol. 15, pp. 12940–12951, 2014.

[19] S. Saha, J. Zack, B. Singh, and G. P. S. Raghava, "VGIchan: prediction and classification of voltage-gated ion channels," *Genomics, Proteomics and Bioinformatics*, vol. 4, no. 4, pp. 253–258, 2006.

[20] L. Zhu, J. Yang, J.-N. Song, K.-C. Chou, and H.-B. Shen, "Improving the accuracy of predicting disulfide connectivity by feature selection," *Journal of Computational Chemistry*, vol. 31, no. 7, pp. 1478–1485, 2010.

[21] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, article 27, 2011.

[22] T. Joachims, *Learning to classify text using support vector machines [M.S. thesis]*, Kluwer Academic Publishers, 2002.

[23] S.-H. Guo, E.-Z. Deng, L.-Q. Xu et al., "INuc-PseKNC: a sequence-based predictor for predicting nucleosome positioning in genomes with pseudo k-tuple nucleotide composition," *Bioinformatics*, vol. 30, no. 11, pp. 1522–1529, 2014.

[24] K.-C. Chou, "Some remarks on protein attribute prediction and pseudo amino acid composition," *Journal of Theoretical Biology*, vol. 273, pp. 236–247, 2011.

[25] M. R. Bakhtiarizadeh, M. Moradi-Shahrbabak, M. Ebrahimi, and E. Ebrahimie, "Neural network and SVM classifiers accurately predict lipid binding proteins, irrespective of sequence homology," *Journal of Theoretical Biology*, vol. 356, pp. 213–222, 2014.

[26] L. Lan, N. Djuric, Y. Guo, and S. Vucetic, "MS-kNN: protein function prediction by integrating multiple data sources," *BMC Bioinformatics*, vol. 14, no. 3, article S8, 2013.

[27] Z.-H. You, Y.-K. Lei, L. Zhu, J. Xia, and B. Wang, "Prediction of protein-protein interactions from amino acid sequences with ensemble extreme learning machines and principal component analysis," *BMC Bioinformatics*, vol. 14, supplement 8, article S10, 2013.

[28] M. K. Leung, H. Y. Xiong, L. J. Lee, and B. J. Frey, "Deep learning of the tissue-regulated splicing code," *Bioinformatics*, vol. 30, pp. i121–i129, 2014.

[29] B.-Q. Li, L.-L. Hu, S. Niu, Y.-D. Cai, and K.-C. Chou, "Predict and analyze S-nitrosylation modification sites with the mRMR and IFS approaches," *Journal of Proteomics*, vol. 75, no. 5, pp. 1654–1665, 2012.

[30] X. Li, H. Hu, and L. Shu, "Predicting human immunodeficiency virus protease cleavage sites in nonlinear projection space," *Molecular and Cellular Biochemistry*, vol. 339, no. 1-2, pp. 127–133, 2010.

[31] S. Zhang, F. Ye, and X. Yuan, "Using principal component analysis and support vector machine to predict protein structural class for low-similarity sequences via PSSM," *Journal of Biomolecular Structure & Dynamics*, vol. 29, no. 6, pp. 634–642, 2012.

[32] H. T. Deng and G. Runger, "Feature selection via regularized trees," in *International Joint Conference on Neural Networks (IJCNN '12)*, IEEE, 2012.

[33] J. Zhang, X. Zhao, P. Sun, and Z. Ma, "PSNO: predicting cysteine S-nitrosylation sites by incorporating various sequence-derived features into the general form of Chou's PseAAC," *International Journal of Molecular Sciences*, vol. 15, no. 7, pp. 11204–11219, 2014.

[34] C. M. Livi and E. Blanzieri, "Protein-specific prediction of mRNA binding using RNA sequences, binding motifs and predicted secondary structures," *BMC Bioinformatics*, vol. 15, no. 1, article 123, 2014.

[35] A. A. Adl, A. Nowzari-Dalini, B. Xue, V. N. Uversky, and X. Qian, "Accurate prediction of protein structural classes using functional domains and predicted secondary structure sequences," *Journal of Biomolecular Structure & Dynamics*, vol. 29, no. 6, pp. 623–633, 2012.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 740506, 4 pages
http://dx.doi.org/10.1155/2014/740506

Research Article

Prediction of DNase I Hypersensitive Sites by Using Pseudo Nucleotide Compositions

Pengmian Feng, Ning Jiang, and Nan Liu

School of Public Health, Hebei United University, Tangshan 063000, China

Correspondence should be addressed to Pengmian Feng; fengpengmian@gmail.com

Received 11 July 2014; Accepted 3 August 2014; Published 19 August 2014

Academic Editor: Hao Lin

DNase I hypersensitive sites (DHS) associated with a wide variety of regulatory DNA elements. Knowledge about the locations of DHS is helpful for deciphering the function of noncoding genomic regions. With the acceleration of genome sequences in the postgenomic age, it is highly desired to develop cost-effective computational methods to identify DHS. In the present work, a support vector machine based model was proposed to identify DHS by using the pseudo dinucleotide composition. In the jackknife test, the proposed model obtained an accuracy of 83%, which is competitive with that of the existing method. This result suggests that the proposed model may become a useful tool for DHS identifications.

1. Introduction

DNase I hypersensitive sites (DHS) are regions of chromatin which are sensitive to cleavage by the DNase I enzyme. Since the discovery of DHSs in 1980s [1], they have been used as markers of regulatory DNA regions. In general, these specific regions are generally nucleosome-free and associate with a wide variety of genomic regulatory elements, such as promoters, enhancers, insulators, silencers, and suppressors [2–4]. Therefore, mapping of DHS has become an effective approach for discovering functional DNA elements from the noncoding sequences.

Although the traditional Southern blotting technique is a gold-standard approach for identifying DHS, obtaining information from Southern blot approach is a tricky, time-consuming, and inaccurate task [5]. Recently, the DNase-seq technique (combination of DNase I digestion and high-throughput sequencing) has been proposed [6] and this technique allows for an unprecedented increase in resolution. However, methodologies for the analysis of DNase-seq data are relatively immature [7]. Therefore, computational models will be an important complement to experimental techniques for identifying DHS.

Based on nucleotide compositions, a support vector machine model for identifying DHS in K562 cell line was proposed [8]. This method yielded quite encouraging results and did play a role in stimulating the development of this area. However, further work is needed due to the following reasons. First, the sequences in their dataset share high sequence similarities. Second, the DNA structural properties were ignored. To solve these problems, we proposed a new model for identifying DHS, which is trained on a high quality benchmark dataset. In the new model, each DNA sample is encoded by using the pseudo dinucleotide composition, into which the DNA structural properties are incorporated.

2. Materials and Methods

2.1. Benchmark Dataset. The experimentally confirmed 280 DHS and 731 non-DHS sequences were obtained from http://noble.gs.washington.edu/proj/hs/, which have been used to train DHS prediction models [8]. As elucidated in [9], a predictor, if trained and tested by a dataset containing redundant samples with high similarity, might yield misleading results with an overestimated accuracy. To get rid of the redundancy and avoid bias, the CD-HIT software [10] was utilized to remove those DNA fragments that have ≥60% pairwise sequence identity to each other.

Finally, we obtained 247 positive and 710 negative samples for the benchmark dataset $\mathbb{S}$, as can be formulated by

$$\mathbb{S} = \mathbb{S}^+ \bigcup \mathbb{S}^-, \tag{1}$$

where the subset $\mathbb{S}^+$ contains 247 DHS sequences and $\mathbb{S}^-$ contains 710 non-DHS sequences, while $\bigcup$ represents the "union" in the set theory. The detailed sequences in the benchmark dataset $\mathbb{S}$ are given in Supplementary Information S1 available online at http://dx.doi.org/10.1155/2014/740506.

2.2. DNA Sequence Representation. In order to integrate the sequence-order effects and DNA physicochemical properties together, the pseudo nucleotide composition was proposed in 2011 [11]. Since then, the concept of pseudo nucleotide composition has penetrated into many branches of computational genomics, such as predicting the recombination spots [12], predicting promoters [13], predicting nucleosome positioning sequences [14], and identifying splice sites [15]. Because of its wide and increasing usage, recently, a flexible web-server, called "pseudo K-tuple nucleotide composition (PseKNC)," was developed [16], which can be used to generate various kinds of pseudo K-tuple nucleotide compositions.

Encouraged by the success of introducing pseudo nucleotide composition to computational genomics, in the current study, the pseudo dinucleotide composition was used to represent DNA sequences in the benchmark dataset, which can be expressed as [12, 16]

$$\mathbf{D} = \left[d_1 \;\; d_2 \;\; \cdots \;\; d_{16} \;\; d_{16+1} \;\; \cdots \;\; d_{16+\lambda}\right]^{\mathbf{T}}, \tag{2}$$

where

$$d_u = \begin{cases} \dfrac{f_u}{\sum_{i=1}^{16} f_i + w\sum_{j=1}^{\lambda}\theta_j}, & (1 \le u \le 16)\,, \\ \dfrac{w\theta_{u-16}}{\sum_{i=1}^{16} f_i + w\sum_{j=1}^{\lambda}\theta_j}, & (16 < u \le 16+\lambda)\,. \end{cases} \tag{3}$$

In (3), f_u ($u = 1, 2, \ldots, 16$) is the normalized occurrence frequency of the dinucleotides in the DNA sequence. λ is the number of the total counted ranks (or tiers) of the correlations along a DNA sequence, and w is the weight factor. The concrete values for λ and w as well as k will be further discussed in Section 3.1, while the correlation factor θ_j represents the j-tier structural correlation factor between all the jth most contiguous dinucleotide R_iR_{i+1} at position i.

2.3. Support Vector Machine (SVM). SVM is a supervised learning algorithm and has been widely used in computational genomics and proteomics [17–23]. The basic principle of SVM is to transform the input vector into a high dimension space and then seek a separating hyperplane with the maximal margin in this space by using the decision function

$$f\left(\vec{X}\right) = \operatorname{sgn}\left(\sum_{i=1}^{N} y_i\alpha_i \cdot K\left(\vec{X}, \vec{X}_i\right) + b\right), \tag{4}$$

where α_i is the Lagrange multipliers, b is the offset, $\vec{X}_i$ is the ith training vector, and y_i represents the type of the ith training vector. $K(\vec{X}, \vec{X}_i)$ is a kernel function which defines an inner product in a high dimensional feature space, and sgn is the sign function. Due to its effectiveness and speed in nonlinear classification process, the radial basis kernel function (RBF) $K(\vec{X}_i, \vec{X}_j) = \exp(-\gamma\|\vec{X}_i, \vec{X}_j\|^2)$ was used in the current study.

The Libsvm 2.84 package [24] was used to perform the SVM, which can be downloaded from http://www.csie.ntu.edu.tw/~cjlin/libsvm/. The regularization parameter C and the kernel width parameter γ were optimized via an optimization procedure using a grid search. The search spaces for C and γ are $[2^{15}, 2^{-5}]$ and $[2^{-5}, 2^{-15}]$ with steps of 2^{-1} and 2, respectively.

2.4. Performance Evaluation. Three cross-validation methods, that is, independent dataset test, subsampling (or K-fold cross-validation) test, and jackknife test, are often used to evaluate the anticipated success rate of a predictor. Among the three methods, the jackknife test is deemed the least arbitrary and most objective one [9, 25] and, hence, has been widely recognized and increasingly adopted by investigators to examine the quality of various predictors [26–30]. Accordingly, the jackknife test was used to examine the performance of the model proposed in the current study. In the jackknife test, each sequence in the training dataset is in turn singled out as an independent test sample and all the rule-parameters are calculated without including the one being identified.

A set of parameters, namely, sensitivity (Sn), specificity (Sp), Matthew's correlation coefficient (MCC), and accuracy (Acc), are used to evaluate the performance of the proposed model and they are defined as follows:

$$\mathrm{Sn} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{5}$$

$$\mathrm{Sp} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}, \tag{6}$$

$$\mathrm{MCC} = \frac{\mathrm{TP} \times \mathrm{TN} - \mathrm{FP} \times \mathrm{FN}}{(\mathrm{TP} + \mathrm{FN}) \times (\mathrm{TN} + \mathrm{FN}) \times (\mathrm{TP} + \mathrm{FP}) \times (\mathrm{TN} + \mathrm{FP})}, \tag{7}$$

$$\mathrm{Acc} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FN} + \mathrm{TN} + \mathrm{FP}}, \tag{8}$$

where TP, TN, FP, and FN represent the number of the correctly recognized DHS, the number of the correctly recognized non-DHS, the number of non-DHS recognized as DHS, and the number of DHS recognized as non-DHS, respectively.

3. Results and Discussions

3.1. Parameter Optimization. By analyzing the dinucleotide composition of DHS and non-DHS sequences, we found that the frequency of CC, CG, GC, and GG is higher in DHS sequences, while the frequency of the remaining dinucleotides is higher in non-DHS (Figure 1). This is self-evident as to why the pseudo dinucleotide composition was used for the current case.

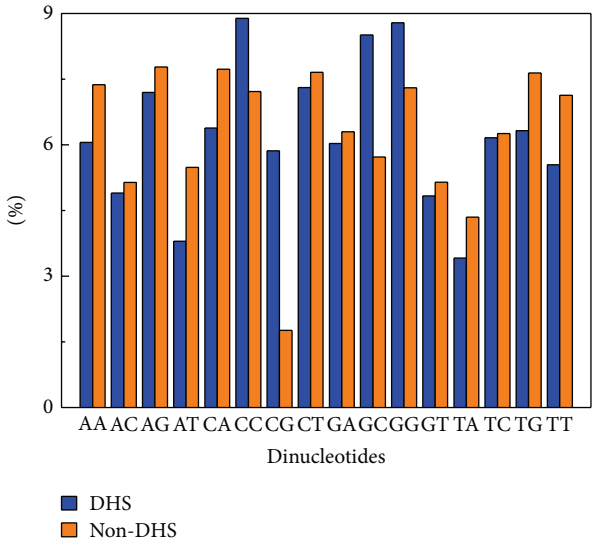


Figure 1: Comparative frequencies of 16 dinucleotides in DHS and non-DHS sequences.

A series of evidences [12, 14, 31, 32] have demonstrated that DNA local structural properties, that is, angular parameters (twist, tilt, and roll) and translational parameters (shift, slide, and rise), are effective in identifying DNA attributes. Therefore, in the present work, the six structural parameters of dinucleotides were used to calculate the pseudo dinucleotide composition by using the PseKNC web-server, which is available at http://lin.uestc.edu.cn/pseknc/default.aspx.

As we can see from (1) and (2), the present model depends on the two parameters w and λ. w is the weight factor usually within the range from 0 to 1 and λ is the global order effect. Generally speaking, the greater the λ is, the more global sequence-order information the model contains. However, if λ is too large, it would reduce the cluster-tolerant capacity so as to lower down the cross-validation accuracy due to overfitting or "high dimension disaster" problem [33]. Therefore, our searching for the optimal values of the two parameters is in the range of $w \in [0, 1]$ and $\lambda \in [1, 10]$ with the steps of 0.1 and 1, respectively.

In order to reduce the computational time, the 5-fold cross-validation approach was used to optimize the two parameters together with the parameters C and γ of the SVM. We found that when $w = 0.2$ and $\lambda = 6$ with $C = 512$ and $\gamma = 0.0078125$, a peak was observed for the Acc. Accordingly, the two numerical values were used for the two uncertain parameters in the following analysis.

3.2. Prediction Quality. The prediction quality measured by the four metrics defined in (5)–(8) for the present model in identifying DHS in the benchmark dataset $\mathbb{S}$ via the rigorous jackknife test was listed in Table 1, where, for facilitating comparison, the corresponding results obtained by the previous predictor [8] on the same benchmark data set are also given. As we can see from Table 1, the current method outperformed the existing model in all the four metrics, indicating that our proposed method may become a useful tool in identifying DHS sequences.

Table 1: Comparison of different methods for identifying DHS by the jackknife test on the same benchmark dataset.

Predictor	Sn (%)	Sp (%)	Acc (%)	MCC
Our method	72.12	86.78	83.00	0.57
Noble et al.[a]	70.43	84.23	80.12	0.52

[a] From Noble et al. [8].

4. Conclusions

Since DHS associates with a wide variety of functional elements, knowledge about the locations of DHS is helpful for deciphering the genomes. However, strong DNA sequence conservation is not observed among DHS sequences, suggesting that it is difficult to computationally identify DHS from primary DNA sequence.

A series of recent studies have demonstrated that the information coded by DNA structural properties is contributable to the identification of regulatory elements in genomes [12, 14, 31, 32]. Hence, in the present study, we proposed a SVM based model for identifying DHS by using the pseudo dinucleotide composition. In this model, we integrate dinucleotide composition with DNA structural properties. The predictive results of our model are better than existing methods. Therefore, it is anticipated that the proposed method may become a useful tool for identifying DHS sequences or, at the very least, it can play a complementary role to the existing methods in this area.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This work was supported by Foundation of Science and Technology Department of Hebei Province (no. 132777133).

References

[1] C. Wu, P. M. Bingham, K. J. Livak, R. Holmgren, and S. C. R. Elgin, "The chromatin structure of specific genes: I. Evidence for higher order domains of defined DNA sequence," *Cell*, vol. 16, no. 4, pp. 797–806, 1979.

[2] D. S. Gross and W. T. Garrard, "Nuclease hypersensitive sites in chromatin," *Annual Review of Biochemistry*, vol. 57, pp. 159–197, 1988.

[3] G. Felsenfeld and M. Groudine, "Controlling the double helix," *Nature*, vol. 421, no. 6921, pp. 448–453, 2003.

[4] G. Felsenfeld, "Chromatin as an essential part of the transcriptional mechanism," *Nature*, vol. 355, no. 6357, pp. 219–224, 1992.

[5] G. E. Crawford, I. E. Holt, J. Whittle et al., "Genome-wide mapping of DNase hypersensitive sites using massively parallel

signature sequencing (MPSS)," *Genome Research*, vol. 16, no. 1, pp. 123–131, 2006.

[6] L. Song and G. E. Crawford, "DNase-seq: a high-resolution technique for mapping active gene regulatory elements across the genome from mammalian cells," *Cold Spring Harbor Protocols*, vol. 5, no. 2, Article ID pdb.prot5384, 2010.

[7] P. Madrigal and P. Krajewski, "Current bioinformatic approaches to identify DNase I hypersensitive sites and genomic footprints from DNase-seq data," *Frontiers in Genetics*, vol. 3, article 230, 2012.

[8] W. S. Noble, S. Kuehn, R. Thurman, M. Yu, and J. Stamatoyannopoulos, "Predicting the in vivo signature of human gene regulatory sequences," *Bioinformatics*, vol. 21, no. 1, pp. i338–i343, 2005.

[9] K. Chou, "Some remarks on protein attribute prediction and pseudo amino acid composition," *Journal of Theoretical Biology*, vol. 273, pp. 236–247, 2011.

[10] L. Fu, B. Niu, Z. Zhu, S. Wu, and W. Li, "CD-HIT: accelerated for clustering the next-generation sequencing data," *Bioinformatics*, vol. 28, no. 23, pp. 3150–3152, 2012.

[11] X. Zhou, Z. Li, Z. Dai, and X. Zou, "Predicting methylation status of human DNA sequences by pseudo-trinucleotide composition," *Talanta*, vol. 85, no. 2, pp. 1143–1147, 2011.

[12] W. Chen, P. Feng, H. Lin, and K. Chou, "IRSpot-PseDNC: identify recombination spots with pseudo dinucleotide composition," *Nucleic Acids Research*, vol. 41, no. 6, article e68, 2013.

[13] X. Zhou, Z. Li, Z. Dai, and X. Zou, "Predicting promoters by pseudo-trinucleotide compositions based on discrete wavelets transform," *Journal of Theoretical Biology*, vol. 319, pp. 1–7, 2013.

[14] S. H. Guo, E. Z. Deng, L. Q. Xu et al., "iNuc-PseKNC: a sequence-based predictor for predicting nucleosome positioning in genomes with pseudo k-tuple nucleotide composition," *Bioinformatics*, vol. 30, no. 11, pp. 1522–1529, 2014.

[15] W. Chen, P. M. Feng, H. Lin, and K. C. Chou, "iSS-PseDNC: identifying splicing sites using pseudo dinucleotide composition," *BioMed Research International*, vol. 2014, Article ID 623149, 12 pages, 2014.

[16] W. Chen, T. Y. Lei, D. C. Jin, H. Lin, and K. C. Chou, "PseKNC: a flexible web server for generating pseudo K-tuple nucleotide composition," *Analytical Biochemistry*, vol. 456, pp. 53–60, 2014.

[17] W. Chen and H. Lin, "Prediction of midbody, centrosome and kinetochore proteins based on gene ontology information," *Biochemical and Biophysical Research Communications*, vol. 401, no. 3, pp. 382–384, 2010.

[18] H. Lin and H. Ding, "Predicting ion channels and their types by the dipeptide mode of pseudo amino acid composition," *Journal of Theoretical Biology*, vol. 269, no. 1, pp. 64–69, 2011.

[19] B. Liu, X. Wang, Q. Chen, Q. Dong, and X. Lan, "Using Amino Acid Physicochemical Distance Transformation for Fast Protein Remote Homology Detection," *PLoS ONE*, vol. 7, no. 9, Article ID e46633, 2012.

[20] B. Liu, X. Wang, L. Lin, B. Tang, and Q. Dong, "Prediction of protein binding sites in protein structures using hidden Markov support vector machine," *BMC Bioinformatics*, vol. 10, article 381, 2009.

[21] B. Liu, X. Wang, L. Lin, Q. Dong, and X. Wang, "Exploiting three kinds of interface propensities to identify protein binding sites," *Computational Biology and Chemistry*, vol. 33, no. 4, pp. 303–311, 2009.

[22] K. C. Chou and Y. D. Cai, "Using functional domain composition and support vector machines for prediction of protein subcellular location," *The Journal of Biological Chemistry*, vol. 277, no. 48, pp. 45765–45769, 2002.

[23] M. Hayat and A. Khan, "MemHyb: predicting membrane protein types by hybridizing SAAC and PSSM," *Journal of Theoretical Biology*, vol. 292, pp. 93–102, 2012.

[24] C. C. Chang and C. J. Lin, "LIBSVM: a library for support vector machines," http://www.csie.ntu.edu.tw/~cjlin/libsvm/.

[25] K.-C. Chou and C.-T. Zhang, "Prediction of protein structural classes," *Critical Reviews in Biochemistry and Molecular Biology*, vol. 30, no. 4, pp. 275–349, 1995.

[26] M. Esmaeili, H. Mohabatkar, and S. Mohsenzadeh, "Using the concept of Chou's pseudo amino acid composition for risk type prediction of human papillomaviruses," *Journal of Theoretical Biology*, vol. 263, no. 2, pp. 203–209, 2010.

[27] C. Ding, L. F. Yuan, S. H. Guo, H. Lin, and W. Chen, "Identification of mycobacterial membrane proteins and their types using over-represented tripeptide compositions," *Journal of Proteomics*, vol. 77, pp. 321–328, 2012.

[28] W. Chen and H. Lin, "Identification of voltage-gated potassium channel subfamilies from sequence information using support vector machine," *Computers in Biology and Medicine*, vol. 42, no. 4, pp. 504–507, 2012.

[29] K. Chou, Z. Wu, and X. Xiao, "iLoc-Euk: a multi-label classifier for predicting the subcellular localization of singleplex and multiplex eukaryotic proteins," *PLoS ONE*, vol. 6, no. 3, Article ID e18258, 2011.

[30] H. Mohabatkar, M. Mohammad Beigi, and A. Esmaeili, "Prediction of GABAA receptor proteins using the concept of Chou's pseudo-amino acid composition and support vector machine," *Journal of Theoretical Biology*, vol. 281, no. 1, pp. 18–23, 2011.

[31] Y. Zuo and Q. Li, "Identification of TATA and TATA-less promoters in plant genomes by integrating diversity measure, GC-Skew and DNA geometric flexibility," *Genomics*, vol. 97, no. 2, pp. 112–120, 2011.

[32] J. R. Goñi, A. Pérez, D. Torrents, and M. Orozco, "Determining promoter location based on DNA structure first-principles calculations," *Genome Biology*, vol. 8, no. 12, article R263, 2007.

[33] T. Wang, J. Yang, H. Shen, and K. Chou, "Predicting membrane protein types by the LLDA algorithm," *Protein and Peptide Letters*, vol. 15, no. 9, pp. 915–921, 2008.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 464093, 6 pages
http://dx.doi.org/10.1155/2014/464093

Research Article

Protein Binding Site Prediction by Combining Hidden Markov Support Vector Machine and Profile-Based Propensities

Bin Liu,[1,2] Bingquan Liu,[3] Fule Liu,[1] and Xiaolong Wang[1,2]

[1] *School of Computer Science and Technology, Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, Guangdong 518055, China*
[2] *Key Laboratory of Network Oriented Intelligent Computation, Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, Guangdong 518055, China*
[3] *School of Computer Science and Technology, Harbin Institute of Technology, Harbin, Heilongjiang 150001, China*

Correspondence should be addressed to Bin Liu; bliu@insun.hit.edu.cn and Bingquan Liu; liubq@insun.hit.edu.cn

Received 4 June 2014; Accepted 1 July 2014; Published 14 July 2014

Academic Editor: Wei Chen

Identification of protein binding sites is critical for studying the function of the proteins. In this paper, we proposed a method for protein binding site prediction, which combined the order profile propensities and hidden Markov support vector machine (HM-SVM). This method employed the sequential labeling technique to the field of protein binding site prediction. The input features of HM-SVM include the profile-based propensities, the Position-Specific Score Matrix (PSSM), and Accessible Surface Area (ASA). When tested on different data sets, the proposed method showed promising results, and outperformed some closely relative methods by more than 10% in terms of AUC.

1. Introduction

Prediction of protein binding sites provides valuable information for studying the function of proteins. The most efficient approaches are the computational methods. By using these approaches, the functionally important amino acid residues can be identified [1].

These computational methods used different features extracted from protein sequences, PSSM, or structure information. Hydrophobic and polar residues tend to occur in protein binding regions [2, 3]. The conservation scores of amino acid are often used as features, because the protein binding sites are more conserved than other surface residues [4]. Some kinds of conservation scores were proposed; a comprehensive evaluation of these scores was reported in [5]. One of the most widely used features is the Accessible Surface Area (ASA) [4], because the binding sites show higher ASA values than those of the other surface residues [6].

Some machine learning methods treated protein binding site prediction as a binary classification task, and some well-known machine learning techniques have been applied to this field, such as support vector machine [7, 8], neural network [1], Bayesian network [9], and hidden Markov model [10]. A comparison of these methods has been performed by Zhou and Qin [11].

In our previous study [12], we introduced a novel profile-level propensity for protein binding site prediction. Experimental results showed that this propensity can significantly improve the performance of the SVM based methods. Recently, we applied hidden Markov support vector machine (HM-SVM) to this field [13], which takes protein binding site prediction as a sequence-labeling task. The advantage of this method is that it is able to incorporate the sequence-order effects into the predictor. However, this method only uses two basic features (PSSM and ASA features) as input for protein binding site prediction. Therefore, it is interesting to explore whether the order profile propensity can improve the performance of HM-SVM based method or not. In this study, we proposed a computational method for protein binding site prediction by combining the hidden Markov support vector machine and the order profile propensity. When tested on six different data sets, the HM-SVM predictor using order profile propensity as an extra feature consistently outperformed the predictor only using two basic features (PSSM and ASA

Table 1: Summary of six data sets.

Data set	Chains	Res.	Surface res.	Interface res.
Heterocomplex I[a]	504	109829	92797	26085
Homocomplex I	620	172917	141295	38170
Mix[b] I	1124	282746	234092	64255
Heterocomplex II[c]	504	109829	92797	32386
Homocomplex II	620	172917	141295	45633
Mix II	1124	282746	234092	78019

[a]Type I data set with minor interface as negative samples.
[b]The mixed data set of heterocomplexes and homocomplexes.
[c]Type II data set with minor interface as positive samples.

features); in particular, in terms of AUC, the performance is improved by more than 10 percent, indicating that combining the order profile propensity and the HM-SVM is a suitable approach to improve the accuracy of protein binding site prediction.

2. Methods

2.1. Dataset Description. The datasets used in this study have been described in the study [13]. 1124 protein chains were selected from the Protein Data Bank (PDB) [14]. The chains were divided into six types of datasets according to homology of interacting chains and the definition of the interface. The information of the six datasets is shown in Table 1, and the process of dataset preparation is shown in the left part of Figure 1. The six datasets can be downloaded from http://bioinformatics.hitsz.edu.cn/HMSVM-OP.

2.2. Feature Description

2.2.1. Order Profile Propensity. The detailed information of how to calculate the order profile propensity was introduced in study [12]. Here we only briefly introduce this process. The order profile propensities were profile-based features, which extracted the evolutionary information from frequency profiles. The frequency profiles were calculated from the multiple sequence alignments outputted by running the PSI-BLAST software [13] searching against the nrdb90 database from EBI [15] with parameters of $j = 10$ and $e = 0.001$. The frequency profiles were converted into order profiles by combining the amino acids whose frequencies were higher than a given threadhold optimized on the benchmark dataset. Order profile can be viewed as a profile-based building block of proteins, which has been used for many tasks in the field of bioinformatics [12, 16].

The order profile propensity was based on the order profile occurrence differences between protein binding regions and other surface regions. The equations of how to calculate this feature were given by [12, Equations (3)–(5)].

2.2.2. Position-Specific Score Matrix (PSSM). PSSM was another profile-based feature, which was generated by using PSI-BLAST [13] with the parameters j and e set as 10 and 0.001, respectively.

2.2.3. Accessible Surface Area (ASA). We employed the DSSP program [17] to calculate the Accessible Surface Area (ASA) features, which were scaled by the nominal maximum area of each residue.

2.3. Hidden Markov Support Vector Machine. Hidden Markov support vector machine proposed by Altun et al. [15] was a sequential labelling model. In our previous study [13], it showed that when using the two basic features (PSSM and ASA features), the HM-SVM based method outperformed other machine learning methods, such as SVM, CRF, and ANN. In this study, we explored new features to improve the performance of HM-SVM based methods. For more information of HM-SVM, please refer to this paper [13].

The flowchart of the proposed computational method for protein binding site prediction was shown in Figure 1, in which the left part shows the process of dataset construction, and the right part shows the prediction process of the model based on HM-SVM.

In this paper, SVM^{hmm} toolkit (V3.10) was employed as the software of HM-SVM model with parameters c and e set as 0.1 and 1, respectively. This parameter combination was optimized with the training data. The input features of HM-SVM include order profile propensity, ASA, and PSSM. These features were extracted from the target residues and its 6 neighbouring residues in each direction.

2.4. Evaluation Methodology. The sensitivity (Sn), specificity (Sp), overall accuracy (Acc), F1 measure (F1), Matthews correlation coefficient MCC, and AUC can be, respectively, expressed as [18–22]

$$\mathrm{Sn} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}},$$

$$\mathrm{Sp} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}},$$

$$\mathrm{Acc} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}},$$

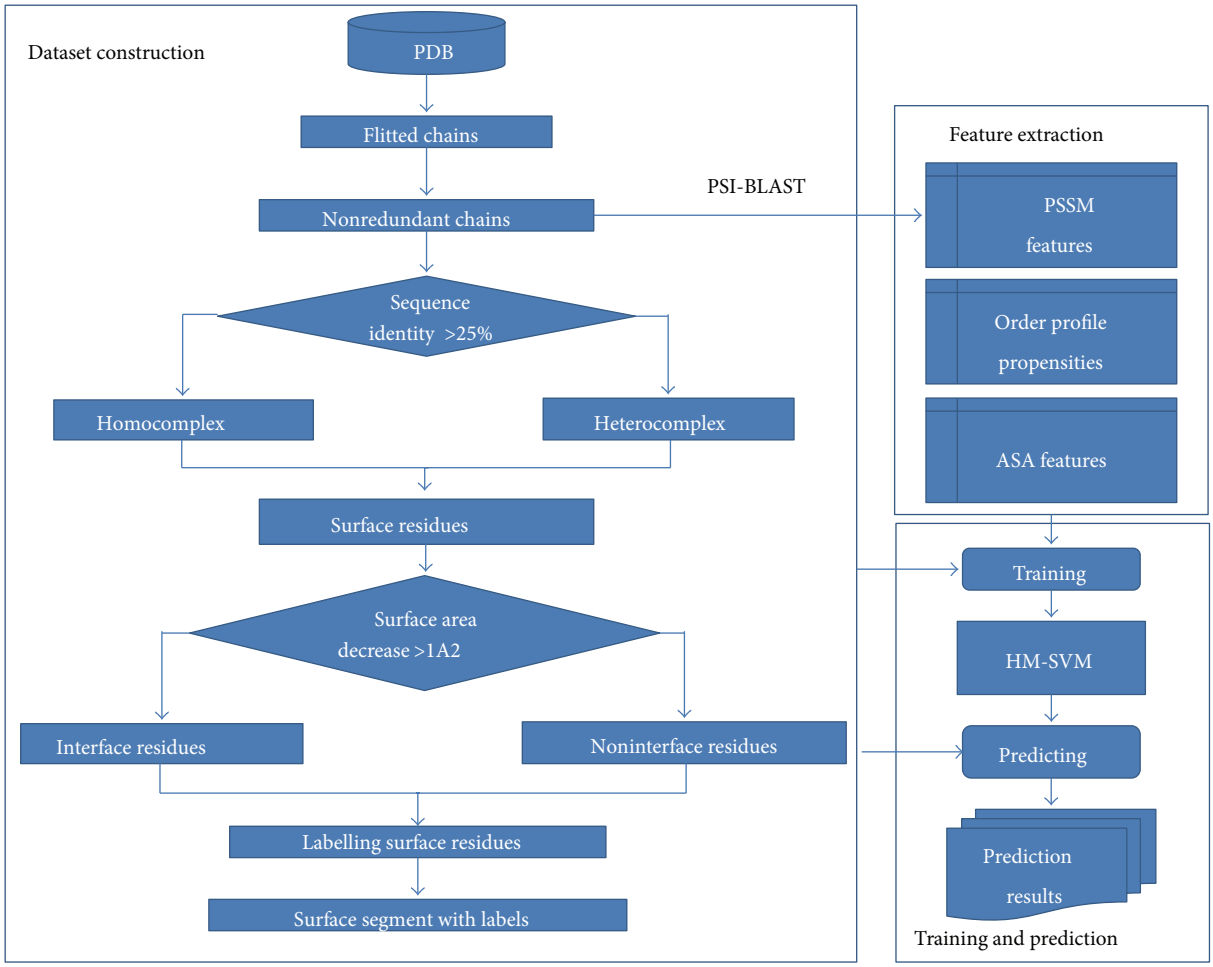


Figure 1: Overview of the proposed framework for protein binding site prediction.

$$\text{MCC} = \frac{(\text{TP} * \text{TN}) - (\text{FN} * \text{FP})}{\sqrt{(\text{TP} + \text{FN}) * (\text{TN} + \text{FP}) * (\text{TP} + \text{FP}) * (\text{TN} + \text{FN})}},$$
$$\text{AUC}: \text{ the area under ROC cure,} \quad (1)$$

where TP represents the true positive, TN represents the true negative, FN represents the false negative, and FP represents the false positive.

3. Results

In order to validate whether the order profile propensities can improve the performance of the HM-SVM based methods or not, two HM-SVM predictors with different features were constructed. The first HM-SVM employed the PSSMs and ASA as input features. This predictor was treated as a baseline predictor. For the second HM-SVM predictor, order profile propensity is added as an extra feature to evaluate whether this feature can improve the performance or not. The performance of the two HM-SVM predictors was evaluated by fivefold cross-validation.

The results of the two HM-SVM predictors on the six datasets are shown in Table 2. It can be seen that the first HM-SVM predictor using the two basic features achieved the lowest performance. The second HM-SVM predictor using the order profile propensity as an extra feature achieved the best performance on all the six data sets, especially its AUC score being about 10% higher than that of the first HM-SVM predictor, indicating that order profile propensity can significantly improve the performance of the HM-SVM based methods. In our previous study [13], we showed that the first HM-SVM predictor outperformed some state-of-the-art methods, such as ANN, CRF, and SVM. The second HM-SVM predictor significantly outperformed the first HM-SVM predictor, indicating that the proposed computational method for protein binding site prediction is a good method in this field.

Šikić et al. [23] proposed a method based on random forest, which was evaluated on a heterocomplex data set, and achieved good performance (Sp = 76.45%, Sn = 38.06%, F1 = 50.82%, and Acc = 80.05%). Our method (results of heterocomplex II dataset) outperformed this method by 14.98% in terms of F1, which further confirms the better performance of our method than some state-of-the-art methods.

Table 2: Performance of HM-SVM based method with and without order profile propensities.

Dataset	Method	Sp %	Sn %	F1 %	Acc %	MCC	AUC %
Heterocomplex I	HM-SVM 1[a]	44.9	56.0	49.8	68.3	0.274	69.5
	HM-SVM 2[b]	**52.4**	**73.5**	**61.2**	**73.8**	**0.436**	**81.4**
Homocomplex I	HM-SVM 1	45.4	60.0	51.70	69.7	0.309	72.2
	HM-SVM 2	**54.5**	**74.6**	**62.9**	**76.3**	**0.474**	**83.6**
Mix I	HM-SVM 1	45.5	58.0	51.0	69.4	0.297	71.2
	HM-SVM 2	**53.5**	**74.0**	**62.1**	**75.0**	**0.455**	**82.5**
Heterocomplex II	HM-SVM 1	54.0	56.7	55.3	68.0	0.305	70.7
	HM-SVM 2	**60.8**	**71.7**	**65.8**	**74.0**	**0.454**	**81.2**
Homocomplex II	HM-SVM 1	53.3	60.1	56.5	70.1	0.340	73.4
	HM-SVM 2	**61.1**	**73.8**	**66.8**	**76.4**	**0.493**	**83.7**
Mix II	HM-SVM 1	53.6	58.6	56.0	69.3	0.326	72.4
	HM-SVM 2	**61.0**	**72.7**	**66.3**	**75.2**	**0.474**	**82.4**

[a]Results of HM-SVM 1 on the six data sets are obtained from [13]. HM-SVM 1 represents the HM-SVM predictor with the basic feature set using PSSM and ASA features; [b]HM-SVM 2 represents the HM-SVM predictor with the feature set using PSSM, ASA, and order profile propensity features.

4. Conclusion

In this study, we proposed a computational method for protein binding site prediction, which combines the order profile propensity and hidden Markov support vector machine. This method predicts the protein binding sites with a sequential labelling approach and uses a recently proposed feature to further improve the performance: order profile propensity, which contains the evolutionary information extracted from the sequence profiles. The main contribution of this study is that we validate the fact that order profile propensity can significantly improve the performance of the HM-SVM based method. The main advantage of the proposed method is that it treats the protein sequence as a whole and is able to use the label information of neighbour residues and the evolutionary information extracted from the frequency profiles. However, the order profile propensity was generated based on the frequency profiles, which require the computational expensive multiple sequences alignment process. It is the main disadvantage of the proposed method.

As noted by Li et al. [24], choosing proper features is a challenging task, especially for sequential labelling method, such as HM-SVM and conditional random field (CRF). In their experiments, the authors found that by simply adding some features into CRF cannot improve the performance of their method. Therefore, the obvious performance improvement when using order profile propensity as an extra feature will benefit our future studies, especially for the research on applying sequential method to this field. As pointed out in a comprehensive review and carried out in a series of recent publications [25–43], finding suitable features is the key step to improve the performance.

Furthermore, since user-friendly and publicly accessible web servers represent the future direction for developing practically more useful predictors [44, 45], we shall make efforts in our future work to provide a web server for the method presented in this paper.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 61300112, 61272383), the Natural Science Foundation of Guangdong Province (no. S2012040007390), the Scientific Research Innovation Foundation in Harbin Institute of Technology (Project no. HIT.NSRIF.201310b3), the Shanghai Key Laboratory of Intelligent Information Processing, China (Grant no. IIPL-2012-002), the Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry, Strategic Emerging Industry Development Special Funds of Shenzhen (ZDSY20120613125401420 and JCYJ2012 0613151940045), and Key Basic Research Foundation of Shenzhen (JC201005260118A, JC201005260175A).

References

[1] A. Porollo and J. Meller, "Prediction-based fingerprints of protein-protein interactions," *Proteins: Structure, Function and Genetics*, vol. 66, no. 3, pp. 630–645, 2007.

[2] F. Glaser, D. M. Steinberg, I. A. Vakser, and N. Ben-Tal, "Residue frequencies and pairing preferences at protein-protein interfaces," *Proteins*, vol. 43, no. 2, pp. 89–102, 2001.

[3] W. L. DeLano, "Unraveling hot spots in binding interfaces: progress and challenges," *Current Opinion in Structural Biology*, vol. 12, no. 1, pp. 14–20, 2002.

[4] H. Zhou and Y. Shan, "Prediction of protein interaction sites from sequence profile and residue neighbor list," *Proteins: Structure, Function and Genetics*, vol. 44, no. 3, pp. 336–343, 2001.

[5] W. S. J. Valdar, "Scoring residue conservation," *Proteins: Structure, Function and Genetics*, vol. 48, no. 2, pp. 227–241, 2002.

[6] H. Chen and H. X. Zhou, "Prediction of interface residues in protein-protein complexes by a consensus neural network method: test against NMR data," *Proteins: Structure, Function and Genetics*, vol. 61, no. 1, pp. 21–35, 2005.

[7] A. Koike and T. Takagi, "Prediction of protein-protein interaction sites using support vector machines," *Protein Engineering, Design and Selection*, vol. 17, no. 2, pp. 165–173, 2004.

[8] B. Wang, P. Chen, D. Huang, J. Li, T. Lok, and M. R. Lyu, "Predicting protein interaction sites from residue spatial sequence profile and evolution rate," *FEBS Letters*, vol. 580, no. 2, pp. 380–384, 2006.

[9] J. R. Bradford, C. J. Needham, A. J. Bulpitt, and D. R. Westhead, "Insights into protein-protein Interfaces using a Bayesian network prediction method," *Journal of Molecular Biology*, vol. 362, no. 2, pp. 365–386, 2006.

[10] T. Friedrich, B. Pils, T. Dandekar, J. Schultz, and T. Müller, "Modelling interaction sites in protein domains with interaction profile hidden Markov models," *Bioinformatics*, vol. 22, no. 23, pp. 2851–2857, 2006.

[11] H. Zhou and S. Qin, "Interaction-site prediction for protein complexes: a critical assessment," *Bioinformatics*, vol. 23, no. 17, pp. 2203–2209, 2007.

[12] B. Liu, X. Wang, L. Lin, Q. Dong, and X. Wang, "Exploiting three kinds of interface propensities to identify protein binding sites," *Computational Biology and Chemistry*, vol. 33, no. 4, pp. 303–311, 2009.

[13] B. Liu, X. Wang, L. Lin, B. Tang, and Q. Dong, "Prediction of protein binding sites in protein structures using hidden Markov support vector machine," *BMC Bioinformatics*, vol. 10, article 381, 2009.

[14] A. Kouranov, L. Xie, J. de la Cruz et al., "The RCSB PDB information portal for structural genomics," *Nucleic Acids Research.*, vol. 34, pp. D302–D305, 2006.

[15] Y. Altun, I. Tsochantaridis, and T. Hofmann, "Hidden markov support vector machines," in *Proceedings of the 20th International Conference on Machine Learning (ICML '03)*, pp. 3–10, August 2003.

[16] B. Liu, L. Lin, and X. Wang, "Protein remote homology detection using order profiles," in *Proceedings of the International Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing (IJCBS '09)*, pp. 255–260, Shanghai, China, August 2009.

[17] W. Kabsch and C. Sander, "Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features," *Biopolymers*, vol. 22, no. 12, pp. 2577–2637, 1983.

[18] B. Liu, X. Wang, Q. Chen, Q. Dong, and X. Lan, "Using amino acid physicochemical distance transformation for fast protein remote homology detection," *PLoS ONE*, vol. 7, no. 9, Article ID e46633, 2012.

[19] B. Liu, X. Wang, L. Lin, and Q. Dong, "A discriminative method for protein remote homology detection and fold recognition combining Top-n-grams and latent semantic analysis," *BMC Bioinformatics*, vol. 9, article 510, 2008.

[20] B. Liu, X. Wang, Q. Zou, Q. Dong, and Q. Chen, "Protein remote homology detection by combining Chou's pseudo amino acid composition and profile-based protein representation," *Molecular Informatics*, vol. 32, no. 9-10, pp. 775–782, 2013.

[21] B. Liu, J. Xu, Q. Zou et al., "Using distances between Top-n-gram and residue pairs for protein remote homology detection," *BMC Bioinformatics*, vol. 15, supplement 2, p. S3, 2014.

[22] B. Liu, "Combining evolutionary information extracted from frequency profiles with sequence-based kernels for protein remote homology detection," *Bioinformatics*, vol. 30, no. 4, pp. 472–479, 2014.

[23] M. Šikić, S. Tomić, and K. Vlahoviček, "Prediction of protein-protein interaction sites in sequences and 3D structures by random forests," *PLoS Computational Biology*, vol. 5, no. 1, Article ID e1000278, 2009.

[24] M. H. Li, L. Lin, X. Wang, and T. Liu, "Protein-protein interaction site prediction based on conditional random fields," *Bioinformatics*, vol. 23, no. 5, pp. 597–604, 2007.

[25] B. Liu, "QChIPat: a quantitative method to identify distinct binding patterns for two biological ChIP-seq samples in different experimental conditions," *BMC Genomics*, vol. 14, supplement 8, p. S3, 2013.

[26] W. Chen, P. Feng, and H. Lin, "Prediction of ketoacyl synthase family using reduced amino acid alphabets," *Journal of Industrial Microbiology and Biotechnology*, vol. 39, no. 4, pp. 579–584, 2012.

[27] H. Lin, W. Chen, and H. Ding, "AcalPred: a sequence-based tool for discriminating between acidic and alkaline enzymes," *PLoS ONE*, vol. 8, no. 10, Article ID e75726, 2013.

[28] W. Chen, H. Lin, and P. M. Feng, "DNA physical parameters modulate nucleosome positioning in the *Saccharomyces cerevisiae* genome," *Current Bioinformatics*, vol. 9, no. 2, pp. 188–193, 2014.

[29] P.-M. Feng, H. Lin, and W. Chen, "Identification of antioxidants from sequence information using Naïve Bayes," *Computational and Mathematical Methods in Medicine*, vol. 2013, Article ID 567529, 5 pages, 2013.

[30] C. Ding, L. Yuan, S. Guo, H. Lin, and W. Chen, "Identification of mycobacterial membrane proteins and their types using over-represented tripeptide compositions," *Journal of Proteomics*, vol. 77, pp. 321–328, 2012.

[31] W. Chen and H. Lin, "Identification of voltage-gated potassium channel subfamilies from sequence information using support vector machine," *Computers in Biology and Medicine*, vol. 42, no. 4, pp. 504–507, 2012.

[32] P. M. Feng, W. Chen, H. Lin, and K. Chou, "iHSP-PseRAAAC: identifying the heat shock protein families using pseudo reduced amino acid alphabet composition," *Analytical Biochemistry*, vol. 442, no. 1, pp. 118–125, 2013.

[33] W. Chen, H. Lin, P. Feng, C. Ding, Y. Zuo, and K. Chou, "iNuc-PhysChem: a sequence-based predictor for identifying nucleosomes via physicochemical properties," *PLoS ONE*, vol. 7, no. 10, Article ID e47843, 2012.

[34] S. H. Guo, E. Z. Deng, L. Q. Xu et al., "iNuc-PseKNC: a sequence-based predictor for predicting nucleosome positioning in genomes with pseudo k-tuple nucleotide composition," *Bioinformatics*, vol. 30, no. 11, pp. 1522–1529, 2014.

[35] W. Chen, P. Feng, H. Lin, and K. Chou, "IRSpot-PseDNC: identify recombination spots with pseudo dinucleotide composition," *Nucleic Acids Research*, vol. 41, no. 6, p. e68, 2013.

[36] P. Feng, H. Ding, W. Chen, and H. Lin, "Naïve bayes classifier with feature selection to identify phage virion proteins," *Computational and Mathematical Methods in Medicine*, vol. 2013, Article ID 530696, 6 pages, 2013.

[37] H. Ding, S. Guo, E. Deng et al., "Prediction of Golgi-resident protein types by using feature selection technique," *Chemometrics and Intelligent Laboratory Systems*, vol. 124, pp. 9–13, 2013.

[38] W. Chen and H. Lin, "Prediction of midbody, centrosome and kinetochore proteins based on gene ontology information," *Biochemical and Biophysical Research Communications*, vol. 401, no. 3, pp. 382–384, 2010.

[39] H. Lin, H. Ding, F. Guo, and J. Huang, "Prediction of subcellular location of mycobacterial protein using feature selection techniques," *Molecular Diversity*, vol. 14, no. 4, pp. 667–671, 2010.

[40] H. Lin and W. Chen, "Prediction of thermophilic proteins using feature selection technique," *Journal of Microbiological Methods*, vol. 84, no. 1, pp. 67–70, 2011.

[41] W. Chen, T. Y. Lei, D. C. Jin, H. Lin, and K. C. Chou, "PseKNC: a flexible web server for generating pseudo K-tuple nucleotide composition," *Analytical Biochemistry*, vol. 456, pp. 53–60, 2014.

[42] H. Lin, W. Chen, L. Yuan, Z. Li, and H. Ding, "Using over-represented tetrapeptides to predict protein submitochondria locations," *Acta Biotheoretica*, vol. 61, no. 2, pp. 259–268, 2013.

[43] Q. Zou, W. Chen, Y. Huang, X. Liu, and Y. Jiang, "Identifying multi-functional enzyme by hierarchical multi-label classifier," *Journal of Computational and Theoretical Nanoscience*, vol. 10, no. 4, pp. 1038–1043, 2013.

[44] Q. Zou, X. Li, Y. Jiang, Y. Zhao, and G. Wang, "Binmempredict: a web server and software for predicting membrane protein types," *Current Proteomics*, vol. 10, no. 1, pp. 2–9, 2013.

[45] C. Lin, Y. Zou, J. Qin et al., "Hierarchical classification of protein folds using a novel ensemble classifier," *PLoS ONE*, vol. 8, no. 2, Article ID e56499, 2013.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 864135, 9 pages
http://dx.doi.org/10.1155/2014/864135

Research Article

acACS: Improving the Prediction Accuracy of Protein Subcellular Locations and Protein Classification by Incorporating the Average Chemical Shifts Composition

Guo-Liang Fan,[1] Yan-Ling Liu,[1] Yong-Chun Zuo,[2] Han-Xue Mei,[1] Yi Rang,[1] Bao-Yan Hou,[1] and Yan Zhao[1]

[1] *Department of Physics, School of Physical Science and Technology, Inner Mongolia University, Hohhot 010021, China*
[2] *The Key Laboratory of Mammalian Reproductive Biology and Biotechnology of the Ministry of Education, College of Life Sciences, Inner Mongolia University, Hohhot 010021, China*

Correspondence should be addressed to Guo-Liang Fan; eeguoliangfan@sina.com and Yong-Chun Zuo; yczuo@imu.edu.cn

Received 19 May 2014; Revised 15 June 2014; Accepted 16 June 2014; Published 2 July 2014

Academic Editor: Hao Lin

The chemical shift is sensitive to changes in the local environments and can report the structural changes. The structure information of a protein can be represented by the average chemical shifts (ACS) composition, which has been broadly applied for enhancing the prediction accuracy in protein subcellular locations and protein classification. However, different kinds of ACS composition can solve different problems. We established an online web server named acACS, which can convert secondary structure into average chemical shift and then compose the vector for representing a protein by using the algorithm of auto covariance. Our solution is easy to use and can meet the needs of users.

1. Introduction

Knowledge of subcellular localization information of a protein may help to unravel its normal cellular function [1]. The proteins within the different compartments have different biological activity and functions; in turn, knowing the subcellular localization of a given protein helps in elucidating its functional role.

Recently, many computational approaches for subcellular localization predictions have been developed and plenty of methods for improving the accuracy of the prediction were applied. From two aspects the predictor can be described. One is the predicting algorithms, like support vector machine (SVM) [2–11], neural network [12], increment of diversity (ID) [13], random forest (RF) [14], K-nearest neighbor (K-NN) [15, 16], generating algorithm [17], and so on, or the combination of them [16, 18]. The other is the information source, such as widely used sequence-based information source, which are amino acid composition (AAC) and sorting signals [19–21], and textual descriptions of proteins [22, 23], which are protein physiochemical property [24], gene ontology (GO) [25], and so on. Actually, the structure information of a protein is very important, especially when it is used for representing the subcellular locations of a protein. However, the structure information of a protein cannot be easily described, and few methods using the structure information can be learned to our knowledge.

However, in NMR spectroscopy, as an important parameter, chemical shift, which is sensitive to changes in the local environments, can report the structural changes. Sibley et al. [26], Mielke and Krishnan [27], Spera and Bax [28], and Zhao et al. [29] have found that the ACS of a protein has intrinsic correlation with the protein's secondary structure and the function of this protein is determined by its structure. According to this point of view, there must be some relationship among the averaged chemical shift, protein structure, and functions [30, 31]. Wishart has developed a web server, namely, CS23D, for rapidly generating accurate 3D protein structures using only assigned NMR chemical shifts [32]. More than 100 proteins from BMRB [33] were tested

and found that the resulting structures generally exhibit good geometry and chemical shift agreement [32]. Also, there are some algorithms, which can predict the chemical shift from protein sequences and conformation [34–37]; few works have been done to determine a protein's functions by the chemical shifts [38, 39]. Therefore, how to use the chemical shift is still important and urgent.

In this paper, a benchmark data set of chemical shift was constructed, which consists of 1,552 proteins derived from BMRB website [33] and then extracted chemical shift values of ^{15}N, $^{13}C_\alpha$, $^{1}H_\alpha$, and $^{1}H_N$ for 20 amino acid residues. Then four types of average chemical shift for 20 amino acid residues were calculated and the autocovariance algorithm was used to convert the average chemical shift into the vector to describe the protein sample. The algorithm acACS (autocovariance of averaged chemical shifts) has been used to enhance the prediction accuracy in protein subcellular locations. The proposed acACS descriptor can be considered as a mode of generalized pseudoamino acid composition, which was summarized in [40]. Recently, the generalized pseudoamino acid composition methods have been systematically implemented by two powerful software, PseAAC-Builder [41] and PseAAC-General [42]. For the readers' convenience in using the current method, the acACS descriptor may be integrated into this software in future works. The details of how to deal with this calculation and how to use this method is shown as follows.

2. Material and Methods

2.1. Data Sets. When an electron moves around a proton, it will produce some magnetic field, which could affect proton's external electron field. Thus, the absorption frequencies of proton in different chemical environments would shift relatively to the absorption frequencies under standard magnetic fields. Chemical shift is the relative resonance frequencies shift of protons between different chemical environment and standard, which can be measured by NMR spectroscopy. Due to its sensitivity to local environments, such as the backbone dihedral angles and the secondary structure types [26, 27, 29], chemical shift can be an indicator for the changes of local conformations.

In order to find out the correlation between chemical shift and the secondary structure of a protein, we construct a high-quality working data set, which started from the following steps: (1) the proteins star file with NMR spectroscopy data were downloaded from BMRB [33]; (2) the proteins less than 50 residues or not matched to PDB [43] entries were discarded; (3) the proteins with sequence identity higher than 40% were excluded by CD-HIT [44]. Finally, the benchmark data set has 1,552 proteins. The data set was available at our website. The data set contained 1,552 proteins sequences and BMRB star file, which was the original chemical shifts data file for all kinds of backbone atoms of each protein. We analyzed the averaged chemical shifts for every kind of amino acids type and secondary structure in order to find out the rules among averaged chemical shifts with every kind of amino acids type and secondary structure types and then used the autocovariance algorithm to calculate the feature vectors of the protein sequences from the statistic results. The feature vectors representing the protein sequences can be used in problems of subcellular location prediction or other protein classifications. Researchers may also develop better algorithms for protein representation using the data set.

2.2. Averaged Chemical Shift (ACS). In order to find the rule between the chemical shifts and structure information, the statistic about averaged chemical shift related to secondary structure and amino acids type was carried out.

Firstly, four types chemical shift values ω of ^{15}N, $^{13}C_\alpha$, $^{1}H_\alpha$, and $^{1}H_N$ from every amino acid residue were extracted from the BMRB star file for further calculation. In the BMRB star file, the amino acid residues, four kinds of protein backbone atoms of each amino acid residue, and matched PDB file were given. For example, the "bmr447.str" was extracted into four files: N_447.txt, Ca_447.txt, Ha_447.txt, and Hn_447.txt, which correspond to ^{15}N, $^{13}C_\alpha$, $^{1}H_\alpha$, and $^{1}H_N$ protein backbone atoms.

Secondly, the secondary structure information was extracted from PDB file which matched to BMRB star file. The secondary structure types of each amino acid residue are denoted by H, E, and C. Then the averaged chemical shifts for all the residues were calculated.

For protein backbone atoms "i" of amino acid type "j" with secondary structure type "k," the averaged chemical shift (ACS) is defined as

$$C_k^i(j) = \frac{1}{N}\sum_N \omega_k^i(j). \quad (1)$$

Here $i = {}^{15}$N, $^{13}C_\alpha$, $^{1}H_\alpha$, or $^{1}H_N$, j is one kind of 20 amino acids and k stands for the secondary structure types (H, E, or C) from DSSP [45] (H = helix, E = strand, and C = the rest). $\omega_k^i(j)$ is the chemical shift value extracted from the BMRB star file and N is the counts of $\omega_k^i(j)$ items.

By calculating the residues' ACS with (1) for 1552 proteins, we found that the ACS regularly varies with the secondary structure types and residues. The statistic results of averaged chemical shifts were listed in four tables, which can be accessed from our website. Take the $^{1}H_\alpha$ as an example, the ACS of $^{1}H_\alpha$ for each of 20 native amino acid residues with three types of secondary structure is shown in Figure 1. According to Figure 1, it can be concluded that we can use the ACS to represent the protein's secondary structure. In order to illustrate the algorithm, the flowchart of ACS is given in Figure 2.

2.3. Algorithm of Autocovariance of Average Chemical Shift (acACS). In order to obtain the correlation information between amino acids of a protein, the autocovariance of ACS was calculated. For a protein P,

$$P = [j_1, j_2 \cdots j_l \cdots j_L]. \quad (2)$$

Here, L is the sequence length and j_l is the amino acid in position l.

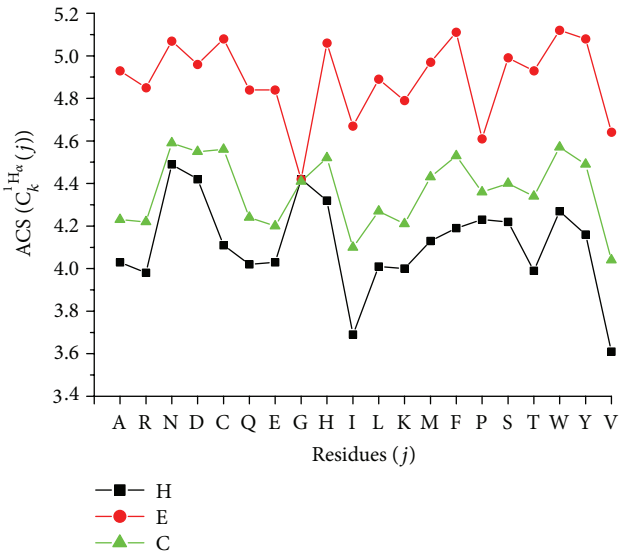


Figure 1: The average chemical shifts (ACS) of $^1H_\alpha$ for each of 20 native amino acid residues (j) with three types of secondary structure (k).

The secondary structure of protein P was predicted from Porter [46, 47] and then

$$P = [k_1, k_2 \cdots k_l \cdots k_L]. \quad (3)$$

Here k is the secondary structure types.

Then, the amino acid j_l in protein P was replaced by its ACS "$C^i_{k_l}(j_l)$" according to its secondary structure type k_l. When $C^i_{k_l}(j_l)$ was redefined as S^i_l, P can be expressed as

$$P = \left[S^i_1, S^i_2 \cdots S^i_l \cdots S^i_L\right] \quad \left(i = {}^{15}\mathrm{N}, {}^{13}\mathrm{C}_\alpha, {}^1\mathrm{H}_\alpha, {}^1\mathrm{H}_N\right). \quad (4)$$

Then, the autocovariance algorithm was used to calculate the correlation between amino acid l and $l+\lambda$ by the following equation:

$$\theta^i(\lambda) = \frac{1}{L-\lambda}\sum_{l=1}^{L-\lambda}\left[S^i_l - S^i_{(l+\lambda)}\right]^2, \quad \left(i = {}^{15}\mathrm{N}, {}^{13}\mathrm{C}_\alpha, {}^1\mathrm{H}_\alpha, {}^1\mathrm{H}_N,\ 0 < \lambda < L\right). \quad (5)$$

After the above calculation, the protein P can be expressed as follows:

$$P_{\mathrm{acACS}} = \left[\theta^i(0), \theta^i(1), \theta^i(2), \theta^i(3), \ldots, \theta^i(\lambda); \ldots\right] \quad \left(i = {}^{15}\mathrm{N}, {}^{13}\mathrm{C}_\alpha, {}^1\mathrm{H}_\alpha, {}^1\mathrm{H}_N,\ 0 < \lambda < L\right). \quad (6)$$

Here, $\theta^i(\lambda)$ is the correlation factor of average chemical shift S^i_l with average chemical shift $S^i_{l+\lambda}$. In particular, when $\lambda = 0$, with (5), $\theta^i(0) = 0$. In order to take use of ACS, the $\theta^i(0)$ was replaced by the average chemical shift S^i_l. The factor λ is a nonnegative integer and reflects the rank of correlation [40]. Based on different problems, in order to get a best result, a certain right number for factor λ should be given and so does i.

In order to give a pictorial representation of chemical shifting technique, a flow diagram is given in Figure 3, which shows how the acACS works.

Table 1: The comparison of the results with and without the acACS for predicting submitochondria locations and three membrane protein types with comparison to that without acACS.

	With acACS	Without acACS
Submitochondria locations	93.57%	91.46%
Three membrane protein types	97.79%	96.10%
Data set of Du [24]	94.95%	93.43%

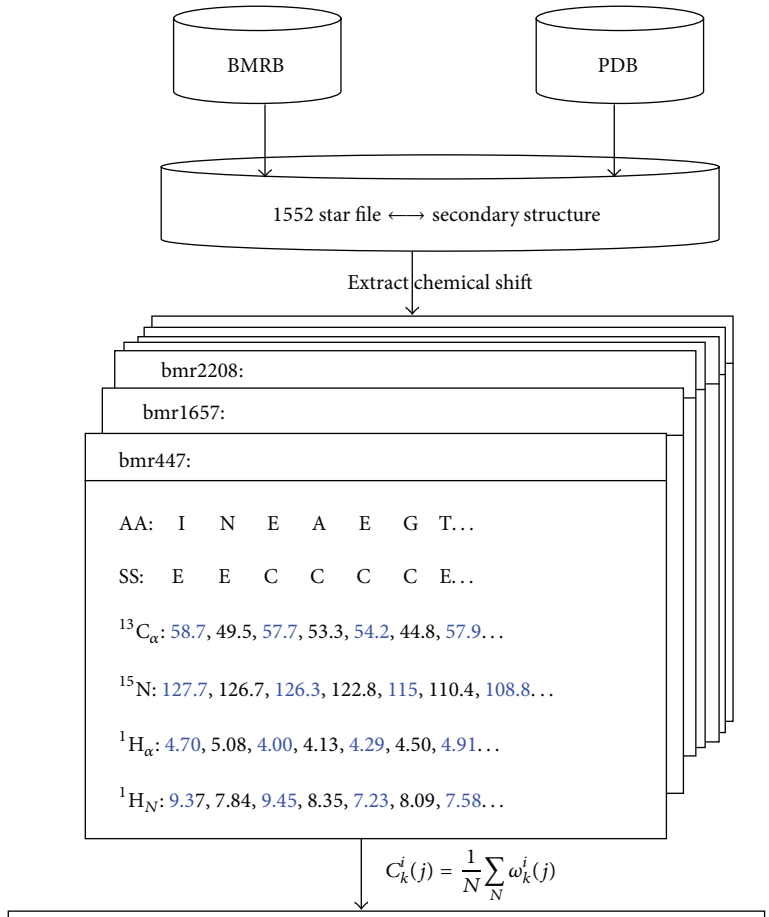
Table 2: The comparison of the results with and without the acACS for predicting mycobacterial subcellular localizations and three membrane protein types.

	With acACS	Without acACS
Mycobacterial subcellular localizations	87.77%	86.19%
Three membrane protein types	85.03%	83.71%
Data set of Rashid [53]	98.12%	96.85%

3. Results and Discussion

By using the acACS algorithm, we successfully represented the protein samples and accurately predicted submitochondria locations. We used the model to test the SML3-983 data set that was along with the SubMito-PSPCP [48]. The data set has 983 proteins sequences which were divided into three locations. Among the data set, there are 661 sequences from inner membrane, 177 sequences from matrix, and 145 sequences from outer membrane. We selected acACS combined with AAC, DC, PSSM, and GO and reduced physicochemical properties (Hn) as feature vectors for representing the proteins and then trained the model. Then 90.74% accuracy was obtained for SML3-983 data set with Jackknife cross-validation, which was 1.63% higher than SubMito-PSPCP. In order to compare the performance of acACS, the feature vector was recombined with AAC, DC, PSSM, GO, and Hn, without acACS. Then we trained the model and obtained the predicting accuracy of 89.52%, which was dropped about 1.2%.

The acACS algorithm has also been checked in our previous works [49–52]. In subcellular location prediction, we compared the results with and without the acACS in the submitochondria locations and mycobacterial proteins subcellular locations and got the better result which was listed in Tables 1 and 2. Actually, the acACS as a feature vector for representing the protein samples can also be used for other kinds of proteins prediction problem. In acidic and alkaline enzymes prediction and bioluminescent and nonbioluminescent proteins discrimination, we also improved the predicting accuracy by about 1.3%, which was listed in Table 3.



Average chemical shift for atoms $^{1}H_\alpha$

Secondary structure / Amino acids	H	E	C
A	4.03	4.93	4.23
R	3.98	4.85	4.22
N	4.49	5.07	4.59
D	4.42	4.96	4.55
C	4.11	5.08	4.56
Q	4.02	4.84	4.24
E	4.03	4.84	4.20
G	4.42	4.42	4.41
H	4.32	5.06	4.52
I	3.69	4.67	4.10
L	4.01	4.89	4.27
K	4.00	4.79	4.21
M	4.13	4.97	4.43
F	4.19	5.11	4.53
P	4.23	4.61	4.36
S	4.22	4.99	4.40
T	3.99	4.93	4.34
W	4.27	5.12	4.57
Y	4.16	5.08	4.49
V	3.61	4.64	4.04

Average chemical shift for atoms ^{15}N

Secondary structure / Amino acids	H	E	C
A	121.83	124.78	122.97
R	119.43	122.65	119.75
N	117.82	121.29	118.14
D	119.67	123.03	119.49
C	117.74	120.94	118.71
Q	118.70	122.13	118.87
E	119.55	122.57	119.98
G	107.82	109.81	110.24
H	118.51	121.68	118.97
I	119.80	123.06	119.65
L	119.97	124.75	120.61
K	119.45	123.14	120.16
M	118.67	122.23	119.46
F	119.71	121.54	118.69
P	119.85	119.40	119.76
S	115.32	117.45	116.02
T	115.08	117.59	113.11
W	120.77	122.80	119.33
Y	119.75	121.95	118.96
V	119.39	122.45	119.28

Average chemical shift for atoms $^{13}C_\alpha$

Secondary structure / Amino acids	H	E	C
A	54.70	50.92	52.94
R	58.92	54.88	56.50
N	55.35	52.46	53.65
D	56.74	53.45	54.64
C	61.74	56.14	57.57
Q	58.39	54.47	56.41
E	58.94	55.04	57.17
G	46.80	45.17	45.72
H	58.67	55.03	56.40
I	64.29	59.84	61.34
L	57.41	53.86	55.29
K	58.85	55.04	56.74
M	57.88	54.34	55.73
F	60.48	56.41	57.65
P	65.03	63.00	63.21
S	60.83	57.07	58.66
T	65.43	60.96	61.83
W	60.00	56.19	56.98
Y	60.63	56.58	57.90
V	65.83	60.66	62.43

Average chemical shift for atoms $^{1}H_N$

Secondary structure / Amino acids	H	E	C
A	8.11	8.63	8.16
R	8.10	8.65	8.18
N	8.54	8.60	8.38
D	8.22	8.53	8.34
C	8.15	8.67	8.12
Q	8.09	8.61	8.20
E	8.25	8.57	8.40
G	8.29	8.37	8.39
H	8.10	8.69	8.23
I	8.04	8.72	7.98
L	8.08	8.74	8.03
K	8.00	8.58	8.15
M	8.09	8.69	8.15
F	8.17	8.78	8.04
P	8.28	8.40	8.30
S	8.16	8.57	8.23
T	8.07	8.58	8.07
W	8.25	8.72	7.81
Y	8.12	8.76	8.07
V	7.99	8.68	8.00

Averaged chemical shifts

Figure 2: The flowchart of calculating the ACS. The AA denotes the amino acids and the SS denotes the secondary structure.

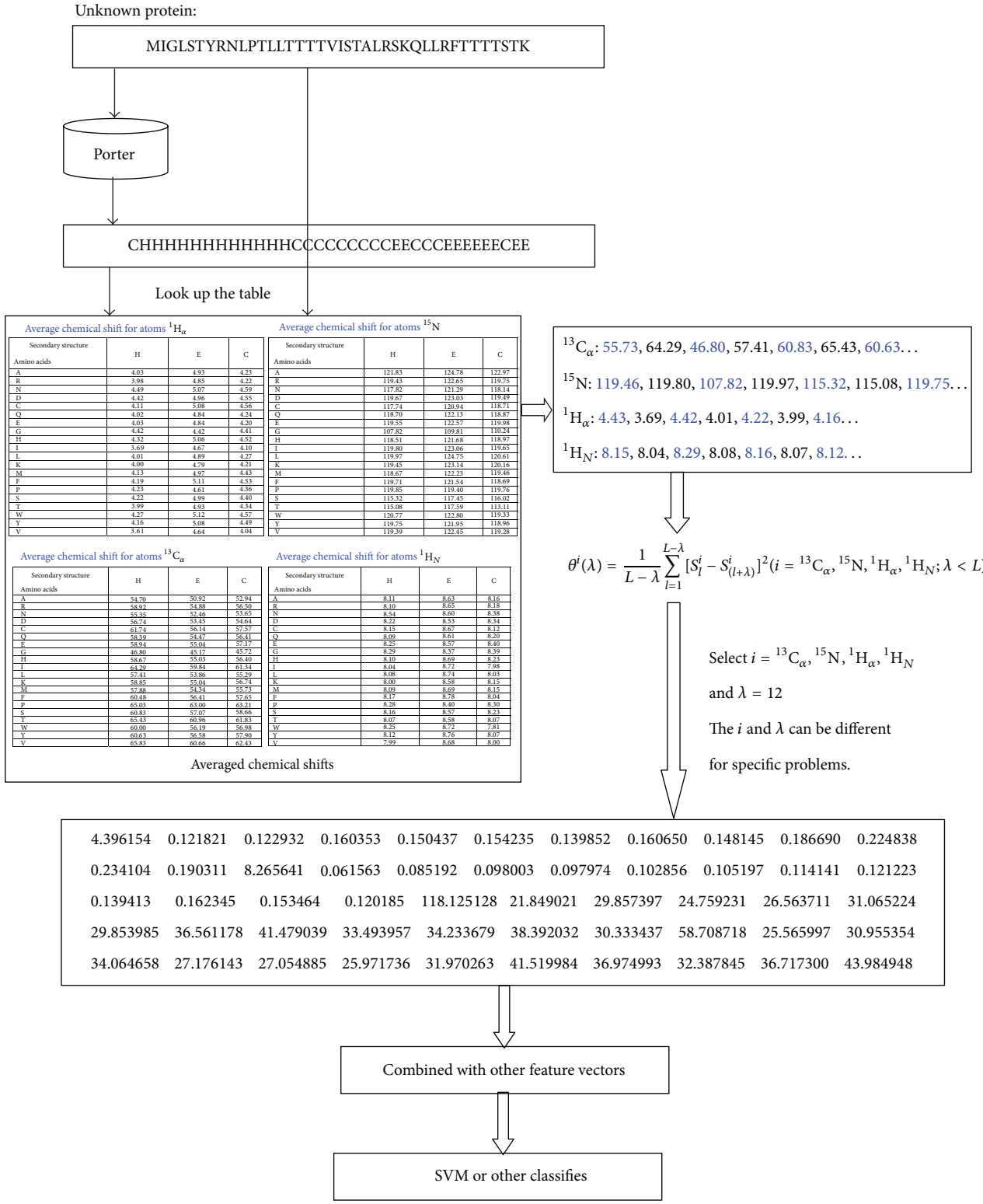


Figure 3: The flow diagram of the processing of the acACS.

Table 3: The comparison of the results with and without the acACS for other kinds of proteins prediction.

	With acACS	Without acACS
Acidic and alkaline enzymes	94.01%	92.52%
Bioluminescent and nonbioluminescent proteins	82.16%	80.90%

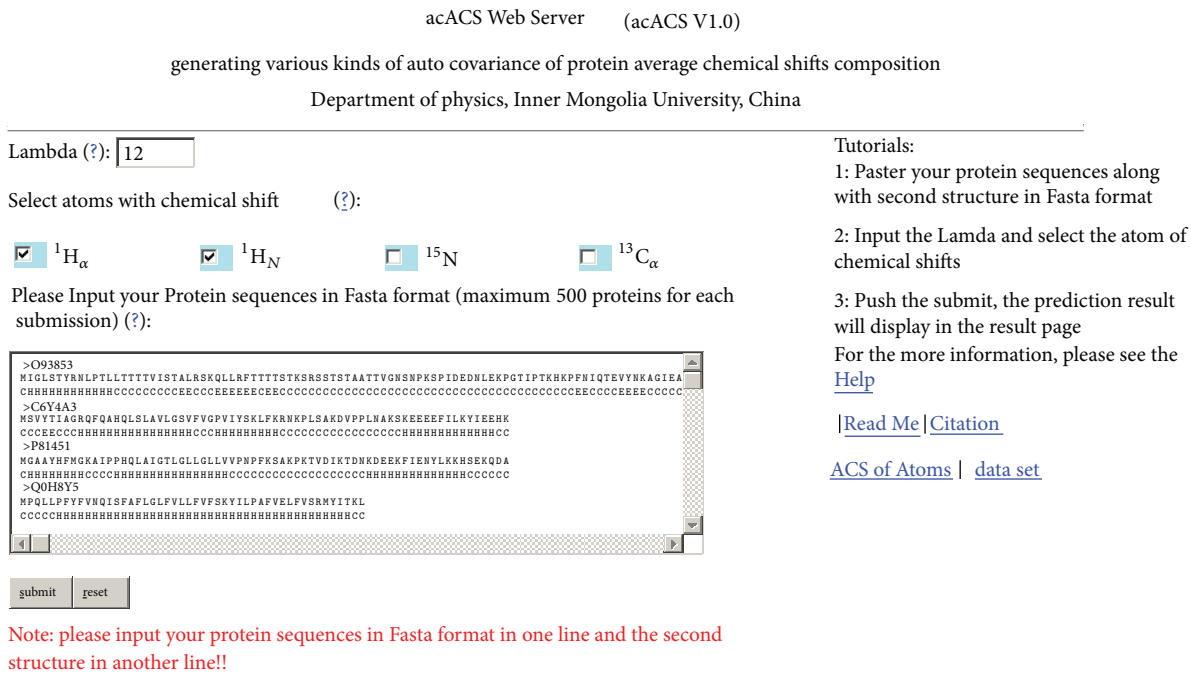


Figure 4: A screenshot for the top page of the web server acACS at http://wlxy.imu.edu.cn/college/biostation/fuwu/acACS/index.asp.

The protein functions, including its subcellular locations, are largely determined by its structure. Developing a novel method for improving the performance of predicting protein subcellular locations is urgent. However, the feature vectors in the methods were almost sequence-based in the past. Therefore, almost state-of-the-art methods tried to incorporate some other sequence-based information as its complement. Our method provides structure-based information and can be perfect complement to the sequence-based methods and can be used for other kinds of protein related problems. Actually, these methods can work side by side to help each other in a practical study.

For the chemical shift, it incorporates the structure information in the first place, so it can represent the protein sample better. What is the better ways to use of the chemical shift is still a hot topic for biologist and chemist. In this work, we used the autocovariance algorithm to process the averaged chemical shift and got the better results, but there are certainly some improvements that could be made for acACS. Actually other algorithms can be adopted to try to find the better method for representing the protein samples in the future. At present stage, it is not convenient for the user, for both the secondary structure information and the protein sequence that are used, to calculate the chemical shift. In the future, the secondary structure information will not be necessary, and it will be integrated into the algorithm.

4. Conclusions

In this work, the raw chemical shifts data set and averaged chemical shift data set were constructed. Then, the averaged chemical shift was calculated and the algorithm of acACS was presented. In order to check the performance of the acACS we got, proteins submitochondria locations, mycobacterial proteins subcellular locations, bioluminescent proteins discrimination, and acidic and alkaline enzymes classification were predicted. Based on the results we obtain, it can be concluded that the acACS can improve the accuracy of prediction at least 1%-2%, the performance of which is correlated with the correlation factor λ and the backbone atoms i. Some recent studies showed that the profile-based features [54–56], pseudoamino acid composition (PseAAC) [57], and features based on physicochemical proprieties of amino acids [58] were able to improve the performance of many computational predictors for protein remote homology detection, protein binding site identification, and so forth. Therefore, these features should be studied for protein subcellular location prediction in the future studies.

We have developed a web server acACS, which could automatically produce the vectors of proteins, when a custom submitted the protein sequences along with the secondary structure in batch mode. The data set can be a very useful addition to biomolecular NMR spectroscopists. The acACS will be of benefit to the proteomic research. The current work will become an important progress in the prediction of the protein subcellular locations and promote the study in the related areas.

5. Web Server and User Guide

To enhance the value of its practical applications, a web server for the acACS generator was established. Moreover, for the convenience of the user, here a step-to-step guide is provided for how to use the web server to get the desired results.

Step 1. Open the web server at http://wlxy.imu.edu.cn/college/biostation/fuwu/acACS/index.asp and you will see the top page of the acACS on your computer screen, as shown

in Figure 4. Click on the Read Me button to see a brief introduction about the acACS.

Step 2. Either type or copy/paste the query protein sequences into the input box at the center of Figure 4, and then copy/paste the secondary structure of the protein sequence in the next line. The input sequence should be in "ONE LINE" format. For the examples of sequences in ONE LINE format, click the "?" button above the input box.

Step 3. Input the Lambda value in the input box right of the Lambda label.

Step 4. Check atoms with chemical shift.

Step 5. Click on the Submit button to see the result page. For example, if you use the default example sequences, Lambda and atoms in the window, after clicking the Submit button, you will see the following message shown on the screen of your computer: "The lamda you have chosen is 12"; "The Atom of chemical shift you have chosen are $^1H_\alpha$, ^1H_N"; "The acACSs of the proteins you submitted are.....". Then the acACS of $^1H_\alpha$ atom was given and the acACS of ^1H_N atom followed for the first protein, then the acACS of second protein, the third, and so forth.

Step 6. Click the ACS of atoms and data set button to download the benchmark dataset used to calculate the ACS.

Step 7. Click the Citation button to find the relevant papers that document the detailed development and algorithm of acACS.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors would like to thank the reviewers for their helpful comments on their paper. This work was supported by a Grants from National Natural Science Foundation of China (61063016 and 31160188), The Scientific Research Program at Universities of Inner Mongolia Autonomous Region of China (NJZY13014), The Natural Science Foundation of Inner Mongolia Autonomous Region of China (2013MS0504 and 2013MS0503), and the Program of Higher-level Talents of Inner Mongolia University (135147).

References

[1] R. Casadio, P. L. Martelli, and A. Pierleoni, "The prediction of protein subcellular localization from sequence: a shortcut to functional genome annotation," *Briefings in Functional Genomics and Proteomics*, vol. 7, no. 1, pp. 63–73, 2008.

[2] J.-Y. Shi, S.-W. Zhang, Q. Pan, Y.-M. Cheng, and J. Xie, "Prediction of protein subcellular localization by support vector machines using multi-scale energy and pseudo amino acid composition," *Amino Acids*, vol. 33, no. 1, pp. 69–74, 2007.

[3] M. R. Bakhtiarizadeh, M. Moradi-Shahrbabak, M. Ebrahimi, and E. Ebrahimie, "Neural network and SVM classifiers accurately predict lipid binding proteins, irrespective of sequence homology," *Journal of Theoretical Biology*, vol. 356, pp. 213–222, 2014.

[4] T. D. Campos, N. D. Young, P. K. Korhonen et al., "Identification of G protein-coupled receptors in *Schistosoma haematobium* and *S. mansoni* by comparative genomics," *Parasites & Vectors*, vol. 7, no. 1, article 242, 2014.

[5] Y. L. Chen and Q. Z. Li, "Prediction of apoptosis protein subcellular location using improved hybrid approach and pseudo-amino acid composition," *Journal of Theoretical Biology*, vol. 248, no. 2, pp. 377–381, 2007.

[6] H. Ding, S. Guo, E. Deng et al., "Prediction of Golgi-resident protein types by using feature selection technique," *Chemometrics and Intelligent Laboratory Systems*, vol. 124, pp. 9–13, 2013.

[7] S. Mondal and P. P. Pai, "Chou's pseudo amino acid composition improves sequence-based antifreeze protein prediction," *Journal of Theoretical Biology*, vol. 356, pp. 30–35, 2014.

[8] L. Zhang, B. Liao, D. Li, and W. Zhu, "A novel representation for apoptosis protein subcellular localization prediction using support vector machine," *Journal of Theoretical Biology*, vol. 259, no. 2, pp. 361–365, 2009.

[9] Y. C. Zuo, Y. Peng, L. Liu, W. Chen, L. Yang, and G. L. Fan, "Predicting peroxidase subcellular location by hybridizing different descriptors of Chou' pseudo amino acid patterns," *Analytical Biochemistry*, vol. 458, pp. 14–19, 2014.

[10] H. Ding, P.-M. Feng, W. Chen, and H. Lin, "Identification of bacteriophage virion proteins by the ANOVA feature selection and analysis," *Molecular Biosystems*, 2014.

[11] H. Ding, E.-Z. Deng, L.-F. Yuan et al., "iCTX-Type: a sequence-based predictor for identifying the types of conotoxins in targeting ion channels," *BioMed Research International*, vol. 2014, Article ID 286419, 10 pages, 2014.

[12] Y. D. Cai and K. C. Chou, "Using neural networks for prediction of subcellular location of prokaryotic and eukaryotic proteins," *Molecular Cell Biology Research Communications*, vol. 4, no. 3, pp. 172–173, 2000.

[13] Q. Z. Li and Z. Q. Lu, "The prediction of the structural class of protein: application of the measure of diversity," *Journal of Theoretical Biology*, vol. 213, no. 3, pp. 493–502, 2001.

[14] Y. Jin, B. Niu, K. Feng, W. Lu, Y. Cai, and G. Li, "Predicting subcellular localization with AdaBoost learner," *Protein and Peptide Letters*, vol. 15, no. 3, pp. 286–289, 2008.

[15] Y. D. Cai and K. C. Chou, "Nearest neighbour algorithm for predicting protein subcellular location by combining functional domain composition and pseudo-amino acid composition," *Biochemical and Biophysical Research Communications*, vol. 305, no. 2, pp. 407–411, 2003.

[16] K. C. Chou and H. B. Shen, "Predicting eukaryotic protein subcellular location by fusing optimized evidence-theoretic K-nearest neighbor classifiers," *Journal of Proteome Research*, vol. 5, no. 8, pp. 1888–1897, 2006.

[17] L. Nanni and A. Lumini, "Genetic programming for creating Chou's pseudo amino acid based features for submitochondria localization," *Amino Acids*, vol. 34, no. 4, pp. 653–660, 2008.

[18] K.-C. Chou and H.-B. Shen, "Predicting protein subcellular location by fusing multiple classifiers," *Journal of Cellular Biochemistry*, vol. 99, no. 2, pp. 517–527, 2006.

[19] O. Emanuelsson, S. Brunak, G. von Heijne, and H. Nielsen, "Locating proteins in the cell using TargetP, SignalP and related tools," *Nature Protocols*, vol. 2, no. 4, pp. 953–971, 2007.

[20] A. Höglund, P. Dönnes, T. Blum, H. Adolph, and O. Kohlbacher, "MultiLoc: prediction of protein subcellular localization using N-terminal targeting sequences, sequence motifs and amino acid composition," *Bioinformatics*, vol. 22, no. 10, pp. 1158–1165, 2006.

[21] P. Horton, K. Park, T. Obayashi et al., "WoLF PSORT: protein localization predictor," *Nucleic Acids Research*, vol. 35, no. 2, pp. W585–W587, 2007.

[22] S. Brady and H. Shatkay, "EPILOC: a (working) text-based system for predicting protein subcellular location," in *Proceedings of the 13th Pacific Symposium on Biocomputing (PSB '08)*, pp. 604–615, January 2008.

[23] A. Fyshe, Y. Liu, D. Szafron, R. Greiner, and P. Lu, "Improving subcellular localization prediction using text classification and the gene ontology," *Bioinformatics*, vol. 24, no. 21, pp. 2512–2517, 2008.

[24] P. Du and Y. Li, "Prediction of protein submitochondria locations by hybridizing pseudo-amino acid composition with various physicochemical features of segmented sequence," *BMC Bioinformatics*, vol. 7, article 518, 2006.

[25] K. C. Chou and Y. D. Cai, "Prediction of protein subcellular locations by GO-FunD-PseAA predictor," *Biochemical and Biophysical Research Communications*, vol. 320, no. 4, pp. 1236–1239, 2004.

[26] A. B. Sibley, M. Cosman, and V. V. Krishnan, "An empirical correlation between secondary structure content and averaged chemical shifts in proteins," *Biophysical Journal*, vol. 84, no. 2 I, pp. 1223–1227, 2003.

[27] S. P. Mielke and V. V. Krishnan, "Protein structural class identification directly from NMR spectra using averaged chemical shifts," *Bioinformatics*, vol. 19, no. 16, pp. 2054–2064, 2003.

[28] S. Spera and A. Bax, "Empirical correlation between protein backbone conformation and C.alpha. and C.beta. 13C nuclear magnetic resonance chemical shifts," *Journal of the American Chemical Society*, vol. 113, no. 14, pp. 5490–5492, 1991.

[29] Y. Zhao, B. Alipanahi, S. C. Li, and M. Li, "Protein secondary structure prediction using NMR chemical shift data," *Journal of Bioinformatics and Computational Biology*, vol. 8, no. 5, pp. 867–884, 2010.

[30] P. Luginbühl, T. Szyperski, and K. Wüthrich, "Statistical basis for the use of $^{13}C^{\alpha}$ chemical shifts in protein structure determination," *Journal of Magnetic Resonance B*, vol. 109, no. 2, pp. 229–233, 1995.

[31] D. S. Wishart, B. D. Sykes, and F. M. Richards, "Relationship between nuclear magnetic resonance chemical shift and protein secondary structure," *Journal of Molecular Biology*, vol. 222, no. 2, pp. 311–333, 1991.

[32] D. S. Wishart, D. Arndt, M. Berjanskii, P. Tang, J. Zhou, and G. Lin, "CS23D: a web server for rapid protein structure generation using NMR chemical shifts and sequence data.," *Nucleic acids research*, vol. 36, pp. W496–502, 2008.

[33] B. R. Seavey, E. A. Farr, W. M. Westler, and J. L. Markley, "A relational database for sequence-specific protein NMR data," *Journal of Biomolecular NMR*, vol. 1, no. 3, pp. 217–236, 1991.

[34] Y. Shen and A. Bax, "Protein backbone chemical shifts predicted from searching a database for torsion angle and sequence homology," *Journal of Biomolecular NMR*, vol. 38, no. 4, pp. 289–302, 2007.

[35] B. Han, Y. Liu, S. W. Ginzinger, and D. S. Wishart, "SHIFTX2: significantly improved protein chemical shift prediction," *Journal of Biomolecular NMR*, vol. 50, no. 1, pp. 43–57, 2011.

[36] Y. Tian, S. J. Opella, and F. M. Marassi, "Improved chemical shift prediction by Rosetta conformational sampling," *Journal of Biomolecular NMR*, vol. 54, no. 3, pp. 237–243, 2012.

[37] J. A. Vila, M. E. Villegas, H. A. Baldoni, and H. A. Scheraga, "Predicting 13Cα chemical shifts for validation of protein structures," *Journal of Biomolecular NMR*, vol. 38, no. 3, pp. 221–235, 2007.

[38] C. J. Markin and L. Spyracopoulos, "Accuracy and precision of protein-ligand interaction kinetics determined from chemical shift titrations," *Journal of Biomolecular NMR*, vol. 54, no. 4, pp. 355–376, 2012.

[39] C. J. Markin and L. Spyracopoulos, "Increased precision for analysis of protein-ligand dissociation constants determined from chemical shift titrations," *Journal of Biomolecular NMR*, vol. 53, no. 2, pp. 125–138, 2012.

[40] K. Chou, "Some remarks on protein attribute prediction and pseudo amino acid composition," *Journal of Theoretical Biology*, vol. 273, no. 1, pp. 236–247, 2011.

[41] P. Du, X. Wang, C. Xu, and Y. Gao, "PseAAC-Builder: a cross-platform stand-alone program for generating various special Chou's pseudo-amino acid compositions," *Analytical Biochemistry*, vol. 425, no. 2, pp. 117–119, 2012.

[42] P. Du, S. Gu, and Y. Jiao, "PseAAC-general: fast building various modes of general form of Chou's pseudo-amino acid composition for large-scale protein datasets," *International Journal of Molecular Sciences*, vol. 15, no. 3, pp. 3495–3506, 2014.

[43] H. M. Berman, J. Westbrook, Z. Feng et al., "The protein data bank," *Nucleic Acids Research*, vol. 28, no. 1, pp. 235–242, 2000.

[44] W. Li, L. Jaroszewski, and A. Godzik, "Clustering of highly homologous sequences to reduce the size of large protein databases," *Bioinformatics*, vol. 17, no. 3, pp. 282–283, 2001.

[45] W. Kabsch and C. Sander, "Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features.," *Biopolymers*, vol. 22, no. 12, pp. 2577–2637, 1983.

[46] G. Pollastri, A. J. M. Martin, C. Mooney, and A. Vullo, "Accurate prediction of protein secondary structure and solvent accessibility by consensus combiners of sequence and structure information," *BMC Bioinformatics*, vol. 8, article 201, 2007.

[47] G. Pollastri and A. McLysaght, "Porter: a new, accurate server for protein secondary structure prediction," *Bioinformatics*, vol. 21, no. 8, pp. 1719–1720, 2005.

[48] P. Du and Y. Yu, "SubMito-PSPCP: predicting protein submitochondrial locations by hybridizing positional specific physicochemical properties with pseudoamino acid compositions," *BioMed Research International*, vol. 2013, Article ID 263829, 7 pages, 2013.

[49] G. L. Fan and Q. Z. Li, "Predicting protein submitochondria locations by combining different descriptors into the general form of Chou's pseudo amino acid composition," *Amino Acids*, vol. 43, no. 2, pp. 545–555, 2012.

[50] G. L. Fan and Q. Z. Li, "Predict mycobacterial proteins subcellular locations by incorporating pseudo-average chemical shift into the general form of Chou's pseudo amino acid composition," *Journal of Theoretical Biology*, vol. 304, pp. 88–95, 2012.

[51] G. L. Fan and Q. Z. Li, "Discriminating bioluminescent proteins by incorporating average chemical shift and evolutionary information into the general form of Chou's pseudo amino acid composition," *Journal of Theoretical Biology*, vol. 334, pp. 45–51, 2013.

[52] G. L. Fan, Q. Z. Li, and Y. C. Zuo, "Predicting acidic and alkaline enzymes by incorporating the average chemical shift and gene ontology informations into the general form of Chou's PseAAC," *Process Biochemistry*, vol. 48, no. 7, pp. 1048–1053, 2013.

[53] M. Rashid, S. Saha, and G. P. S. Raghava, "Support Vector Machine-based method for predicting subcellular localization of mycobacterial proteins using evolutionary information and motifs," *BMC Bioinformatics*, vol. 8, article 337, 2007.

[54] B. Liu, X. Wang, L. Lin, and Q. Dong, "A discriminative method for protein remote homology detection and fold recognition combining Top-*n*-grams and latent semantic analysis," *BMC Bioinformatics*, vol. 9, article 510, 2008.

[55] B. Liu, J. Xu, and Q. Zou, "Using distances between Top-n-gram and residue pairs for protein remote homology detection," *Bmc Bioinformatics*, vol. 15, article S3, 2014.

[56] B. Liu, D. Zhang, R. Xu et al., " Combining evolutionary information extracted from frequency profiles with sequence-based kernels for protein remote homology detection," *Bioinformatics*, vol. 30, no. 4, pp. 472–479, 2014.

[57] B. Liu, X. Wang, Q. Zou, Q. Dong, and Q. Chen, "Protein remote homology detection by combining Chou's pseudo amino acid composition and profile-based protein representation," *Molecular Informatics*, vol. 32, no. 9-10, pp. 775–782, 2013.

[58] B. Liu, X. Wang, Q. Chen, Q. Dong, and X. Lan, "Using amino acid physicochemical distance transformation for fast protein remote homology detection," *PLoS ONE*, vol. 7, no. 9, Article ID e46633, 2012.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 978503, 5 pages
http://dx.doi.org/10.1155/2014/978503

Research Article

Prediction of Four Kinds of Simple Supersecondary Structures in Protein by Using Chemical Shifts

Feng Yonge

College of Science, Inner Mongolia Agriculture University, Hohhot 010018, China

Correspondence should be addressed to Feng Yonge; fengyonge@163.com

Received 7 May 2014; Revised 3 June 2014; Accepted 4 June 2014; Published 18 June 2014

Academic Editor: Hao Lin

Knowledge of supersecondary structures can provide important information about its spatial structure of protein. Some approaches have been developed for the prediction of protein supersecondary structure. However, the feature used by these approaches is primarily based on amino acid sequences. In this study, a novel model is presented to predict protein supersecondary structure by use of chemical shifts (CSs) information derived from nuclear magnetic resonance (NMR) spectroscopy. Using these CSs as inputs of the method of quadratic discriminant analysis (QD), we achieve the overall prediction accuracy of 77.3%, which is competitive with the same method for predicting supersecondary structures from amino acid compositions in threefold cross-validation. Moreover, our finding suggests that the combined use of different chemical shifts will influence the accuracy of prediction.

1. Introduction

The prediction of protein structure is always one of the most important research topics in the field of bioinformatics. However, it is very difficult to predict the spatial structure directly from the protein sequence. Therefore, the prediction of supersecondary structure is an important step in the prediction of protein spatial structure. The supersecondary structural motifs are composed of a few secondary structural elements (namely, α or β) connected by loops. At present, there are four kinds of simple supersecondary structures, namely, α-loop-β, α-loop-α, β-loop-α, and β-loop-β. These motifs play an important role in protein folding and stability because a large number of motifs exist in protein spatial structure. Many researches have focused on exploring methods for protein supersecondary structure prediction [1, 2]. In 1995, Sun et al. predicted protein supersecondary structure and achieved an accuracy of between 70 and 80% by using neural networks [3]. Chou and Blinn presented a method for predicting beta turns [4–6], alpha turns [7], and all the tight turns [6]. Cruz et al. identified β-hairpin and non-β-hairpin [8]. Hu and Li identified four kinds of simple supersecondary structures in 2088 proteins and achieved an accuracy of 78~83 % [9]. Zou et al. also predicted four kinds of simple supersecondary structures from 3088 proteins by using support vector machine [10]. And the overall accuracy of 78% was achieved. The features of these studies were mainly derived from the amino acid compositions or dipeptide compositions.

Nuclear magnetic resonance (NMR) technique plays an important role in the determination of three-dimensional biological macromolecule structures. NMR chemical shifts encode subtle information about the local chemical environment of nuclear spins. For many years, there has been growing interest to access this information and utilize it for biomolecular structure determination [11, 12]. Recent progress was made by combining chemical shifts with protein structure prediction programs [13–20], showing that chemical shifts information is a power parameter for the determination of protein structure. In this paper, we utilized chemical shifts as parameters to predict four kinds of simple supersecondary structures in protein by the method of quadratic discriminant analysis. Using the benchmark dataset, we achieved the average of sensitivity of 76.3% and specificity of 74.3% and the overall prediction accuracy of 77.3% in threefold cross-validation by using six CSs ($C, C_\alpha, C_\beta, H, H_\alpha, N$) as features. Moreover, we have performed the prediction by combining the different chemical shifts as features. Results showed that the redundant information has great influence on the accuracy.

2. Materials and Methods

2.1. Database. The chemical shifts of all nuclei ($C, C_\alpha, C_\beta, H, H_\alpha, N$) in proteins were extracted from re-referenced protein chemical shift database (namely, RefDB [21]). The following steps were performed to construct the dataset. Firstly, only proteins with six nuclei assigned CSs were considered. Secondly, only proteins with the supersecondary structures information in ArchDB40 [22] were available. We finally utilized the PISCES program [23] to remove the highly similar sequences. After strictly following the aforementioned procedures, 114 proteins were obtained which have both CSs and supersecondary structures. Among 114 proteins, 92% (105 sequences) proteins have less than 25% sequence identity, and the sequence identity of the remains ranges from 25 to 30%. The appendix lists 114 proteins used in this study. Finally, we obtained 90 α-loop-α (HH), 89 α-loop-β (HE), 97 β-loop-α (EH), and 122 β-loop-β (EE) motifs, including the β-β link and β-β hairpin.

2.2. Feature Parameter. In the four data subsets $\{HH, HE, EH, EE\}$, we calculated the averaged CSs of six nuclei for a sequence of length l using the following formula:

$$t_i = \frac{1}{l}\sum_{i=1}^{l} \mathrm{CS}_i, \tag{1}$$

where $i = C, C_\alpha, C_\beta, H, H_\alpha, N$. Therefore, a sequence can be converted into a six-dimensional vector $R : \{t_i\}$.

2.3. Prediction Algorithm. To design an efficient and accurate predicted algorithm the key step is in protein supersecondary structure prediction. The quadratic discriminant analysis [24] is a power algorithm that has been widely applied in genomic and proteomic bioinformatics. Thus, we used it here to perform prediction.

2.4. Quadratic Discriminant Analysis (QD). For a sequence X to be classified, we calculated the averaged CSs of six nuclei using (1). So, the sequence is converted into a six-dimensional vector $R : \{t_i\}$:

$$R = \{t_i\} \quad \left(i = C, C_\alpha, C_\beta, H, H_\alpha, N\right). \tag{2}$$

Here we integrated six-dimensional vector by using quadratic discriminant analysis function. Consider a sequence X is classified into four groups (HH, HE, EH, EE). The discriminant analysis function between group i and group j is defined by

$$\xi_{ij} = \ln p\left(\omega_i \mid X\right) - \ln p\left(\omega_j \mid X\right). \tag{3}$$

According to Bayes' Theorem, we deduce

$$\begin{aligned}\xi_{ij} &= \ln\frac{p_i}{p_j} - \frac{\delta_i - \delta_j}{2} - \frac{1}{2}\ln\frac{\left|\Sigma_i\right|}{\left|\Sigma_j\right|} \\ &= \left(\ln p_i - \frac{1}{2}\delta_i - \frac{1}{2}\ln\left|\Sigma_i\right|\right) \\ &\quad - \left(\ln p_j - \frac{1}{2}\delta_j - \frac{1}{2}\ln\left|\Sigma_j\right|\right).\end{aligned} \tag{4}$$

The result can be generalized to *four* groups directly and described as follows.

Set

$$\begin{aligned}\eta_v &= \ln p_v - \frac{\delta_v}{2} - \frac{1}{2}\ln\left|\Sigma_v\right| \\ &\left(v = HH, EH, HE, EE\right),\end{aligned} \tag{5}$$

where

$$\delta_v = \left(R - \mu_v\right)^T \Sigma_v^{-1}\left(R - \mu_v\right), \tag{6}$$

where p_v denotes the number of samples in group v, δ_v is the square mahalanobis distance between R and μ_v with respect to Σ_v (note: μ_v and $|\Sigma_v|$ are calculated in training set), and μ_v denotes chemical shift values of six nuclei $R : \{t_i\}$ averaged over group v; $|\Sigma_v|$ is the determinant of matrix Σ_v.

The six-dimensional vector μ_v can be written as

$$\mu_v^{(i)} = \frac{1}{p_v}\sum_{i=1}^{p_v} t_i, \tag{7}$$

where $v = HH, EH, HE, EE$; $i = C, C_\alpha, C_\beta, H_\alpha, H, N$; Σ_v is the covariance matrix of 6×6 dimension, quantifying correlations between the chemical shifts of six nuclei:

$$\Sigma_v = \begin{bmatrix} \sigma^v{}_{1,1} & \sigma^v{}_{1,2} & \cdots & \sigma^v{}_{1,6} \\ \sigma^v{}_{2,1} & \sigma^v{}_{2,2} & \cdots & \sigma^v{}_{2,6} \\ \vdots & \vdots & \vdots & \vdots \\ \sigma^v{}_{6,1} & \sigma^v{}_{6,2} & \cdots & \sigma^v{}_{6,6} \end{bmatrix}, \tag{8}$$

where the element

$$\sigma_{i,j}^v = \frac{1}{p_v}\sum\left(t_i - \mu_v^{(i)}\right)\left(t_j - \mu_v^{(j)}\right). \tag{9}$$

Here $v = HH, EH, HE, EE$; $i, j = C, C_\alpha, C_\beta, H_\alpha, H, N$.

From (4) and (5), we have concluded

$$\xi_{ij} = \eta_i - \eta_j. \tag{10}$$

It can be easily proved that $p(\omega_k \mid X)$ is the maximum of $p(\omega_v \mid X)$, if η_k is the maximal one in η_v ($v = HH, EH, HE, EE$). Then, we predict that X belongs to group k.

2.5. Correction in the Error Allowed Scope. A sequence X is predicted for four kinds of supersecondary structures by using (1)~(10). If η_i is the maximal one in η_k ($k = HH, EH, HE, EE$), then we predict that X belongs to group i. However, there are slight differences among

η_k ($k = HH, EH, HE, EE$). To correct predicted results, we define the coefficient of the error allowed scope as

$$R = \frac{\eta_{\text{corr}} - \eta_{\text{wro}}}{\eta_{\text{corr}}}, \quad (11)$$

where η_{corr} denotes X belonging to itself class η, η_{wro} denotes X being predicted another class η. For example, if X is the super-secondary structure of HH, then η_{corr} is η_{HH} and η_{wro} is the maximum among $\eta_{EH}, \eta_{HE}, \eta_{EE}$.

2.6. Performance Evaluation. In statistical prediction, independent dataset test, cross-validation test, and jackknife test can be used to examine a predictor for its effectiveness in practical application. Among the three test methods, the jackknife test is deemed to be the least arbitrary that can always yield a unique result for a given benchmark dataset [25] and has been widely used to examine the performance of various predictors [26–37]. However, in this study we have used the threefold cross-validation to examine the performance of our method; in order to reduce the computational time, we randomly divided the training set into three parts, two of which are for training and the rest for testing. The process is repeated three times. The following three parameters: sensitivity (SN_i), specificity (SP_i), and overall accuracy (Q_{total}), are used to evaluate the predictive performance of our approach:

$$\text{SN}_i = \frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i} \times 100\%, \quad (12)$$

$$\text{SP}_i = \frac{\text{TP}_i}{\text{TP}_i + \text{FP}_i} \times 100\%, \quad (13)$$

$$Q_{\text{total}} = \frac{\sum_i \text{TP}_i}{N} \times 100\%, \quad (14)$$

where $i = HH, HE, EH, EE$ and TP, FN, TN, and FP denote, respectively, true positives, false positives, true negatives, and false positives. N is total number of sequences in four data subsets.

3. Results and Discussion

Under the benchmark dataset, we calculated the average chemical shift values using (1). The sequences from four data subsets are converted, respectively, into six-dimensional vectors, which are derived from chemical shift values of six nuclei; then μ is also a six-dimensional mean vector, which is calculated in each of the datasets. In the training sets, determinant and inverse matrix of covariance matrix Σ_v are calculated. Given a sequence of the testing sets, we may calculate η_v by using (4)~(10) and compare the results. Then the class of sequence X was determined by the maximum of η_v ($v = HH, HE, EH, EE$). Moreover, the coefficient R given in (11) is used to correct predicted results. The current study utilized $R < 0.4$. The results of threefold cross-validation are listed in Table 1.

From Table 1, we can see that the averaged sensitivity, specificity, and overall accuracy of four kinds of supersecondary structures are 76.3%, 74.3%, and 77.3%, respectively, indicating that CSs are highly informative with regard to supersecondary structures.

Table 1: The predicted accuracies by using six CSs as features (3-fold cross-validation).

Class structure	SN (%)	SP (%)	Average SN (%)	Average SP (%)	Q_{total} (%)
	$R < 0.4$				
HH	73.0	71.0	76.3	74.3	77.3
EH	75.8	78.1			
HE	69.0	66.7			
EE	87.5	81.4			

Generally speaking, chemical shift measurements can be incomplete for a multitude of reasons. Often, chemical shifts can only be assigned partially or are missing. To assess the impact of incomplete chemical shift assignments, we performed the prediction by using the combination of the different chemical shifts as features. The results are shown in Table 2.

From Table 2, we found that omission of some CSs can result in radically different accuracy. Theoretically, incomplete chemical shifts provide relatively less information, so the predicted accuracy is also declined. But it actually did not in prediction. We used CSs of H, H_α, C as features and achieved the highest accuracy of prediction, indicating that the results are affected by the redundant data. According to the performances, we concluded that CSs of N, C_α, C_β are the most informative features in the prediction of four kinds of protein supersecondary structures. In addition, the information of C, H_α, N is commonly provided in protein database; we achieved the prediction accuracy of 79.1% by using CSs of C, H_α, N as the only inputs.

To test the method and facilitate comparison with other features, we used amino acid compositions (AAC) as inputs of the method of quadratic discriminant analysis. The compared results are recorded in Table 2. Compared results show that the performances of CSs are superior to that of AAC for supersecondary structures prediction, except *HE* structure (compared with six CSs).

4. Conclusions

In this paper, we have introduced a prediction model for supersecondary structures from protein chemical shifts. Our model is both simple and easy to perform. However, owing to the limitation of both information of supersecondary structures and corresponding chemical shifts of six nuclei that should be considered, only 114 proteins have been selected in this study. Based on the benchmark dataset, we investigated the relationship between supersecondary structures and chemical shifts. We achieved the overall accuracy of 77.3% by using six CSs as features and the maximum overall accuracy of 89.2% by using the combination of CSs of N, C_α, C_β. Results show that chemical shift is a good parameter for the prediction of four kinds of protein supersecondary structures. In summary, the chemical shifts

Table 2: Predicted results of different feature combinations ($R < 0.4$).

Feature combinations	*HH*		*EH*		*HE*		*EE*		Average SN (%)	Average SP (%)	Q_{total} (%)
	SN (%)	SP (%)	SN (%)	SP (%)	SN (%)	SP (%)	SN (%)	SP (%)			
$C, C_\alpha, C_\beta, H, H_\alpha$	63.3	77.0	84.5	45.6	34.8	100	71.3	77.0	63.4	74.9	64.6
$C, C_\alpha, C_\beta, H_\alpha, N$	90.0	85.3	66.0	97.0	85.4	86.4	93.4	75.5	83.7	86.1	84.2
C, C_α, C_β, N	55.6	87.7	61.9	80	44.9	93.0	95.1	52.5	64.4	78.3	66.8
C_α, C_β, N	90.0	87.1	94.8	83.6	79.8	93.4	91.0	91.7	88.9	89.0	89.2
C, H_α, N	90.0	73.6	75.3	82.0	79.8	81.6	73.8	80.4	79.7	79.4	79.1
AAC	73.3	73.6	73.0	77.8	72.4	71.3	77.5	75.8	74.1	74.6	75.8

Table 3: PDB 114 chains used in this work.

1a6g	1a6j	1a7g	1ail	1akh	1am7	1avs	1b2v
1b56	1bdo	1bed	1bgf	1bja	1by9	1byf	1c44
1cex	1cy5	1dfu	1dhn	1dqe	1dtl	1dyt	1e0c
1edh	1ejf	1ekg	1epf	1ew4	1f2l	1f35	1f3v
1f80	1F8H	1fdq	1ff3	1fil	1g6a	1g6h	1gaw
1gns	1gnu	1go4	1gwy	1gwy	1h4a	1h70	1hcb
1hfc	1hh8	1hrh	1hsl	1huu	1i4f	1ifo	1iho
1iko	1iw0	1iwm	1j1v	1j54	1j7d	1j97	1jr1
1jiw	1jr2	1jl3	1jrl	1jhf	1k82	1l0s	1l1d
1l6x	1lfo	1ljp	1lld	1m1f	1ml4	1mo1	1mxe
1naq	1ng2	1o15	1o5u	1oqr	1osp	1php	1ppf
1pz4	1q4r	1qav	1qfj	1qg7	1qog	1qst	1r5r
1rro	1rsy	1scj	1slm	1snc	1t15	1tkv	1tn3
1tph	1umu	1uoh	1uuh	1uv0	1vap	1vjh	1ycq
1ze3	256b						

will become a new parameter in prediction of the protein supersecondary structures in the near future.

Appendix

See Table 3.

Conflict of Interests

The author declares that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The author is grateful to the anonymous reviewers for their valuable suggestions and comments, which have led to the improvement of this paper. The work was supported by Inner Mongolia Agriculture University PhD Research Fund (no. BJ08-30) and Basic Science of Inner Mongolia Agriculture University Research Fund (no. JC2013004).

References

[1] T. Blundell, D. Carney, S. Gardner et al., "Knowledge-based protein modelling and design," *European Journal of Biochemistry*, vol. 172, no. 3, pp. 513–520, 1988.

[2] H. J. Dyson and P. E. Wright, "Peptide conformation and protein folding," *Current Opinion in Structural Biology*, vol. 3, no. 1, pp. 60–65, 1993.

[3] Z. Sun, X. Rao, L. Peng, and D. Xu, "Prediction of protein supersecondary structures based on the artificial neural network method," *Protein Engineering*, vol. 10, no. 7, pp. 763–769, 1997.

[4] K. C. Chou, "Prediction of beta-turns in proteins," *Journal of Peptide Research*, vol. 49, pp. 120–144, 1997.

[5] K.-C. Chou and J. R. Blinn, "Classification and prediction of β-turn types," *Journal of Protein Chemistry* , vol. 16, no. 6, pp. 575–595, 1997.

[6] K.-C. Chou, "Prediction of tight turns and their types in proteins," *Analytical Biochemistry*, vol. 286, no. 1, pp. 1–16, 2000.

[7] K.-C. Chou, "Prediction and classification of α-turn types," *Biopolymers*, vol. 42, no. 7, pp. 837–853, 1997.

[8] X. de la Cruz, E. G. Hutchinson, A. Shepherd, and J. M. Thornton, "Toward predicting protein topology: an approach to identifying β hairpins," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 17, pp. 11157–11162, 2002.

[9] X. Z. Hu and Q. Z. Li, "Prediction of the β-hairpins in proteins using support vector machine," *Protein Journal*, vol. 27, no. 2, pp. 115–122, 2008.

[10] D. S. Zou, Z. S. He, J. Y. He, and Y. Xia, "Supersecondary structure prediction using Chou's pseudo amino acid composition," *Journal of Computational Chemistry*, vol. 32, no. 2, pp. 271–278, 2011.

[11] D. A. Case, "The use of chemical shifts and their anisotropies in biomolecular structure determination," *Current Opinion in Structural Biology*, vol. 8, no. 5, pp. 624–630, 1998.

[12] D. S. Wishart and D. A. Case, "Use of chemical shifts in macromolecular structure determination," *Methods in Enzymology*, vol. 338, pp. 3–34, 2001.

[13] A. Cavalli, X. Salvatella, C. M. Dobson, and M. Vendruscolo, "Protein structure determination from NMR chemical shifts," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 104, no. 23, pp. 9615–9620, 2007.

[14] Y. Shen, O. Lange, F. Delaglio et al., "Consistent blind protein structure generation from NMR chemical shift data," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 105, no. 12, pp. 4685–4690, 2008.

[15] H. Lin, C. Ding, Q. Song et al., "The prediction of protein structural class using averaged chemical shifts," *Journal of Biomolecular Structure & Dynamics*, vol. 29, no. 6, pp. 643–649, 2012.

[16] M. Mechelke and M. Habeck, "A probabilistic model for secondary structure prediction from protein chemical shifts," *Proteins*, vol. 81, no. 6, pp. 984–993, 2013.

[17] S. P. Mielke and V. V. Krishnan, "Protein structural class identification directly from NMR spectra using averaged chemical shifts," *Bioinformatics*, vol. 19, no. 16, pp. 2054–2064, 2003.

[18] A. Pastore and V. Saudek, "The relationship between chemical shift and secondary structure in proteins," *Journal of Magnetic Resonance*, vol. 90, no. 1, pp. 165–176, 1990.

[19] Y. Wang, "Secondary structural effects on protein NMR chemical shifts," *Journal of Biomolecular NMR*, vol. 30, no. 3, pp. 233–244, 2004.

[20] W. S. Mao, P. S. Cong, Z. H. Wang, L. J. Lu, Z. L. Zhu, and T. H. Li, "NMRDSP: an accurate prediction of protein shape strings from NMR chemical shifts and sequence data," *PLoS ONE*, vol. 8, no. 12, Article ID e83532, 2013.

[21] H. Zhang, S. Neal, and D. S. Wishart, "RefDB: a database of uniformly referenced protein chemical shifts," *Journal of Biomolecular NMR*, vol. 25, no. 3, pp. 173–195, 2003.

[22] N. Fernandez-Fuentes, A. Hermoso, J. Espadaler, E. Querol, F. X. Aviles, and B. Oliva, "Classification of common functional loops of kinase super-families," *Proteins*, vol. 56, no. 3, pp. 539–555, 2004.

[23] G. Wang and R. L. Dunbrack Jr., "PISCES: recent improvements to a PDB sequence culling server," *Nucleic Acids Research*, vol. 33, no. 2, pp. W94–W98, 2005.

[24] Y. Feng and L. Luo, "Use of tetrapeptide signals for protein secondary-structure prediction," *Amino Acids*, vol. 35, no. 3, pp. 607–614, 2008.

[25] K.-C. Chou and H.-B. Shen, "Cell-PLoc: a package of Web servers for predicting subcellular localization of proteins in various organisms," *Nature Protocols*, vol. 3, no. 2, pp. 153–162, 2008.

[26] K.-C. Chou, "Some remarks on protein attribute prediction and pseudo amino acid composition," *Journal of Theoretical Biology*, vol. 273, no. 1, pp. 236–247, 2011.

[27] M. Esmaeili, H. Mohabatkar, and S. Mohsenzadeh, "Using the concept of Chou's pseudo amino acid composition for risk type prediction of human papillomaviruses," *Journal of Theoretical Biology*, vol. 263, no. 2, pp. 203–209, 2010.

[28] 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 and Peptide Letters*, vol. 19, no. 4, pp. 411–421, 2012.

[29] C. Ding, L.-F. Yuan, S.-H. Guo, H. Lin, and W. Chen, "Identification of mycobacterial membrane proteins and their types using over-represented tripeptide compositions," *Journal of Proteomics*, vol. 77, pp. 321–328, 2012.

[30] C. Chen, Z.-B. Shen, and X.-Y. Zou, "Dual-layer wavelet SVM for predicting protein structural class via the general form of Chou's pseudo amino acid composition," *Protein and Peptide Letters*, vol. 19, no. 4, pp. 422–429, 2012.

[31] K.-C. Chou and H.-B. Shen, "Plant-mPLoc: a top-down strategy to augment the power for predicting plant protein subcellular localization," *PLoS ONE*, vol. 5, no. 6, Article ID e11335, 2010.

[32] W. Chen, P.-M. Feng, H. Lin, and K.-C. Chou, "IRSpot-PseDNC: identify recombination spots with pseudo dinucleotide composition," *Nucleic Acids Research*, vol. 41, no. 6, article e68, 2013.

[33] W. Chen, H. Lin, P.-M. Feng, C. Ding, Y.-C. Zuo, and K.-C. Chou, "iNuc-PhysChem: a sequence-based predictor for identifying nucleosomes via physicochemical properties," *PLoS ONE*, vol. 7, no. 10, Article ID e47843, 2012.

[34] H. Lin, W. Chen, L.-F. Yuan, Z.-Q. Li, and H. Ding, "Using over-represented tetrapeptides to predict protein submitochondria locations," *Acta Biotheoretica*, vol. 61, no. 2, pp. 259–268, 2013.

[35] H. Lin, C. Ding, L.-F. Yuan et al., "Predicting subchloroplast locations of proteins based on the general form of Chou's pseudo amino acid composition: approached from optimal tripeptide composition," *International Journal of Biomathematics*, vol. 6, no. 2, Article ID 13500034, 2013.

[36] W.-Z. Lin, J.-A. Fang, X. Xiao, and K.-C. Chou, "ILoc-Animal: a multi-label learning classifier for predicting subcellular localization of animal proteins," *Molecular BioSystems*, vol. 9, no. 4, pp. 634–644, 2013.

[37] X. Xiao, P. Wang, W.-Z. Lin, J.-H. Jia, and K.-C. Chou, "IAMP-2L: a two-level multi-label classifier for identifying antimicrobial peptides and their functional types," *Analytical Biochemistry*, vol. 436, no. 2, pp. 168–177, 2013.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 236717, 17 pages
http://dx.doi.org/10.1155/2014/236717

Research Article

An Empirical Study of Different Approaches for Protein Classification

Loris Nanni,[1] Alessandra Lumini,[2] and Sheryl Brahnam[3]

[1] *Dipartimento di Ingegneria dell'Informazione, Via Gradenigo 6/A, 35131 Padova, Italy*
[2] *DISI, Università di Bologna, Via Venezia 52, 47521 Cesena, Italy*
[3] *Computer Information Systems, Missouri State University, 901 South National, Springfield, MO 65804, USA*

Correspondence should be addressed to Sheryl Brahnam; sbrahnam@missouristate.edu

Received 24 March 2014; Revised 5 May 2014; Accepted 7 May 2014; Published 15 June 2014

Academic Editor: Wei Chen

Many domains would benefit from reliable and efficient systems for automatic protein classification. An area of particular interest in recent studies on automatic protein classification is the exploration of new methods for extracting features from a protein that work well for specific problems. These methods, however, are not generalizable and have proven useful in only a few domains. Our goal is to evaluate several feature extraction approaches for representing proteins by testing them across multiple datasets. Different types of protein representations are evaluated: those starting from the position specific scoring matrix of the proteins (PSSM), those derived from the amino-acid sequence, two matrix representations, and features taken from the 3D tertiary structure of the protein. We also test new variants of proteins descriptors. We develop our system experimentally by comparing and combining different descriptors taken from the protein representations. Each descriptor is used to train a separate support vector machine (SVM), and the results are combined by sum rule. Some stand-alone descriptors work well on some datasets but not on others. Through fusion, the different descriptors provide a performance that works well across all tested datasets, in some cases performing better than the state-of-the-art.

1. Introduction

The explosion of protein sequences generated in the postgenomic era has not been followed by an equal increase in the knowledge of protein biological attributes, which are essential for basic research and drug development. Since manual classification of proteins by means of biological experiments is both time-consuming and costly, much effort has been applied to the problem of automating this process using various machine learning algorithms and computational tools for fast and effective classification of proteins given their sequence information [1]. According to [2], a process designed to predict an attribute of a protein based on its sequence generally involves the following procedures: (1) constructing a benchmark dataset for testing and training machine learning predictors, (2) formulating a protein representation based on a discrete numerical model that is correlated with the attribute to predict, (3) proposing a powerful machine learning approach to perform the prediction, (4) evaluating the accuracy of the method according to a fair testing protocol, and (5) establishing a user-friendly webserver accessible to the public.

In this work we are mainly interested in the second procedure, that is, in the definition of a discrete numerical representation for a protein. Since many different representations have been proposed in the literature, it would be valuable to investigate which of these are most useful for the specific applications, such as subcellular localization and protein-protein interactions [3–6], to which these representations are applied [7, 8].

Two kinds of models are typically employed to represent protein samples: the sequential model and the discrete model. The most widely used sequential model is based on the entire amino-acid sequence of a protein, expressed by the sequence of its residues, with each one belonging to one of the 20 native amino-acid types:

$$P = (p_1, p_2, \ldots, p_N) \quad \text{where } p_i \in \mathscr{A} = [\mathrm{A}, \mathrm{C}, \mathrm{D}, \ldots, \mathrm{Y}]. \tag{1}$$

This kind of approach, whose length varies depending on the protein structure, is not suited for most machine learning predictors and fails to work when the query protein does not have significant sequence similarity to any attribute-known proteins.

More suitable for machine learning purposes are protein discrete models, which fall into two main classes. The first class includes the simple amino-acid composition (AAC) and approaches that are based on the AAC-discrete model, such as Chou's pseudo-amino-acid composition (PseAAC) [3–5], which is arguably one of the most popular methods for extracting features from proteins. This first class includes techniques based on vector representations of the protein, that is, where a protein sequence $P = (p_1, p_2, \ldots, p_N)$ is represented by a vector $\in \Re^N$. In [9] AAC is a vector of length 20 that includes the normalized occurrence frequencies of the 20 native amino acids. PseAAC [10, 11] expands AAC by retaining information embedded in protein sequences, such as some additional factors that incorporate information regarding a protein's sequential order. Various modes, such as a series of rank-different correlation factors along a protein chain, represent the sequential information. For an excellent history of the development of PseAAC that includes how to use the concept of Chou's PseAAC to develop 16 variant forms of PseAAC, the reader is referred to [11]. In [12] a protein representation based on physicochemical encodings is proposed that combines the value of a given property for an amino acid with its 2 grams representation. Another vector representation is the quasiresidue couple [13], a model which combines information related to a fixed physicochemical property of the protein with the sequence order effect of the composition of the amino acid. Other approaches belonging to this class of protein representation include dipeptide [14], tripeptide [15], and tetrapeptide [16]. These approaches are based on n-peptide descriptors, where each protein is represented by a vector of length 20^n that includes the normalized occurrence frequencies of the given n-peptide. For reducing the dimensionality of the descriptor, a feature selection algorithm may be used, as in [16].

Before proceeding to the second class of representations based on protein discrete models, it should be noted that a number of different PseAAC methods have been developed for specific applications, such as for predicting certain biological attributes. Some examples include cellular automata image classification [17–20], complexity measure factor [19, 21], gray dynamic model [17, 18], and functional domain composition [20].

The second class of protein feature extraction methods is based on kernels. One of the first kernel-based methods (proposed for remote homology detection) is the Fisher kernel [22]. A kernel that performs equally well but with lower computational cost is the mismatch string kernel proposed in [23, 24] that measures sequence similarity based on shared occurrences of subsequences. In [25] another class of kernels is proposed for vectors derived from the k-peptide vector, mapped by a matrix of high-scored pairs of k-peptides measured by BLOSUM62 scores. These kernel functions train a support vector machine (SVM). In [26] the biobasis function neural network trains sequence distances obtained using sequence alignment.

Aside from using AAC and protein properties for protein representation, several high performing features have also been derived from the position-specific scoring matrix (PSSM) [27]. PSSM describes a protein starting from the evolutionary information contained in a PSI-BLAST similarity search. For a survey of research using descriptors extracted from PSSM, see [28].

The main drawback of the methods based on structural or sequential features is that they only focus on the local variation of the protein itself. For this reason, cellular interactions of proteins have been investigated, as in [29], for solving some particular problems. In [30], the combination of traditional sequential features of the amino acid, such as PSSM, and different networks, such as KEGG enrichment scores of the protein neighbors in STRING network [31], were studied. Protein interaction networks have also been examined in [32, 33].

In this study our objective is to search for a general ensemble method that works well across different protein classification datasets. To accomplish our goal we focus on structural and sequential features. We are motivated to study protein classification methods that generalize well because such systems offer the potential of deepening our understanding of protein representation and of speeding up real world development in new areas involving protein classification. Such investigations also have the potential of promoting and laying the foundations for the development of more robust and powerful classification systems.

The present paper provides an in-depth look at the protein representations that have led to the evolution of some of our previous work in this area.

(i) Reference [12], where an ensemble of approaches based on the amino-acid sequence was proposed.

(ii) Reference [34], where several feature extraction methods based on the calculation of texture descriptors starting from a wavelet representation of the protein were proposed.

(iii) Reference [28], where several feature extraction methods based on the calculation of PSSM were compared.

In this work we explain and compare several state-of-the-art descriptors and some new variants starting from different types of protein representations: the PSSM, the amino-acid sequence, two matrix representations of the protein, and the 3D tertiary structure representations of the protein. We also develop a new ensemble (based on the above cited works) that performs well across multiple datasets, with our ensemble obtaining state-of-the-art performances on several datasets. For the sake of fairness, we use the same ensemble with the same set of parameters (i.e., the same weights in the weighted sum rule) across all tested datasets.

The remainder of this paper is organized as follows. In Section 2 the entire ensemble system is described, including all the protein representation approaches and feature extraction methods we use, all of which are detailed in Sections 3

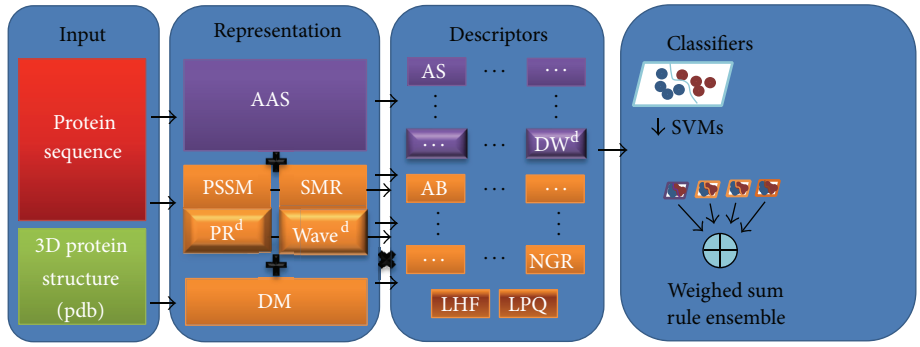


Figure 1: Schema of the proposed method.

and 4, respectively. In Section 5 the datasets used for experiments are described, and the results of several experiments conducted both with stand-alone approaches and ensembles of methods are reported and discussed. Finally, in Section 6, a number of conclusions are drawn and some future lines of research are proposed.

2. A General Machine Learning Approach for Protein Classification

Since several problems in the bioinformatics literature require the classification of proteins, a number of datasets are available for experiments, and recent research has focused on finding a compact and effective representation of proteins [3–5, 12], possibly based on a fixed-length descriptor so that the classification problem can be solved by a machine learning approach. In this work several solutions are evaluated based on a general representation approach that is coupled with a fixed-length encoding scheme so that it can be used with a general purpose classifier.

The classification system illustrated in Figure 1 is an ensemble of classifiers trained using the different descriptors. Five types of protein representations are considered for all the datasets: the simple amino-acid sequence (AAS), PSSM of the proteins, substitution matrix representation (SMR), physicochemical property response matrix (PR), and the wavelet image (WAVE). A detailed description of each representation is given in Section 3. From each representation several descriptors are extracted, which we describe in Section 4. Some descriptors are extracted multiple times, once for each physicochemical property considered in the extraction process. The set of physicochemical properties is obtained from the amino-acid index [35] database (available at http://www.genome.jp/dbget/aaindex.html but note that we do not consider properties where the amino acids have value 0 or 1). An amino-acid index is a set of 20 numerical values representing the different physicochemical properties of amino acids. This database currently contains 544 indices and 94 substitution matrices, but a reduced number of properties are enough for classification task. According to [12], a selection of 25 properties is performed to reduce the number of properties considered in the feature extraction process.

The combination of representation and descriptors is summarized in Table 1, with the size of each descriptor reported in Table 1.

Each descriptor is used to train a general purpose classifier. SVMs are used for the classification task due to their wide diffusion and high generalization ability. SVMs derive from the field of statistical learning theory [36] and are binary-class prediction methods. The basic idea behind SVMs is to find the equation of a hyperplane (called the margin) that divides the training set into two classes so that all the points of the same class are located on the same side while simultaneously maximizing the distance between the two classes and the margin. In those problems where a linear decision boundary does not exist, kernel functions are used to project the data onto a higher-dimensional feature space so that they can be separated by a hyperplane. Typical kernels include polynomial kernels and the radial basis function kernel. SVMs can be easily extended to multiclass problems by considering the one-versus-all classification task. In the experiments reported in this work, all features used for training an SVM are linearly normalized to $[0, 1]$ considering the training data. In each dataset the SVM is tuned considering only the training data (in other words, the test is blind) using a grid search approach. In our system, SVM is implemented as in the LibSVM toolbox (available at http://www.csie.ntu.edu.tw/?cjlin/libsvm).

The ensemble approaches based on the fusion of different descriptors are obtained by combining the pool of SVMs by weighted sum rule; this rule simply sums the scores obtained by the pool of SVMs classifiers, where to each SVM a given weight is applied.

3. Protein Representation Approaches

3.1. A Sequential Representation for Proteins: Amino-Acid Sequence (AAS). The most widely used representation for proteins is a sequential model of the amino-acid sequence:

$$P = (p_1, p_2, \ldots, p_N), \tag{2}$$

where $p_i \in \mathscr{A} = [\mathrm{A}, \mathrm{C}, \mathrm{D}, \ldots, \mathrm{Y}]$ and $\mathscr{A}$ is the set of the 20 native amino-acid types. Several studies [35] have shown that AAS coupled with other information related to the physicochemical properties of amino acids produces

Table 1: Summarized description of the datasets (if available, the number of training and independent samples is given in column "number of samples"). The column BKB reports whether it is possible from the dataset to obtain the PDB of the proteins for extracting the backbone structure.

Name	Short name	Number of samples	Number of classes	Protocol	BKB
Membrane subcellular	MEM	3249 + 4333	8	Independent training and testing sets	NO
Human pairs	HU	1882	2	10-fold cross validation	NO
Protein fold	PF	698	27	Independent training and testing sets	YES
GPCR	GP	730	2	10-fold cross validation	NO
GRAM	GR	452	5	10-fold cross validation	NO
Viral	VR	112	4	10-fold cross validation	NO
Cysteines	CY	957	3	10-fold cross validation	YES
SubCell	SC	121	3	10-fold cross validation	YES
DNA-binding proteins	DNA	349	2	10-fold cross validation	YES
Enzyme	ENZ	1094	6	10-fold cross validation	YES
GO dataset	GO	168	4	10-fold cross validation	YES
Human interaction	HI	8161	2	10-fold cross validation	NO
Submitochondria locations	SL	317	3	10-fold cross validation	NO
Virulent independent set 1	VI1	2055 + 83	2	Independent training and testing sets	NO
Virulent independent set 2	VI2	2055 + 284	2	Independent training and testing sets	NO
Adhesins	AD	2055 + 1172	2	Independent training and testing sets	NO

many useful descriptors, some of which will be described in Section 4.

3.2. A Matrix Representation for Proteins: Position-Specific Scoring Matrix (PSSM). The PSSM representation of a protein, first proposed in [27], is obtained from a group of sequences previously aligned by structural or sequence similarity. Such representations can be calculated using the application PSI-BLAST (position-specific iterated BLAST), which compares PSSM profiles for detecting remotely related homologous proteins or DNA.

The PSSM representation considers the following parameters.

(1) Position: the index of each amino-acid residue in a sequence after multiple sequence alignment.

(2) Probe: a group of typical sequences of functionally related proteins already aligned by sequence or structural similarity.

(3) Profile: a matrix of 20 columns corresponding to the 20 amino acids.

(4) Consensus: the sequence of amino-acid residues most similar to all the alignment residues of probes at each position. The consensus sequence is generated by selecting the highest score in the profile at each position.

A PSSM representation for a given protein of length N is an $N \times 20$ matrix, whose elements PSSM(i, j) are calculated as

$$\mathrm{PSSM}(i,j) = \sum_{k=1}^{20} w(i,k) \times Y(j,k), \quad i = 1,\ldots,N, \quad j = 1,\ldots,20, \tag{3}$$

where $w(i,k)$ is the ratio between the frequency of the kth amino acid at the position i of the probe and total number of probes and $Y(j,k)$ is the value of Dayhoff's mutation matrix between the jth and kth amino acids ($Y(j,k)$ is a substitution matrix). A substitution matrix describes the rate at which one character in a protein sequence changes to other character states over time. Substitution matrices are usually seen in the context of amino acid or DNA sequence alignments, where the similarity between sequences depends on their divergence time and the substitution rates as represented in the matrix.

Small values of PSSM(i, j) indicate weakly conserved positions and large values indicate strongly conserved positions. In our study, we used PSI-BLAST which can be called from MATLAB for extracting PSSM using the command system ("blastpgp.exe-i input.txt-d swissprot-Q output.txt-j 3," where "input.txt" is the protein sequence and "output.txt" contains the PSSM matrix to create PSSMs for each protein sequence).

3.3. A Matrix Representation for Proteins: Substitution Matrix Representation (SMR). In [28] a variant of a representation method called the substitution matrix representation (SMR) proposed by [37] is developed where the SMR for a given protein $P = (p_1, p_2, \ldots, p_N)$ is a $N \times 20$ matrix obtained as

$$\mathrm{SMR}^d(i,j) = M(p_i, j), \quad i = 1,\ldots,N, \quad j = 1,\ldots,20, \tag{4}$$

where $M(a, j)$ is a 20×20 substitution matrix whose element $M_{a,j}$ represents the probability of amino acid a mutating to amino acid j during the evolution process (note: the MATLAB code for this representation is available at http://bias.csr.unibo.it/nanni/SMR.rar).

In the experiments reported below, 25 random physicochemical properties have been selected to create an ensemble (labelled SMR) of SMR^d-based predictors.

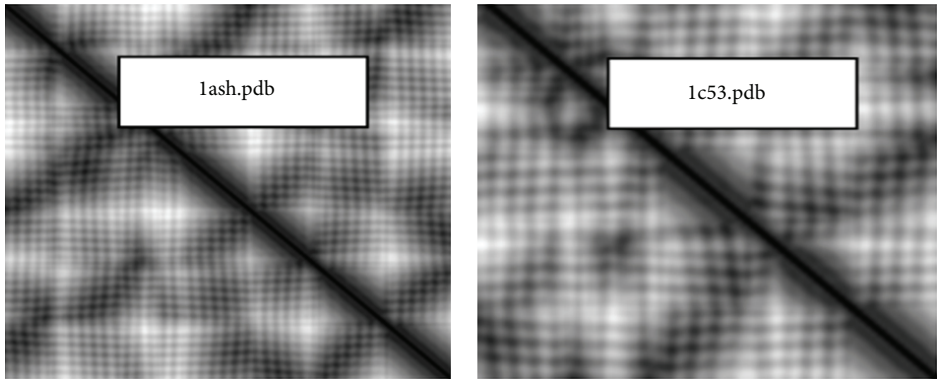


Figure 2: DM images extracted from 2 sample proteins of the DNA dataset.

3.4. A Matrix Representation for Proteins: Physicochemical Property Response Matrix (PR). In [34] a representation matrix based on physicochemical properties is proposed. First the physicochemical property response matrix $\mathrm{PRM}^d(i,j) \in \Re^{N\times N}$ is obtained for a given protein $P = (p_1, p_2, \ldots, p_N)$ by selecting a physicochemical property d and setting the value of the element $\mathrm{PRM}^d(i,j)$ to the sum of the value of the physicochemical property d of the amino-acid in position i of the protein and the value of the physicochemical property of the amino-acid in position j. Consider

$$\mathrm{PRM}^d(i,j) = \mathrm{index}(p_i, d) + \mathrm{index}(p_j, d), \quad i,j = 1,\ldots,N, \tag{5}$$

where $\mathrm{index}(a,d)$ returns the value of the property d for the amino acid a.

Then PRM^d is handled as an image and resized to 250×250 if larger to obtain the final matrix PR^d. In the experiments reported below, 25 random physicochemical properties have been selected to create an ensemble (PR) of PR^d-based predictors.

3.5. A Matrix Representation for Proteins: Wavelet (WAVE). Wavelets are very useful descriptors with lots of different applications. First [38] then later [34] have suggested using wavelets as a method to represents proteins. Since wavelet encoding requires a numerical representation, the protein sequence is first numerically encoded by substituting each amino acid with a value of a given physicochemical property d. Then the Meyer continuous wavelet is applied to the wavelet transform coefficients (labelled WAVE^d). Features are extracted by considering 100 decomposition scales. In the experiments reported below, 25 random physicochemical properties have been selected to create an ensemble WAVE of 25 WAVE^d-based predictors.

3.6. A Matrix Representation for Proteins: 3D Tertiary Structure (DM). The 3D tertiary structure representation for proteins is based on the protein backbone (i.e., the sequence of its C_α atoms) to characterize the whole protein structure [39, 40]. Given a protein P and its backbone $B = (\mathbf{Coor}_1, \mathbf{Coor}_2, \ldots, \mathbf{Coor}_M)$ (obtained as the 3D coordinates of its M C_α atoms), a distance matrix DM is defined as

$$\mathrm{DM}(i,j) = \mathrm{dist}\left(\mathbf{Coor}_i, \mathbf{Coor}_j\right), \quad 1 \le i,\ j \le M, \tag{6}$$

where $\mathrm{dist}(\cdot)$ is the Euclidean distance. (Note: the MATLAB code for extracting the distance matrix is available at http://bias.csr.unibo.it/nanni/DM.zip).

As with the other matrix representations introduced above, DM is regarded as a grayscale image, which is used to extract texture descriptors, as illustrated in Figure 2.

4. Protein Feature Extraction Approaches

In this section the approaches used to extract descriptors from the different representations introduced above are described. Most of the descriptors extracted from the primary representation are based on substituting the letter representation of an amino acid with its value of a fixed physicochemical property. In order to make the result independent on the selected property, a selection on 25 or 50 properties is done by random, and the resulting descriptors are used to train an ensemble of SVM classifiers.

4.1. A Descriptor for Primary Representation: Amino-Acid Composition (AS). Amino-acid composition is the simpler method for extracting features from a protein representation that is based on counting the fraction of a given amino acid:

$$\mathrm{AS}(i) = \frac{h(i)}{N}, \quad i \in [1,\ldots,20], \tag{7}$$

where $h(i)$ counts the number of occurrences of a given amino acid in a protein sequence of length N.

4.2. A Descriptor for Primary Representation: 2 Grams (2G). The standard 2 grams descriptor is a vector of 20^2 values,

each counting the number of occurrences of a given couple of amino acids in a protein sequence. Consider

$$2\mathrm{G}(k) = \left(\frac{h(i,j)}{N}\right), \quad i,j \in [1,\ldots,20],\ k = j + 20 \times (i-1), \tag{8}$$

where the function $h(i, j)$ counts the number of occurrences of a given couple of amino acids (i, j) in a protein sequence of length N. The 2G descriptor is a 400-dimensional vector.

4.3. A Descriptor for Primary Representation: Quasiresidue Couple (QRC). Quasiresidue Couple is a method for extracting features from the primary sequence of a protein [12] that is inspired by Chou's quasi-sequence-order model and Yuan's Markov chain model [41]. The original residue couple model was designed to represent both the information of the amino-acid composition and the order of the amino acids in the protein sequences. The quasiresidue couple descriptor is obtained by selecting a physicochemical property d and combining its values with each nonzero entry in the residue couple. The parameter m denotes the order of the composition (values $m \leq 3$ are considered enough to represent a sequence).

The QRC model (of order $m \leq 3$) for a physicochemical property d is given by

$$\begin{aligned}\mathrm{QRC}_m^d(k) = \frac{1}{N-m}\sum_{n=1}^{N-m} H_{i,j}(n, n+m, d) \\ + H_{j,i}(n+m, n, d),\end{aligned}$$
$$i,j \in [1,\ldots,20], \quad k = j + 20(i-1), \tag{9}$$

where i and j are the 20 different amino acids, N is the length of the protein, the function index(i, d) returns the value of the property d for the amino acid i, and the function $H_{i,j}(a, b, d) = \mathrm{index}(i, d)$, if $p_a = i$ and $p_b = j$, $H_{i,j}(a, b, d) = 0$ otherwise.

In our experiments, the QRC^d features are extracted for m ranging from 1 to 3 and concatenated into a 1200-dimensional vector. In the experiments reported below, 25 random physicochemical properties have been selected to create an ensemble of QRC descriptors (Note: the MATLAB code for QRC is available at http://bias.csr.unibo.it/nanni/QRcouple2.zip).

4.4. A Descriptor for Primary Representation: Autocovariance Approach (AC). The autocovariance approach [42] is a sequence-based variant of Chou's pseudo-amino-acid composition, which extracts a set of pseudo-amino-acid-based features (extracted by the MATLAB code shared by the original authors) from a given protein as the concatenation of the 20 standard amino-acid composition values and m values reflecting the effect of sequence order. The parameter m denotes the maximum distance between two considered amino acids i, j (set to 20 in the tests reported below).

Given a protein $P = (p_1, p_2, \ldots, p_N)$ and fixing a physicochemical property d, the autocovariance descriptor is $\mathrm{AC}^d \in \Re^{20+m}$:

$$\mathrm{AC}^d(i) = \begin{cases} \dfrac{h(i)}{N}, \\ \quad i \in [1,\ldots,20], \\ \displaystyle\sum_{k=1}^{N-i+20} \dfrac{(\mathrm{index}(p_k, d) - \mu_d)\cdot(\mathrm{index}(p_{k+i-20}, d) - \mu_d)}{\sigma_d \cdot (N - i + 20)}, \\ \quad i \in [21,\ldots,20+m], \end{cases} \tag{10}$$

where the function index(i, d) returns the value of the property d for the amino acid i, the function $h(i)$ counts the number of occurrences of a given amino acid in a protein sequence, and μ_d and σ_d are normalization factors denoting mean and the variance of d on the 20 amino acids:

$$\begin{aligned}\mu_d &= \frac{1}{20}\sum_{i=1}^{20}\mathrm{index}(i, d), \\ \sigma_d &= \frac{1}{20}\sum_{i=1}^{20}(\mathrm{index}(i, d) - \mu_d)^2.\end{aligned} \tag{11}$$

In the experiments reported below, 25 random physicochemical properties have been selected to create an ensemble of 25 AC descriptors (Note: the MATLAB code for AC is available at http://bias.csr.unibo.it/nanni/EstraggoFeaturesAC.rar).

4.5. A Descriptor for Primary Representation: AAIndexLoc (AA). The AAIndexLoc is a descriptor proposed in [43]. AAIndexLoc is composed of the following features.

(i) *Amino-acid composition (20 features)*: this is a fraction of a given amino acid.

(ii) *Weighted amino-acid composition (20 features)*: this is defined for a given amino acid i as (amino-acid composition of i) ×index(i, d).

(iii) *Five-level grouping composition (25 features)*: the result of a five-level dipeptide composition applied to a classification of the amino acids by k-means (into five groups) considering their amino-acid index values. The five-level dipeptide composition is defined as the composition of the occurrence of two consecutive groups; see [43] for more details.

In the experiments reported below, 25 random physicochemical properties have been selected to create an ensemble of 25 AA^d descriptors.

4.6. A Descriptor for Primary Representation: Global Encoding (GE). Global encoding is a descriptor proposed in [44] that is based on a classification (labelled here as A) of amino acids into six classes: A1 = {A, V, L, I, M, C}, A2 = {F, W, Y, H}, A3 = {S, T, N, Q}, A4 = {K, R}, A5 = {D, E}, and A6 = {G, P}. The final descriptor GE is obtained by extracting measures

from the 10 *characteristic sequences,* CS_{Pt}, obtained from each of the 10 different *partitions* Pt of A into 2 subsets of three classes (e.g., Pt = {(A1, A2, A3), (A4, A5, A6)} is one of the 10 partitions). CS_{Pt} is obtained by transforming the protein into a numerical sequence where a given amino acid is represented by 1 if it belongs to the first class of the partition (i.e., (A1, A2, A3)) and 0 otherwise. The two sets of measures used to define GE are the frequency of 0s and 1s in each CS_{Pt} and the frequency of transitions (i.e., number of 1s followed by a 0, and vice versa).

4.7. Descriptor for Primary Representation: Physicochemical 2 Grams (P2G). The physicochemical 2 grams [13] are descriptors that combine the value of a given physicochemical property d for an amino acid together with the 2 grams representation of a protein. The standard 2 grams representation is a vector of 20^2 values, each counting the number of occurrences of a given couple of amino acids in a protein sequence. The physicochemical 2 grams (P2G) for a given physiochemical property d is defined as

$$\mathrm{P2G}^d(k) = \left(\frac{h(i,j) \cdot \mathrm{index}(i,d)}{N-1}, \frac{h(i,j) \cdot \mathrm{index}(j,d)}{N-1} \right), \quad i,j \in [1,\ldots,20], \quad k = j + 20(i-1), \tag{12}$$

where i and j denote the 20 different amino acids, N is the length of the protein, the function index(i, d) returns the value of the property d for the amino acid I, and the function $h(i, j)$ counts the number of occurrences of a given couple of amino acids (i, j) in a protein sequence. The $\mathrm{P2G}^d$ descriptor is an 800-dimensional vector. In the experiments reported below, 25 random physicochemical properties have been selected to create an ensemble of 25 $\mathrm{P2G}^d$ descriptors.

4.8. A Descriptor for Primary Representation: N-Gram (NG). The N-gram descriptor is similar to the standard 2 grams descriptor but is obtained on a different N-peptide composition using different amino-acid alphabets. In this work 5 alphabets proposed in [45] are considered, and we train five different SVMs. Each classifier is trained using a different N-peptide composition with different amino-acid alphabets:

(i) A1 = G–I–V–F–Y–W–A–L–M–E–Q–R–K–P–N–D–H–S–T–C,

(ii) A2 = LVIM–C–A–G–S–T–P–FY–W–E–D–N–Q–KR–H,

(iii) A3 = LVIMC–AG–ST–P–FYW–EDNQ–KR–H,

(iv) A4 = LVIMC–ASGTP–FYW–EDNQ–KRH,

(v) A5 = LVIMC–ASGTP–FYW–EDNQKRH.

Each protein is first translated according to the 5 alphabets. Then 2 gram representations are calculated from A1 to A2 languages, and the 3 gram representations are calculated from A3 to A4 to A5. The five descriptors are

$$\mathrm{NG}^{\mathscr{A}} \quad \mathscr{A} \in [\mathrm{A1}, \mathrm{A2}, \mathrm{A3}, \mathrm{A4}, \mathrm{A5}], \tag{13}$$

having dimensions $\#\mathscr{A}^n$, where $n = 2$ for 2 gram representations and $n = 3$ for 3 gram representations. The 5 representations are fused together by weighted sum rule (with weights 1, 1, 1, 0.5, and 0.25).

4.9. A Descriptor for Primary Representation: Split Amino-Acid Composition (SAC). Split amino-acid composition is a descriptor proposed by [46] that is based on the subdivision of the protein sequence into parts from which a separate descriptor is calculated (i.e., the standard amino-acid composition for each part). In this work each protein is divided into the following three parts: (i) 20 amino acids of N-terminus, (ii) 20 amino acids of C-terminus, and (iii) the region between these two termini.

4.10. A Descriptor for Primary Representation: Discrete Wavelet (DW). A sequence descriptor based on biorthogonal discrete wavelet is proposed in [34]. Given a protein $P = (p_1, p_2, \ldots, p_N)$ and a fixed physicochemical property d, each amino acid of the sequence is substituted by its value of d:

$$\mathrm{PP}^d(i) = \mathrm{index}(p_i, d), \quad i \in [1,\ldots,N], \tag{14}$$

where the function index(i, d) returns the value of the property d for the amino acid i.

The vector PP^d is then transformed in the wavelet space by a four-scale biorthogonal discrete wavelet. The final descriptor DW^d is obtained as the first five discrete cosine coefficients from the approximation coefficients and the maximum, minimum, mean, and standard deviation values from both detail and approximation coefficients. This choice is motivated by the fact that high-frequency components are noisy; thus the low-frequency components are more useful for the classification task.

In the experiments reported below, 25 random physicochemical properties have been selected to create an ensemble of 25 DW^d descriptors.

4.11. A Matrix-Based Descriptor: Average Blocks (AB). This matrix descriptor was originally proposed in [47] for the PSSM representation of a protein, but it is used for other matrix representations in this work. Average blocks is a fixed-length vector $\mathrm{AB} \in \Re^{400}$ elements obtained as the local average of the input matrix $\mathrm{Mat} \in \Re^{n \times 20}$:

$$\mathrm{AB}(k) = \frac{20}{N} \sum_{z=1}^{N/20} \mathrm{Mat}\left(z + (i-1) * \frac{N}{20}, j\right), \quad i = 1,\ldots,20, \quad j = 1,\ldots,20, \quad k = j + 20 \times (i-1), \tag{15}$$

where k is a linear index used to scan the cells of Mat. Thus the final descriptor is a vector obtained as the average of Mat blocks (each related to the 5% of a sequence).

4.12. A Matrix-Based Descriptor: Single Average (SA). This descriptor [48] is a variant of the previous one and is designed to group together rows related to the same amino acid, thus

considering domains of a sequence with similar conservation rates.

The descriptors SA $\in \Re^{400}$ for a protein $P = (p_1, p_2, \ldots, p_N)$ and its matrix representation Mat is

$$\begin{aligned} &\mathrm{SA}(k) = \mathrm{avg}_{i=1,\ldots,N} \mathrm{Mat}(i,j) * \delta(P(i), \mathscr{A}(z)), \\ &z = 1,\ldots,20, \quad j = 1,\ldots,20, \quad k = j + 20 \times (z-1), \end{aligned} \tag{16}$$

where k is a linear index used to scan the cells of Mat, where $\mathscr{A} = [\mathrm{A}, \mathrm{C}, \mathrm{D}, \ldots, \mathrm{Y}]$ is the ordered set of amino acids, and where $\delta(\cdot,\cdot)$ is the Kronecker delta function.

In this work two variants of the single average descriptor are used: the one described above (labelled SA) and a version including matrix normalization using a sigmoid function by which each element of Mat is scaled to $[0, 1]$ (labelled SAN).

4.13. A Matrix-Based Descriptor: Autocovariance Matrix (AM). The autocovariance matrix is a matrix descriptor proposed in [49] that aims at avoiding the loss of the local sequence-order information. Each column of the input matrix is reduced to a fixed length by autocovariance variables. An autocovariance matrix (AM) describes the average correlation between positions in a series of lags (i.e., the residue number when applied to protein sequences) throughout the protein sequence.

AM can be calculated from an input matrix Mat $\in \Re^{N\times 20}$ as follows:

$$\begin{aligned} \mathrm{AM}(k) = {} & \frac{1}{N-\mathrm{lag}} \sum_{i=1}^{N-\mathrm{lag}} \left(\mathrm{Mat}(i,j) - \frac{1}{N} \sum_{i=1}^{N} \mathrm{Mat}(i,j) \right) \\ & \times \left(\mathrm{Mat}(i+\mathrm{lag}, j) - \frac{1}{N} \sum_{i=1}^{N} \mathrm{Mat}(i,j) \right), \\ & j = 1,\ldots,20, \quad \mathrm{lag} = 1,\ldots,15, \quad k = j + 20 \times (\mathrm{lag} - 1), \end{aligned} \tag{17}$$

where k is a linear index used to scan the cells of Mat, lag denotes the distance between one residue and its neighbors, and N is the length of the sequence.

4.14. A Matrix-Based Descriptor: Pseudo-PSSM (PP). This pseudo-PSSM approach (PP) is one of the most widely used matrix descriptors for proteins (see [47, 50]). Usually applied to the PSSM matrix representation of a protein, PP is extended in this work to SMR. This descriptor is designed to retain information about amino-acid sequence by considering the pseudo-amino-acid composition.

Given an input matrix Mat $\in \Re^{N\times 20}$, the pseudo-PSSM descriptor is a vector PP $\in \Re^{320}$ defined as

$$\mathrm{PP}(k) = \begin{cases} \dfrac{1}{N} \displaystyle\sum_{i=1}^{N} E(i,j), \\ \quad k = 1,\ldots,20, \\ \dfrac{1}{N-\mathrm{lag}} \displaystyle\sum_{i=1}^{N-\mathrm{lag}} \left[E(i,j) - E(i+\mathrm{lag}, j) \right]^2, \\ \quad j = 1,\ldots,20, \ \mathrm{lag} = 1,\ldots,15, \\ \quad k = 20 + j + 20 \cdot (\mathrm{lag} - 1), \end{cases} \tag{18}$$

where k is a linear index used to scan the cells of Mat, where lag denotes the distance between one residue and its neighbors, and where N is the length of the sequence and $E \in \Re^{N\times 20}$, which is a normalized version of Mat defined as

$$\begin{aligned} E(i,j) = {} & \frac{\mathrm{Mat}(i,j) - (1/20) \sum_{v=1}^{20} \mathrm{Mat}(i,v)}{\sqrt{(1/20) \sum_{u=1}^{20} \left(\mathrm{Mat}(i,u) - (1/20) \sum_{v=1}^{20} \mathrm{Mat}(i,v) \right)^2}}, \\ & i = 1,\ldots,N, \quad j = 1,\ldots,20. \end{aligned} \tag{19}$$

4.15. A Matrix-Based Descriptor: Singular Value Decomposition (SVD). Singular value decomposition is a general purpose matrix factorization approach [51] that has many useful applications in signal processing and statistics. In this work SVD is applied to a matrix representation of a protein with the aim of reducing its dimensionality.

Given an input matrix Mat $\in \Re^{N\times M}$, SVD is used to calculate its factorization of the form: Mat $= U\Sigma V$, where Σ is a diagonal matrix whose diagonal entries are known as the singular values of Mat. The resulting descriptor is the ordered set of singular values: SVD $\in \Re^{L}$, where $L = \min\{M, N\}$.

4.16. A Matrix-Based Descriptor: Discrete Cosine Transform (DCT). DCT [52] is a linear separable transformation for converting a signal into elementary frequency components; it is widely used in image compression for its capability to concentrate information into a small number of coefficients. Given an input matrix Mat $\in \Re^{N\times M}$, its DCT transformation is defined as

$$\begin{aligned} \mathrm{DCT}(i,j) = {} & a_i a_j \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \mathrm{Mat}(m,n) \cos \frac{\pi (2m+1) i}{2M} \\ & \times \cos \frac{\pi (2n+1) i}{2N}, \\ & 0 \le i \le M, \quad 0 \le j \le N, \end{aligned} \tag{20}$$

where

$$a_i = \begin{cases} \dfrac{1}{\sqrt{M}} & i = 0, \\ \sqrt{\dfrac{2}{M}} & 1 \le i \le M-1, \end{cases} \qquad a_i = \begin{cases} \dfrac{1}{\sqrt{N}} & j = 0, \\ \sqrt{\dfrac{2}{N}} & 1 \le j \le N-1. \end{cases} \tag{21}$$

In this work the final DCT descriptor is obtained by retaining the first 400 coefficients.

4.17. A Matrix-Based Descriptor: N-Gram Features (NGR). The N-gram descriptor is usually extracted from the primary protein sequence (as already described in Section 4.6). In [53] this descriptor is extracted directly from the PSSM matrix by accumulating the probabilities of each of the N-gram according to the probability information contained in PSSM.

Given an input matrix $\mathrm{Mat} \in \Re^{N\times 20}$ representing the PSSM of a given protein, the frequency of occurrence of transition from ith amino acid to jth amino acid is calculated as follows for 2 grams (BGR) and 3 grams (TGR), respectively:

$$\mathrm{BGR}(l) = \sum_{z=1}^{N-1} \mathrm{Mat}(z,i) \times \mathrm{Mat}(z+1,j),$$

$$i = 1,\ldots,20, \quad j = 1,\ldots,20, \quad l = (i-1) * 20 + j,$$

$$\mathrm{TGR}(l) = \sum_{z=1}^{N-2} \mathrm{Mat}(z,i) \times \mathrm{Mat}(z+1,j) \times \mathrm{Mat}(z+2,k)$$

$$i = 1,\ldots,20, \quad j = 1,\ldots,20, \quad k = 1,\ldots,20,$$

$$l = (i-1) * 400 + (j-1) * 20 + k. \tag{22}$$

4.18. A Matrix-Based Descriptor: Texture Descriptors. A very interesting feature extraction approach for proteins is to treat a protein matrix representation as an image and to use well-known image texture descriptors for extracting features. In this work two high performing descriptors are evaluated: local binary pattern histogram fourier (LHF) descriptors [54] and local phase quantization (LPQ) (Note: the MATLAB code for LBQ is available at http://www.cse.oulu.fi/CMV/Downloads/LPQMatlab) [55]. Both these descriptors are extracted according to a global and a local evaluation (i.e., from the whole image or from subwindows of an image with the results of each concatenated). The feature vectors extracted are labelled, respectively: LHF_G and LPQ_G, when extracted from the whole image, and LHF_L and LPQ_L, when obtained by dividing the image into three equal subwindows and concatenating the resulting feature vectors.

4.18.1. Local Binary Pattern Histogram Fourier (LHF). First proposed by [54], LHF is a rotation invariant image descriptor that is computed from the discrete Fourier transforms of local binary pattern (LBP) histograms. The LHF descriptor computes a noninvariant LBP histogram and constructs rotationally invariant features from the histogram using discrete Fourier transform. The features are invariant to cyclic shifts in the input vector. In this work the final vector is the concatenation of results obtained using the following parameters for LBP computation: ($P = 16$; $R = 2$) and ($P = 8$; $R = 1$).

4.18.2. Local Phase Quantization (LPQ). The LPQ operator [55] is based on the blur invariance property of the Fourier phase spectrum. LPQ uses the local phase information extracted from the two-dimensional short-term Fourier transform (STFT) computed over a rectangular neighborhood defined by each pixel position. After STFT only four complex coefficients are retained corresponding to four fixed two-dimensional frequencies, which are separated into real and imaginary parts and quantized as integers between 0–255 using a binary coding scheme. The final feature vector is a normalized histogram of such coefficients. In this work the final vector is the concatenation of results obtained with two different radii for LPQ computation: radii 3 and 5, using the Gaussian derivative quadrature filter pair for local frequency estimation (Note: we used the MATLAB code for LPQ available at http://www.ee.oulu.fi/mvg/download/lpq/).

5. Experiments

This section reports the results of an experimental evaluation of the protein descriptors on sequence-based protein classification problems performed on several datasets.

5.1. Datasets, Testing Protocols, and Performance Indicators. The proposed approach has been evaluated on the 15 datasets listed below and according to the testing protocols suggested by the developers of the datasets. A brief summary description of each dataset and related testing protocol is reported in Table 2.

Membrane Subcellular (MEM) (See [56]). This is a dataset containing membrane proteins belonging to eight membrane types: (1) single-pass type I transmembrane, (2) single-pass type II, (3) single-pass type III, (4) single-pass type IV, (5) multipass transmembrane, (6) lipid-chain-anchored membrane, (7) GPI-anchored membrane, and (8) peripheral membrane.The objective of this dataset is to classify a given query protein in a given localization. All proteins in the same subcellular location have less than 80% sequence identity. The testing protocol is based on a given subdivision, each of which is divided into a training set (3249 proteins) and a testing set (4333 proteins).

DNA-Binding Proteins (DNA) (See [57]). This is a dataset containing 118 DNA-binding proteins and 231 non-DNA-binding proteins with less than 35% sequence identity between each pair.

Enzyme (ENZ) (See [58]). This is a dataset that was created using the PDB archive and includes proteins annotated as enzymes, specifically, 381 hydrolases and 713 enzymes of different kinds.

GO Dataset (GO) (See [58]). This is a dataset that was extracted from the PDB archive by selecting proteins according to GO annotations. It distinguishes the biological processes "immune response" (33 proteins) and "DNA repair" (43 proteins) and the molecular functions "substrate specific transporter activity" (39 proteins) and "signal transducer activity" (53 proteins). The presence of highly similar proteins within the same class was avoided by removing sequences which had more than 30% identity.

Human Interaction (HI) (See [59]). This is from the positive protein-protein-interaction (PPI) dataset [59], which was

Table 2: Summary of the descriptors (short names are defined in Sections 3 and 4).

Protein representation	Descriptors	
	Descriptor	Size
AAS	AS	20
	2G	400
	QRC[d]	1200
	AC[d]	40
	P2G[d]	800
	AA[d]	65
	GE	480
	NG	400, 225, 512, 125, 64
	SAC	20
	DW[d]	52
PSSM/SMR PR/WAVE (ensembles of 25 PR[d]/WAVE[d]) DM	AB	400
	SAN	400
	SA	400
	AM	300
	PP	320
	SVD	Depends on the input representation
	DCT	400
	LHF_G	176
	LPQ_G	512
	LHF_L	528
	LPQ_L	1536
	BGR	400
	TGR	8000

downloaded from the human protein references database (HPRD, June 2007 version). This version of HPRD contains 38,788 protein-protein pairs of experimentally verified PPIs from 9,630 different human proteins. Self-interactions and duplicate interactions from the dataset were eliminated to obtain 36,630 unique positive protein-protein pairs. The benchmark negative dataset was obtained from the SWISS-PROT database (version 57.3 released on 26-May-2009) by selecting 36,480 protein couples with different cellular compartments that do not interact with each other (see [59] for details). The final dataset was constructed from the original benchmark dataset by excluding proteins having more 25% sequence identity to any of the other proteins using the PISCES program. Accordingly, the number of proteins in the positive dataset was reduced from 9,630 to 2,502, and the number of proteins in the negative dates was reduced from 2,184 to 661 for a total of 3,899 positive samples of protein pairs and 4,262 negative samples of protein pairs. This dataset is not used in all experiments because of its large size (e.g., it is not used in the first experiment to calculate the rank of the compared approaches).

Submitochondria Locations (SL) (See [60]). This is a dataset containing 317 proteins classified into three submitochondria locations: 131 inner membrane proteins; 41 outer membrane proteins; and 145 matrix proteins. To obtain a balance between the homologous bias and the size of the training set, no more than 40% similarity was allowed (i.e., the identity between any 2 sequences in the processed dataset had to be less than 40%).

Virulent Datasets 1 and 2 (VI1, VI2) (See [48]). This is a dataset containing bacterial virulent protein sequences that were retrieved from the SWISS-PROT and VFDB (an integrated and comprehensive database of virulence factors of bacterial pathogens). The two independent sets share the same training set of 1025 virulent and 1030 nonvirulent bacterial sequences. The virulent independent dataset 1 (VI1) contains 83 protein sequences, selected so that no two sequences are more than 40% similar. The virulent independent dataset 2 (VI2) contains 141 virulent and 143 nonvirulent sequences from bacterial pathogens sequences of organisms that were not represented in the training set.

ADHESINS Dataset (AD) (See [48]). This is a dataset containing 469 adhesins and 703 nonadhesins proteins (including several archaebacterial, viral, and yeast nonvirulent proteins). The training set contains 1025 virulent and 1030 nonvirulent bacterial sequences.

GPCR (GP) (See [20]). This is a dataset containing G protein-coupled receptors (GPCR) and non-GPCRs. The aim of this dataset is to identify a query protein as either GPCR or non-GPCR. None of the proteins included have ≥40% pairwise sequence identity to any other in the same subset.

GRAM (GR) (See [61]). This is a dataset containing gram-positive proteins that belong to five subcellular location sites: (1) cell wall, (2) cytoplasm, (3) extracellular, (4) periplasm, and (5) plasma membrane. The aim of this dataset is to classify a given query protein in a given localization. Only those proteins that have <25% sequence identity to any other in a same subcellular location were allowed to be included in the benchmark datasets. In this way, redundancy and homology bias is limited.

Human Protein-Protein Interaction (HU) (See [62]). This is a dataset containing a total of 1882 human protein pairs. Each pair of proteins is labeled as either an *interacting pair* or a *noninteracting pair*.

Viral (VR) (See [63]). This is a dataset containing proteins that belong to four classes: cytoplasm, extracellular, nucleus, and plasma membrane. The aim of this dataset is to classify a given query protein in a given localization. None of the proteins have 25% sequence identity to any other in the same subset (subcellular location). Subcellular localization of viral proteins within a host cell or virus-infected cell is very useful for studying the function of viral proteins as well as designing antiviral drugs.

Protein Fold (PF) (See [15]). The dataset used in this work is a subset of the database presented in [15]. The training set contains 313 proteins and the testing set contains 385 samples from 27 classes. The sequence similarities are less than 35% and 40%, respectively. The testing protocol uses the training

set to build the classifier models and independently uses the testing set to evaluate performance.

Cysteine (CY) (See [64]). This is a dataset that was constructed to predict the state of cysteines. It contains 957 protein sequences, having a sequence identity lower than 25%. The dataset is divided into three classes: proteins that do not have disulfide bonds, which are labeled as "none" and two others that are labelled "mix" and "all" depending on whether all the cysteines have been formed into disulfide bonds or not.

SubCell (SC). This is a dataset containing proteins that belong to three subcellular location sites: (1) nucleus, (2) cytoplasm, and (3) extracellular. Only proteins where the PDB format is available are used. The aim is to classify a given query protein in a given localization.

The testing protocol employed in the experiments depended on the datasets. In cases where the original dataset is not divided into training and testing sets, a 10-fold cross-validation was performed (results averaged on ten experiments); otherwise the subdivision of the training and testing sets was maintained.

Three performance indicators are used in the reported results: the classification accuracy, the area under the ROC curve (AUC), and the statistical rank. The accuracy is the ratio between the number of samples correctly classified and the total number of samples. The ROC curve is a graphical plot of the sensitivity of a binary classifier versus false positives (1—specificity), as its discrimination threshold is varied. AUC [65] is a scalar measure that can be interpreted as the probability that the classifier will assign a lower score to a randomly picked positive pattern than to a randomly picked negative pattern. When a multiclass dataset is used, the one-versus-all area under ROC curve is used [66]. Since AUC is considered one of the most reliable performance indicators [67], internal comparisons are evaluated according to AUC, while accuracy is used to compare results with the literature in those cases where AUC is not reported.

The statistical rank returns the relative position of a method against other tested methods. The average rank is the most stable indicator to average performance on different datasets and is calculated using the Friedman's test (alpha = 0.05) applying the Holm post hoc procedure [68].

5.2. Experimental Results. The first experiment is aimed at comparing all the descriptors detailed in Section 3 and is summarized in Table 2 in terms of the statistical rank evaluated considering all the datasets (excluding HI).

In Table 3 the different methods are evaluated by their average rank, with the best descriptor for each representation highlighted. Notice that the representation method DM is not included in this table; this is because it is available only in a subset of datasets (i.e., where the PDB format is obtainable). Examining Table 3 it is clear that on average PSSM is the best representation. The other representations, however, are useful for building a strong ensemble that outperforms the results of the best stand-alone descriptors, as demonstrated by experiments reported.

Table 3: Comparison among the different feature extractors in terms of the statistical rank on the different datasets. The 2 best descriptors for each representation are in boldface.

Descriptors		Rank
Protein representation	Descriptor	
AAS	AS	23.42
	2G	27.25
	QRC[d]	21.54
	AC[d]	**11.52**
	P2G[d]	39.78
	AA[d]	**21.36**
	GE	30.24
	NG	27.85
	SAC	23.45
	DW[d]	29.48
PSSM	AB	15.25
	SAN	**7.25**
	SA	13.20
	AM	20.50
	PP	**5.02**
	SVD	39.56
	DCT	28.56
	LHF_G	24.10
	LPQ_G	14.87
	LHF_L	31.81
	LPQ_L	26.72
	BGR	12.44
	TGR	15.68
SMR	AB	28.78
	SAN	24.80
	SA	24.82
	AM	40.52
	PP	**12.50**
	SVD	29.20
	DCT	32.45
	LHF_G	**17.02**
	LPQ_G	17.22
	LHF_L	26.24
	LPQ_L	31.24
	BGR	19.86
	TGR	23.24
PR (ensemble of 25 PR[d])	SVD	**38.25**
	DCT	**37.85**
	LHF_G	41.25
	LPQ_G	38.38
	LHF_L	44.02
	LPQ_L	38.48
WAVE (ensemble of 25 WAVE[d])	SVD	40.25
	DCT	47.00
	LHF_G	**38.95**
	LPQ_G	**34.01**
	LHF_L	41.10
	LPQ_L	40.20

The second experiment is aimed at comparing only the best descriptors found in Table 3. In Tables 4 and 5, we report the performance (in terms of AUC) of the two best

Table 4: Comparison in terms of AUC in 2 class problems.

AUC		Datasets						
Protein representation	Descriptor	DNA	HU	HI	GP	AD	VI1	VI2
AAS	AC[d]	92.6	71.8	**96.4**	99.1	80.9	**90.0**	76.5
	AA[d]	90.6	68.3	—	98.8	78.9	**89.2**	75.6
PSSM	PP	**95.5**	**81.3**	94.8	**99.8**	**87.7**	86.2	**87.3**
	SAN	**95.2**	**76.4**	**95.7**	**99.7**	**82.7**	87.3	**85.7**
SMR	PP	92.9	73.8	—	99.5	79.8	88.5	76.0
	LHF_G	89.3	69.0	—	99.3	81.6	83.4	71.1
PR	SVD	79.6	74.2	—	98.0	72.3	59.1	73.3
	DCT	83.4	67.7	—	95.9	73.4	68.4	63.0
WAVE	LPQ_G	83.1	68.6	—	98.5	74.0	67.4	67.6
	LHF_G	77.7	68.6	—	97.8	68.9	65.1	60.8

Table 5: Comparison in terms of AUC in multiclass problems.

AUC		Datasets								
Protein representation	Descriptor	MEM	PF	ENZ	GR	VR	SL	CY	GO	SC
AAS	AC[d]	93.6	84.8	66.7	92.7	**81.8**	93.2	78.4	70.0	67.6
	AA[d]	90.4	84.2	63.7	92.6	72.2	91.1	76.5	69.5	65.5
PSSM	PP	**96.8**	**93.1**	**78.0**	80.8	**81.8**	**95.7**	**79.4**	**84.5**	**70.3**
	SAN	95.5	**87.7**	**71.2**	**93.0**	72.0	**94.1**	**81.8**	**78.6**	**73.9**
SMR	PP	94.2	85.9	66.2	**92.8**	76.9	92.2	78.7	69.0	66.2
	LHF_G	**96.2**	87.6	65.6	91.3	**82.4**	89.5	78.2	72.4	62.9
PR	SVD	94.4	83.5	59.4	80.8	76.0	85.4	73.5	59.7	60.3
	DCT	91.7	79.5	60.8	82.6	74.2	83.9	71.7	65.3	64.2
WAVE	LPQ_G	94.2	87.2	63.2	82.7	79.2	83.4	68.1	65.7	58.1
	LHF_G	92.7	86.2	61.5	80.3	80.6	81.0	66.6	65.2	57.0

Table 6: Comparisons with previous versions of WAVE and PR.

AUC		Dataset		
Protein representation	Descriptor	HU	GP	AD
WAVE	Best in [34]	66.1	96.6	67.1
PR	Best in [34]	62.8	87.8	57.5
WAVE	LPQ_G	68.6	**98.5**	72.3
PR	SVD	**74.2**	98.0	**74.0**

descriptors for each representation (see Table 3) related to 2-class and multiclass datasets, respectively. The two best results for each dataset are highlighted. On average PP coupled with PSSM obtains the best results in most datasets, but in some problems the PP descriptor is outperformed by SAN (which is always coupled to PSSM). Some results related to HI are not reported due to the high computational costs.

The best results in the previous tables are almost always obtained with PSSM and AAS representations of proteins. Comparing the reported results of PR and WAVE with [34], where the first versions of those representations were tested, with the experiments reported here, it is clear that there is a boost in performance in PR and WAVE. See, for comparison, Table 6, where previous results obtained using these representations (SVM is the classifier) are reported.

The third experiment tests some ensemble approaches based on the fusion of some of the best descriptors, selected considering all the datasets, excluding HI. The ensembles tested in this experiment are obtained as the weighed fusion of the following methods, labelled in terms of representation (descriptor):

(i) FUS1: 2 × AAS(AC) + 2 × PSSM(SAN) + 4 × PSSM(PP) + PSSM(LHF_G) + PSSM(BGR) + PSSM(TGR) + SMR(PP) + SMR(BGR),

(ii) FUS2: 2 × AAS(AC) + 2 × PSSM(SAN) + 4 × PSSM(PP) + PSSM(LHF_G) + PSSM(BGR) + PSSM(TGR) + SMR(PP) + SMR(BGR) + 2 × DM(LPQ_G) = FUS1 + 2 × DM(LPQ_G).

The results of these two ensembles are compared in Tables 7 and 8 with the best three single methods. Results related to FUS2 are reported for only a few datasets since it contains a descriptor based on the DM matrix.

The most interesting result among those reported in Tables 7 and 8 is that of our ensemble FUS1, which outperforms the other approaches in nearly all the datasets and accomplishes this performance gain without changing its weights. It is also interesting to note that even though the recent representation of SMR works rather poorly compared with PSSM and AAS, it is nonetheless useful when combined with PSSM and AAS. The other representations, WAVE, PR,

Table 7: Comparison among ensembles and best stand-alone descriptors in terms of AUC in 2 class problems.

AUC	Datasets						
Protein representation	DNA	HU	HI	GP	AD	VI1	VI2
PSSM(PP)	95.5	81.2	94.8	99.8	87.7	86.2	87.2
PSSM(SAN)	95.2	76.4	95.7	99.7	82.7	87.3	85.7
AAS(AC)	92.6	71.8	95.9	99.1	80.9	90.0	76.4
FUS1	97.2	**82.0**	**98.4**	**99.9**	**88.2**	**89.0**	**88.7**
FUS2	**97.3**	—	—	—	—	—	—

Table 8: Comparison among ensembles and best stand-alone descriptors in terms of AUC in multiclass problems.

AUC	Datasets								
Protein representation	MEM	PF	ENZ	GR	VR	SL	CY	GO	SC
PSSM(PP)	96.8	93.1	78.0	80.8	81.8	95.7	79.4	**84.5**	70.3
PSSM(SAN)	95.5	87.7	71.1	**93.0**	72.0	94.1	81.8	78.6	73.4
AAS(AC)	93.6	84.8	66.7	92.7	81.8	93.2	78.4	70.0	67.6
FUS1	**97.1**	92.7	**80.2**	92.3	**84.7**	**96.7**	**84.5**	83.8	75.3
FUS2	—	**95.9**	80.1	—	—	—	84.3	82.8	**76.4**

and DM, are not yet useful in fusion; in our opinion, a wide survey on different texture descriptors still needs to be performed to determine which set of descriptors can boost the performance of these representations.

The forth experiment is aimed at comparing our ensembles FUS1 and FUS2 with the performance reported in the literature by other state-of-the-art approaches. Unfortunately, a fair comparison with other approaches is not always easy for the following reasons.

(i) Several papers use self-collected datasets and only in a few cases is the code for feature extraction available.

(ii) Many works report results obtained on small datasets, without a clear indication of the testing protocol used; therefore, it is difficult to know whether parameter optimization was performed on the entire dataset (thereby overfitting results) or only on a training set. Overfitting is particularly dangerous in small datasets.

The comparison is much easier when considering large datasets (as with HI and MEM) or when an independent dataset separate from the training set is available (as in PF). So in the following tests we compare our results only when we are quite sure that the comparison is fair.

Tables 9 and 10 report the performance in terms of AUC and accuracy, respectively. When available we have used original source code for comparing methods. When results are extracted from the original reference paper, the best method reported in the paper is considered in the comparison. It should also be noted that in what follows we are comparing the most widely used descriptors in the literature, using whenever possible the original source code for the descriptors (not our reimplementation). To ensure fair comparison, we have also used the same testing protocol that was used in the original reference. Moreover, it should be noted that although it is the case that when small datasets are used (or when only a few datasets are tested) a jackknife approach is quite feasible, in our survey using several datasets and many descriptors, the jackknife approach becomes computationally unfeasible.

The results reported in Tables 9 and 10 are interesting not only because in this work we outperform all our previously proposed ensembles but also because we obtain state-of-the-art performances on such large datasets as HI and on such widely used benchmarks as PF and MEM. Please note that our ensemble FUS1 works well across nearly all the tested datasets, without any parameter tuning to optimize performance for a given dataset.

Considering the dataset PF, which is one of the most widely used benchmarks, FUS1 compares very well with the other approaches where features are not extracted using 3D information (for a fair comparison). The performance FUS1 is all the more valuable when considering that unlike the older approaches, ours is obtained without ad-hoc feature extractors (where the features are validated only on PF with a high risk of overfitting).

The compared approaches on PF are the following.

(i) Reference [15], where six kinds of features denoted by C, S, H, P, V, and Z are proposed. C is the popular amino-acid composition; the remaining five indicate the features of polarity, polarizability, normalized Van Der Waals volume, hydrophobicity, and predicted secondary structure, respectively.

(ii) Reference [70], where the same CSHPVZ features proposed by [15] are used, but with different classifier systems.

(iii) Reference [69], where the authors combine CSHPVZ features with bigram-coded feature (B) and spaced bigram-coded feature (SB).

(iv) Reference [72], where the authors do the same work as [69] but improve the classifier system using the technique of data fusion.

Table 9: Comparison with the state-of-the-art using AUC as performance indicator.

AUC	Datasets													
Methods	HU	PF	GP	GR	VR	DNA	ENZ	MEM	GO	SL	HI	AD	VI1	VI2
[48]												77.0	87.0	83.4
[58]						93.3	72.5		50.0					
[12]	72.5		99.7	**94.7**	82.5			96.0				82.9	86.1	76.0
[59]											98.2			
[34]												81.6	91.2	84.1
[28]						95.9	79.4	96.8		93.8	98.0		87.1	87.9
FUS1	**82.0**	92.7	**99.9**	92.3	**84.7**	**97.2**	**80.2**	**97.1**	**83.8**	**96.7**	**98.4**	**88.2**	**89.0**	**88.7**

Table 10: Comparison with the state-of-the-art using accuracy as performance indicator.

Accuracy	Datasets													
Methods	HU	PF	GP	GR	VR	DNA	ENZ	MEM	GO	SL	HI	AD	VI1	VI2
[15]		56.50												
[69]		65.50												
[70]		58.18												
[40]		61.04												
[71]	70.0													
[72]		69.60												
[3–5]								91.6						
[20]			91.6											
[61]				84.1										
[1]								92.7						
[73]								92.6						
[12]	70.0		98.1	84.4	78.6			91.5						
[28]							56.2	94.1	59.4	85.8	93.1		85.5	81.7
FUS1	**75.0**	68.6	**99.2**	87.9	**76.2**	**93.7**	**56.9**	94.3	64.3	**87.0**	93.9	**78.0**	**84.3**	**83.5**
FUS2		74.6				**94.6**	**57.1**		63.0					

Since the PF dataset aims at predicting the 3D structure of a protein, features extracted from 3D representations are highly useful as proven by the better performance obtained by FUS2 with respect to FUS1.

Given the results reported above, our proposed ensemble FUS1 should prove useful for practitioners and experts alike since it can form the base for building systems that are optimized for particular problems (e.g., SVM optimization and physicochemical properties selection). Obviously, it is very important that only the training data be used for physicochemical properties selection; it is not fair to choose the physicochemical properties using the entire dataset to do so. Moreover, when the ensemble is optimized for a given dataset, it is very important to consider that large descriptors work better when a large training set is available (because of the curse of dimensionality). As an example of this, we report below the performance of AAS(RC) and AAS(AC). AAS(RC) has high dimensionality and, accordingly, as seen in Table 11, works well mainly on large datasets. For this reason, it can be used in an ensemble only in the case where a large training set is available (as with MEM or HI). Notice that in HI the method AAS(RC) outperforms our best ensemble.

A similar behavior occurs with some other methods. In Table 12 we report the performance obtained by PSSM(LPQ_G). It works very poorly in some datasets but very well in others (mainly with the larger datasets).

It is clear from our experimental results that it is difficult to find an ensemble that performs the best across each of the datasets. Nonetheless, we have shown that among the several tested and proposed protein descriptors, it is always possible to find an ensemble that performs well in each type of dataset.

6. Conclusion

One goal in this work was to provide a survey of several state-of-the-art descriptors and some new variants starting from different protein representations. We compare the performance of these descriptors across several benchmark datasets. The results reported in this paper show that the best protein representation is PSSM, but AAS and SMR also work well. We found that no single descriptor is superior to all others across all tested datasets.

Another objective of this study was to search for a general ensemble method that could work well on different protein classification datasets. Accordingly, we performed several fusions for finding experimentally a set of descriptors based on different representations that worked well across each of the tested datasets. A couple of representations, such as

Table 11: Comparison between AAS(RC) and AAS(AC).

AUC	Datasets															
Methods	HU	PF	GP	GR	VR	DNA	ENZ	MEM	GO	SL	HI	AD	VI1	VI2	CY	SC
AAS(RC)	70.3	**87.2**	98.9	90.0	69.0	86.2	64.5	**95.9**	68.3	87.8	**98.9**	**81.1**	89.2	75.9	77.6	62.4
AAS(AC)	**71.8**	84.8	**99.1**	**92.7**	**81.8**	**92.6**	**66.7**	93.6	**70.0**	**93.2**	95.9	80.9	**90.0**	**76.4**	**78.4**	**67.6**

Table 12: Comparison among ensembles and best stand-alone descriptors in terms of AUC.

AUC	Datasets															
Methods	HU	PF	GP	GR	VR	DNA	ENZ	MEM	GO	SL	HI	AD	VI1	VI2	CY	SC
PSSM(PP)	**81.2**	**93.1**	99.8	80.8	**81.8**	**95.5**	**78.0**	**96.8**	**84.5**	**95.7**	94.8	**87.7**	86.2	**87.2**	79.4	70.3
PSSM(SAN)	76.4	87.7	99.7	**93.0**	72.0	95.2	71.1	95.5	78.6	94.1	95.7	82.7	**87.3**	85.7	**81.8**	**73.4**
PSSM(LPQ_G)	72.0	89.5	**99.9**	82.3	77.7	89.5	66.2	93.6	73.0	93.7	**97.6**	86.8	82.3	83.9	70.3	61.6

WAVE and PR, were not useful in fusion. Given the results of our experiments, we concluded that a wide survey of different texture descriptors needs to be performed since different descriptors contain different information that might boost performance when combined with others.

Our major contribution is to propose an ensemble of descriptors/classifiers for sequence-based protein classification that not only works well across several datasets but also, in some cases, proves superior to the state-of-the-art. Unlike other papers that develop a web server, we share almost all the MATLAB codes used in the proposed approaches. Our proposed ensemble could be considered a baseline system for developing an ad-hoc system for a given problem. Issues to consider when optimizing such a base system for a given dataset were also discussed. For instance, the size of datasets seems to play a role in the choice of protein representation, with some descriptors showing stronger performance on large datasets. In particular, approaches that use a high dimensional representation (e.g., RC) requires larger datasets in order to avoid the curse of dimensionality.

To further improve the performance of our methods, we plan, in the future, on testing more classification approaches. We are particularly interested in investigating ensembles made with AdaBoost and Rotation forest [74] classifiers. The main drawback using these ensemble methods is that they require more computational power than SVM, the classifier used in this work. Although this would not be a problem for the testing phase, it would be a drawback in the training phase if we want to compare a number of different descriptors across several (preferably large) datasets.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] J. Wang, Y. Li, Q. Wang et al., "ProClusEnsem: predicting membrane protein types by fusing different models of pseudo amino acid composition," *Computers in Biology and Medicine*, vol. 42, no. 5, pp. 564–574, 2012.

[2] K.-C. Chou, "Some remarks on protein attribute prediction and pseudo amino acid composition," *Journal of Theoretical Biology*, vol. 273, no. 1, pp. 236–247, 2011.

[3] K.-C. Chou and H.-B. Shen, "MemType-2L: a Web server for predicting membrane proteins and their types by incorporating evolution information through Pse-PSSM," *Biochemical and Biophysical Research Communications*, vol. 360, no. 2, pp. 339–345, 2007.

[4] K.-C. Chou and H.-B. Shen, "Recent progress in protein subcellular location prediction," *Analytical Biochemistry*, vol. 370, no. 1, pp. 1–16, 2007.

[5] K.-C. Chou and H.-B. Shen, "Signal-CF: a subsite-coupled and window-fusing approach for predicting signal peptides," *Biochemical and Biophysical Research Communications*, vol. 357, no. 3, pp. 633–640, 2007.

[6] K.-C. Chou and H.-B. Shen, "A new method for predicting the subcellular localization of eukaryotic proteins with both single and multiple sites: Euk-mPLoc 2.0," *PLoS ONE*, vol. 5, no. 4, Article ID e9931, 2010.

[7] M. Maddouri and M. Elloumi, "Encoding of primary structures of biological macromolecules within a data mining perspective," *Journal of Computer Science and Technology*, vol. 19, no. 1, pp. 78–88, 2004.

[8] R. Saidi, M. Maddouri, and E. Mephu Nguifo, "Protein sequences classification by means of feature extraction with substitution matrices," *BMC Bioinformatics*, vol. 11, article 175, 2010.

[9] H. Nakashima, K. Nishikawa, and T. Ooi, "The folding type of a protein is relevant to the amino acid composition," *Journal of Biochemistry*, vol. 99, no. 1, pp. 153–162, 1986.

[10] K.-C. Chou, "Prediction of protein cellular attributes using pseudo-amino acid composition," *Proteins: Structure, Function and Genetics*, vol. 43, no. 3, pp. 246–255, 2001.

[11] K.-C. Chou, "Pseudo amino acid composition and its applications in bioinformatics, proteomics and system biology," *Current Proteomics*, vol. 6, no. 4, pp. 262–274, 2009.

[12] L. Nanni, S. Brahnam, and A. Lumini, "High performance set of PseAAC and sequence based descriptors for protein classification," *Journal of Theoretical Biology*, vol. 266, no. 1, pp. 1–10, 2010.

[13] L. Nanni and A. Lumini, "An ensemble of K-local hyperplanes for predicting protein-protein interactions," *Bioinformatics*, vol. 22, no. 10, pp. 1207–1210, 2006.

[14] H. Lin and H. Ding, "Predicting ion channels and their types by the dipeptide mode of pseudo amino acid composition," *Journal of Theoretical Biology*, vol. 269, no. 1, pp. 64–69, 2011.

[15] C. H. Q. Ding and I. Dubchak, "Multi-class protein fold recognition using support vector machines and neural networks," *Bioinformatics*, vol. 17, no. 4, pp. 349–358, 2001.

[16] H. Lin, W. Chen, L. Yuan, Z. Li, and H. Ding, "Using over-represented tetrapeptides to predict protein submitochondrial locations," *Acta Biotheoretica*, vol. 61, no. 2, pp. 259–268, 2013.

[17] W.-Z. Lin, X. Xiao, and K.-C. Chou, "GPCR-GIA: a web-server for identifying G-protein coupled receptors and their families with grey incidence analysis," *Protein Engineering, Design and Selection*, vol. 22, no. 11, pp. 699–705, 2009.

[18] X. Xiao, W.-Z. Lin, and K.-C. Chou, "Using grey dynamic modeling and pseudo amino acid composition to predict protein structural classes," *Journal of Computational Chemistry*, vol. 29, no. 12, pp. 2018–2024, 2008.

[19] X. Xiao, S. Shao, Y. Ding, Z. Huang, and K.-C. Chou, "Using cellular automata images and pseudo amino acid composition to predict protein subcellular location," *Amino Acids*, vol. 30, no. 1, pp. 49–54, 2006.

[20] X. Xiao, P. Wang, and K.-C. Chou, "GPCR-CA: a cellular automaton image approach for predicting G-protein-coupled receptor functional classes," *Journal of Computational Chemistry*, vol. 30, no. 9, pp. 1414–1423, 2009.

[21] X. Xiao, S. Shao, Y. Ding, Z. Huang, Y. Huang, and K.-C. Chou, "Using complexity measure factor to predict protein subcellular location," *Amino Acids*, vol. 28, no. 1, pp. 57–61, 2005.

[22] T. Jaakkola, M. Diekhans, and D. Haussler, "Using the Fisher kernel method to detect remote protein homologies," *Proceedings of the 7th International Conference on Intelligent Systems for Molecular Biology (ISMB '99)*, AAAI Press, pp. 149–158, 1999.

[23] C. Leslie, E. Eskin, and W. S. Noble, "The spectrum kernel: a string kernel for SVM protein classification," *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, pp. 564–575, 2002.

[24] C. S. Leslie, E. Eskin, A. Cohen, J. Weston, and W. S. Noble, "Mismatch string kernels for discriminative protein classification," *Bioinformatics*, vol. 20, no. 4, pp. 467–476, 2004.

[25] Z. Lei and Y. Dai, "An SVM-based system for predicting protein subnuclear localizations," *BMC Bioinformatics*, vol. 6, article 291, 2005.

[26] Z. R. Yang and R. Thomson, "Bio-basis function neural network for prediction of protease cleavage sites in proteins," *IEEE Transactions on Neural Networks*, vol. 16, no. 1, pp. 263–274, 2005.

[27] M. Gribskov, A. D. McLachlan, and D. Eisenberg, "Profile analysis: detection of distantly related proteins," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 84, no. 13, pp. 4355–4358, 1987.

[28] L. Nanni, A. Lumini, and S. Brahnam, "An empirical study on the matrix-based protein representations and their combination with sequence-based approaches," *Amino Acids*, vol. 44, no. 3, pp. 887–901, 2013.

[29] S. R. Hegde, K. Pal, and S. C. Mande, "Differential enrichment of regulatory motifs in the composite network of protein-protein and gene regulatory interactions," *BMC Systems Biology*, vol. 8, p. 26, 2014.

[30] T. Huang, P. Wang, Z. Ye et al., "Prediction of deleterious non-synonymous SNPs based on protein interaction network and hybrid properties," *PLoS ONE*, vol. 5, no. 7, Article ID e11900, 2010.

[31] L. J. Jensen, M. Kuhn, M. Stark et al., "STRING 8—a global view on proteins and their functional interactions in 630 organisms," *Nucleic Acids Research*, vol. 37, no. 1, pp. D412–D416, 2009.

[32] A. Mora and I. M. Donaldson, "Effects of protein interaction data integration, representation and reliability on the use of network properties for drug target prediction," *BMC Bioinformatics*, vol. 13, p. 294, 2012.

[33] R. Schweiger, M. Linial, and N. Linial, "Generative probabilistic models for protein-protein interaction networks-the biclique perspective," *Bioinformatics*, vol. 27, no. 13, pp. i142–i148, 2011.

[34] L. Nanni, S. Brahnam, and A. Lumini, "Wavelet images and Chou's pseudo amino acid composition for protein classification," *Amino Acids*, vol. 43, no. 2, pp. 657–665, 2012.

[35] S. Kawashima and M. Kanehisa, "AAindex: amino acid index database," *Nucleic Acids Research*, vol. 28, no. 1, p. 374, 2000.

[36] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*, Cambridge University Press, Cambridge, UK, 2000.

[37] X. Yu, X. Zheng, T. Liu, Y. Dou, and J. Wang, "Predicting subcellular location of apoptosis proteins with pseudo amino acid composition: approach from amino acid substitution matrix and auto covariance transformation," *Amino Acids*, vol. 42, no. 5, pp. 1619–1625, 2012.

[38] F.-M. Li and Q.-Z. Li, "Predicting protein subcellular location using Chou's pseudo amino acid composition and improved hybrid approach," *Protein and Peptide Letters*, vol. 15, no. 6, pp. 612–616, 2008.

[39] S. Brahnam, L. Nanni, J.-Y. Shi, and A. Lumini, "Local phase quantization texture descriptor for protein classification," in *International Conference on Bioinformatics and Computational Biology (BIOCOMP '10)*, pp. 159–165, Las Vegas, Nev, USA, 2010.

[40] J.-Y. Shi and Y.-N. Zhang, "Using texture descriptor and radon transform to characterize protein structure and build fast fold recognition," in *International Association of Computer Science and Information Technology (IACSIT-SC '09)*, pp. 466–470, April 2009.

[41] J. Guo, Y. Lin, and Z. Sun, "A novel method for protein subcellular localization: combining residue-couple model and SVM," in *Proceedings of 3rd Asia-Pacific Bioinformatics Conference*, pp. 117–129, 2005.

[42] Y.-H. Zeng, Y.-Z. Guo, R.-Q. Xiao, L. Yang, L.-Z. Yu, and M.-L. Li, "Using the augmented Chou's pseudo amino acid composition for predicting protein submitochondria locations based on auto covariance approach," *Journal of Theoretical Biology*, vol. 259, no. 2, pp. 366–372, 2009.

[43] E. Tantoso and K.-B. Li, "AAIndexLoc: predicting subcellular localization of proteins based on a new representation of sequences using amino acid indices," *Amino Acids*, vol. 35, no. 2, pp. 345–353, 2008.

[44] X. Li, B. Liao, Y. Shu, Q. Zeng, and J. Luo, "Protein functional class prediction using global encoding of amino acid sequence," *Journal of Theoretical Biology*, vol. 261, no. 2, pp. 290–293, 2009.

[45] L. R. Murphy, A. Wallqvist, and R. M. Levy, "Simplified amino acid alphabets for protein fold recognition and implications for folding," *Protein Engineering*, vol. 13, no. 3, pp. 149–152, 2000.

[46] M. Kumar, R. Verma, and G. P. S. Raghava, "Prediction of mitochondrial proteins using support vector machine and hidden Markov model," *Journal of Biological Chemistry*, vol. 281, no. 9, pp. 5357–5363, 2006.

[47] J. C. Jeong, X. Lin, and X.-W. Chen, "On position-specific scoring matrix for protein function prediction," *IEEE/ACM*

Transactions on Computational Biology and Bioinformatics, vol. 8, no. 2, pp. 308–315, 2011.

[48] A. Garg and D. Gupta, "VirulentPred: a SVM based prediction method for virulent proteins in bacterial pathogens," *BMC Bioinformatics*, vol. 9, article 62, 2008.

[49] L. Yang, Y. Li, R. Xiao et al., "Using auto covariance method for functional discrimination of membrane proteins based on evolution information," *Amino Acids*, vol. 38, no. 5, pp. 1497–1503, 2010.

[50] G.-L. Fan and Q.-Z. Li, "Predicting protein submitochondrion locations by combining different descriptors into the general form of Chou's pseudo amino acid composition," *Amino Acids*, vol. 20, pp. 1–11, 2011.

[51] G. H. Golub and C. Reinsch, "Singular value decomposition and least squares solutions," *Numerische Mathematik*, vol. 14, no. 5, pp. 403–420, 1970.

[52] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Transactions on Computers*, vol. 23, no. 1, pp. 90–93, 1974.

[53] A. Sharma, J. Lyons, A. Dehzangi, and K. K. Paliwal, "A feature extraction technique using bi-gram probabilities of position specific scoring matrix for protein fold recognition," *Journal of Theoretical Biology*, vol. 320, pp. 41–46, 2013.

[54] T. Ahonen, J. Matas, C. He, and M. Pietikäinen, "Rotation invariant image description with local binary pattern histogram fourier features," in *Image Analysis*, vol. 5575 of *Lecture Notes in Computer Science*, pp. 61–70, Springer, 2009.

[55] V. Ojansivu and J. Heikkilä, "Blur insensitive texture classification using local phase quantization," *Image and Signal Processing*, vol. 5099, pp. 236–243, 2008.

[56] K.-C. Chou and H.-B. Shen, "Recent progress in protein subcellular location prediction," *Analytical Biochemistry*, vol. 370, no. 1, pp. 1–16, 2007.

[57] Y. Fang, Y. Guo, Y. Feng, and M. Li, "Predicting DNA-binding proteins: approached from Chou's pseudo amino acid composition and other specific sequence features," *Amino Acids*, vol. 34, no. 1, pp. 103–109, 2008.

[58] L. Nanni, S. Mazzara, L. Pattini, and A. Lumini, "Protein classification combining surface analysis and primary structure," *Protein Engineering, Design and Selection*, vol. 22, no. 4, pp. 267–272, 2009.

[59] X.-Y. Pan, Y.-N. Zhang, and H.-B. Shen, "Large-scale prediction of human protein-protein interactions from amino acid sequence based on latent topic features," *Journal of Proteome Research*, vol. 9, no. 10, pp. 4992–5001, 2010.

[60] P. Du and Y. Li, "Prediction of protein submitochondria locations by hybridizing pseudo-amino acid composition with various physicochemical features of segmented sequence," *BMC Bioinformatics*, vol. 7, article 518, 2006.

[61] H.-B. Shen and K.-C. Chou, "Gpos-PLoc: an ensemble classifier for predicting subcellular localization of Gram-positive bacterial proteins," *Protein Engineering, Design and Selection*, vol. 20, no. 1, pp. 39–46, 2007.

[62] J. R. Bock and D. A. Gough, "Whole-proteome interaction mining," *Bioinformatics*, vol. 19, no. 1, pp. 125–135, 2003.

[63] H.-B. Shen and K.-C. Chou, "Virus-PLoc: a fusion classifier for predicting the subcellular localization of viral proteins within host and virus-infected cells," *Biopolymers*, vol. 85, no. 3, pp. 233–240, 2007.

[64] X. Guang, Y. Guo, J. Xiao et al., "Predicting the state of cysteines based on sequence information," *Journal of Theoretical Biology*, vol. 267, no. 3, pp. 312–318, 2010.

[65] T. Fawcett, *ROC Graphs: Notes and Practical Considerations for Researchers*, HP Laboratories, Palo Alto, Calif, USA, 2004.

[66] T. C. W. Landgrebe and R. P. W. Duin, "Approximating the multiclass ROC by pairwise analysis," *Pattern Recognition Letters*, vol. 28, no. 13, pp. 1747–1758, 2007.

[67] Z.-C. Qin, "ROC analysis for predictions made by probabilistic classifiers," in *International Conference on Machine Learning and Cybernetics (ICMLC '05)*, pp. 3119–3124, August 2005.

[68] A. Ulaş, O. T. Yldz, and E. Alpaydn, "Cost-conscious comparison of supervised learning algorithms over multiple data sets," *Pattern Recognition*, vol. 45, no. 4, pp. 1772–1781, 2012.

[69] C.-D. Huang, C.-T. Lin, and N. R. Pal, "Hierarchical learning architecture with automatic feature selection for multiclass protein fold classification," *IEEE Transactions on Nanobioscience*, vol. 2, no. 4, pp. 221–232, 2003.

[70] A. Chinnasamy, W.-K. Sung, and A. Mittal, "Protein structure and fold prediction using Tree-Augmented naïve Bayesian classifier," *Journal of Bioinformatics and Computational Biology*, vol. 3, no. 4, pp. 803–819, 2005.

[71] S. Martin, D. Roe, and J.-L. Faulon, "Predicting protein-protein interactions using signature products," *Bioinformatics*, vol. 21, no. 2, pp. 218–226, 2005.

[72] K.-L. Lin, C.-Y. Lin, C.-D. Huang et al., "Feature selection and combination criteria for improving accuracy in protein structure prediction," *IEEE Transactions on Nanobioscience*, vol. 6, no. 2, pp. 186–196, 2007.

[73] Y.-K. Chen and K.-B. Li, "Predicting membrane protein types by incorporating protein topology, domains, signal peptides, and physicochemical properties into the general form of Chou's pseudo amino acid composition," *Journal of Theoretical Biology*, vol. 318, pp. 1–12, 2013.

[74] J. J. Rodríguez, L. I. Kuncheva, and C. J. Alonso, "Rotation forest: a new classifier ensemble method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, pp. 1619–1630, 2006.