Protein Folds Prediction with Hierarchical Structured SVM

Dapeng Li[a], Ying Ju[b] and Quan Zou[c,d,*]

[a]Department of Internal Medicine-Oncology, The Fourth Hospital in Qinhuangdao, Qinhuangdao, Hebei, P.R. China; [b]School of Information Science and Technology, Xiamen University, Xiamen, P.R. China; [c]School of Computer Science and Technology, Tianjin University, Tianjin, P.R. China; [d]School of Computer Engineering, Nanyang Technological University, Singapore

Dapeng Li

Abstract: Protein folds prediction is an essential and basic problem for protein structure and function research. As far as we see, there are generally three problems for the protein folds prediction. The first one is the overfitting problem due to the lack of training samples. The second one is the missing information of hierarchical labels. Small size of the current benchmark is another troubling issue. In this paper, we proposed structured SVM to overcome the first and second problems. We also contributed three comparatively huge datasets as benchmark for protein folds prediction. Experiments on different datasets can prove the performance and robustness of our structured SVM.

Keywords: Structured support vector machine, protein folds prediction, protein structure, machine learning, bioinformatics, protein secondary structure.

1. INTRODUCTION

Sequencing technology has improved vastly, and its cost has been substantially decreased in the recent years, which has yielded an incredible increase in the acquisition of genomic and transcriptomic data. It is impossible to annotate all the coding regions and related protein functions using 'wet-lab,' molecular biology experiments. Therefore, computer aided prediction is used, which has become the major annotation approach. Protein function is primarily related to protein structure. Therefore, it is of vital importance to predict structural and functional information from protein primary sequences in silico.

Proteins can be categorized hierarchically into four levels; databases such as the Structural Classification Of Proteins (SCOP) [1, 2] can do so. From macro to micro dimensions, these categories in turn are named as class, fold, superfamily, and family. SCOP (version 1.75) contains 7 classes, 1,194 folds, 1,961 superfamilies and 4,493 families, as shown in (Fig. **1**). Proteins from the same family are clearly evolutionarily and generally share 30% or greater pairwise sequence identity, which can be discovered by sequence alignment software, such as PSI-BLAST [3]. Proteins from the same superfamily have a high probability to originate from the same ancestor, and as such, are likely homologous, and always share the same or a similar structure and function. Predicting the superfamily of a novel protein is important for understanding the function of the protein. PSI-BLAST is helpful for finding a protein's family category, however, often fails regarding its superfamily.

Predicting whether two proteins belong to the same superfamily is an aspect of protein remote homology detection, and is considered a core and 'twilight zone' problem in bioinformatics [4, 5].

Many in silico remote homology detection approaches have been proposed to address this problem in the recent years, of which the discriminative methods are considered to be the most accurate. Multiple classes and hierarchical classifiers are employed to predict superfamily information; however, they cannot achieve perfect performance due to a large number of superfamily counts. [6-8]. Due to the difficulty of large multiple class learning, Liao *et al.* suggested to reduce the complexity of the problem by determining whether a protein has remote homology to another particular family instead of superfamily [9]. Therefore, the multiple class learning problem can be reduced to a binary imbalance classification, and is more suitable for comparing the classification power of the proposed features. Moreover, Liao *et al.* also supplied a benchmark dataset for testing the average performance on 54 families' remote homology recognition. Almost all the superfamily prediction methods [10, 11] are trained from Liao's dataset, which means that only 54 superfamilies can be predicted for a novel protein sequence. Therefore, it contributes few for novel protein function analyses.

Therefore, the prediction of folds and classes is of vital importance for protein function research. There are only 7 classes for all the proteins. (In the former research, only 4 classes were considered [12].) As a computational problem, it is proper to do 7-classes-classification, and some works have obtained good performance [13, 14]. However, from biological point view, it is rather few for function research if there are only 7 classes. Thousands of GO (Gene Ontology) repositories have been employed for annotating protein func-

*Address correspondence to this author at the School of Computer Science and Technology, Tianjin University, Tianjin, 300350, P.R. China; Tel/Fax: +86-17092261008; E-mails: zouquan@nclab.net

1570-1646/16 $58.00+.00

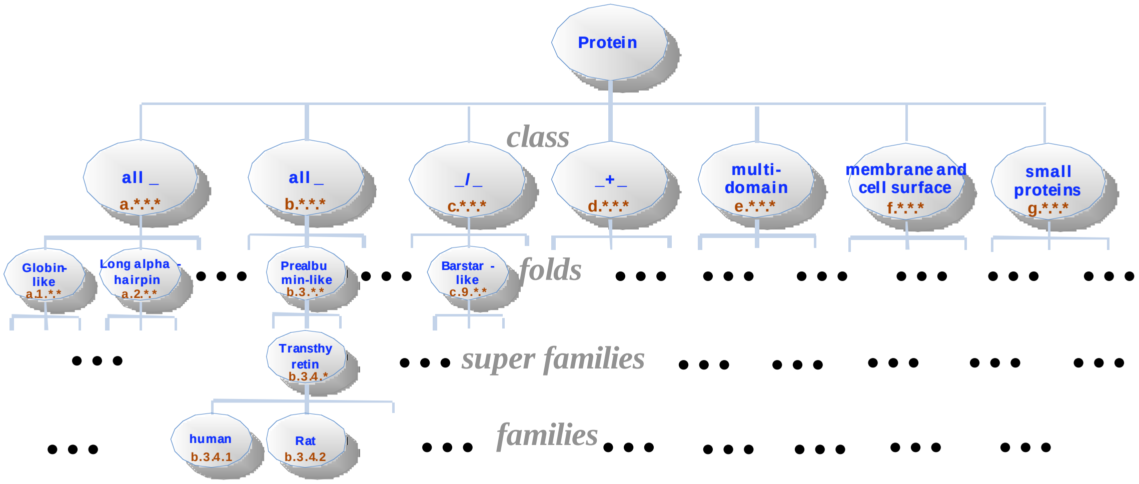

Fig. (1). Hierarchical structure for proteins.

tions. Therefore, biologists pay more attention to the folds prediction of protein. However, thousands-classes-classification is an unsolvable problem in the machine learning fields. Few bioinformatics works have attempted been on the protein folds prediction so far. Ding and Dubchak [15] firstly did the prediction work for protein folds. In order to avoid the ultra-large multiple classification problem, they deleted the folds having only a few samples, and reconstructed a benchmark dataset (often called DD dataset) for protein folds prediction. In their dataset, 311 protein sequences with ≤ 40% sequence similarity were included in the training set, and 383 protein sequences with ≤ 35% sequence similarity were set in the testing set. There are 27-fold classes altogether, which are far fewer than the real folds in world. Nearly, all of the subsequent works are employed in this dataset for protein folds prediction [16-18]. Later, several works are involved in more protein folds for training and predicting [19]. The datasets were found to be extended to 95 classes and 194 classes [20].

However, there are still two main problems for the protein folds prediction. First, the training samples are not enough and therefore, cannot represent all the sample space. So overfitting often occurs especially when ensembling classifier [21, 22] or optimized parameters are employed. Second, the hierarchical structure of the folds is employed. For example, “globin-like” and “long α-hairpin” belong to the “all α proteins” class, while “prealbumin-like” and “L21p-like” belong to the “all β proteins” class. The protein classes information may help the folds prediction.

In this paper, we employed hierarchical structured SVM (Support Vector Machine) as the multiple-classes-classifier, which can involve the different distances between different folds. SVM is considered as the best solution to the overfitting problem. Furthermore, we deleted the folds containing very few protein sequences, and extended the popular DD dataset. New datasets can help to check the robustness of the proposed methods.

2. FEATURE SELECTION AND FUSION

Feature selection and fusion is the most important and the key techniques for protein classification [22]. The features are extracted for the identification of special proteins, such as enzyme [23], cytokine [24], DNA binding protein [25-27], *etc*. The features can also be used for protein-protein interaction [28] and sites [29] prediction.

Several features have been proposed for describing protein structural information. The most naive features are n-grams, which come from computational linguistics [9]. These approaches are also applied to extract the features of DNA or RNA sequences [30, 31]. Alignment mismatch and n-gram scores are always employed as a kernel for representing the distance between protein sequences, rather than being extracted as features [32-34]. Therefore, profile related features have been employed to improve performance and involve more evolutionary information. Training protein sequences are aligned to a known protein database, and evolutionary information, including feature variation, is extracted from the profiles, which is the alignment result. Profile features achieve good performance; however, running time is quite slow because the alignment step is very time consuming [35-37]. Subsequently, amino acid physicochemical characters have been used as features rather than individual amino acids, to achieve time acceleration due to a reduced alphabet. These include hydrophobicity, normalized van der Waals volume, polarity, polarizability, charge, and surface tension [38, 39].

Here, we extracted several popular features following the former work empirically. Then the features are combined and ranked with a novel metric. Finally, the features contributed dramatically for the classification are selected as the final ones. There are 473 features altogether, sequence features and secondary structure features. First, the PSSM was computed and the 20 amino acid average values in the PSSM were computed for the first 20D features. Then the protein sequence is transformed into the evolutionary profile, and every amino acid is replaced by the new amino acid which appears to be the most often in the evolutionary process. 1-gram and 2-gram of the transformed sequence are computed. So, this is the next 20D and 400D features. 6D features of the global secondary structure are the next part. LX3 matrix, which is similar to PSSM, is computed for the last 27D local secondary structure features.

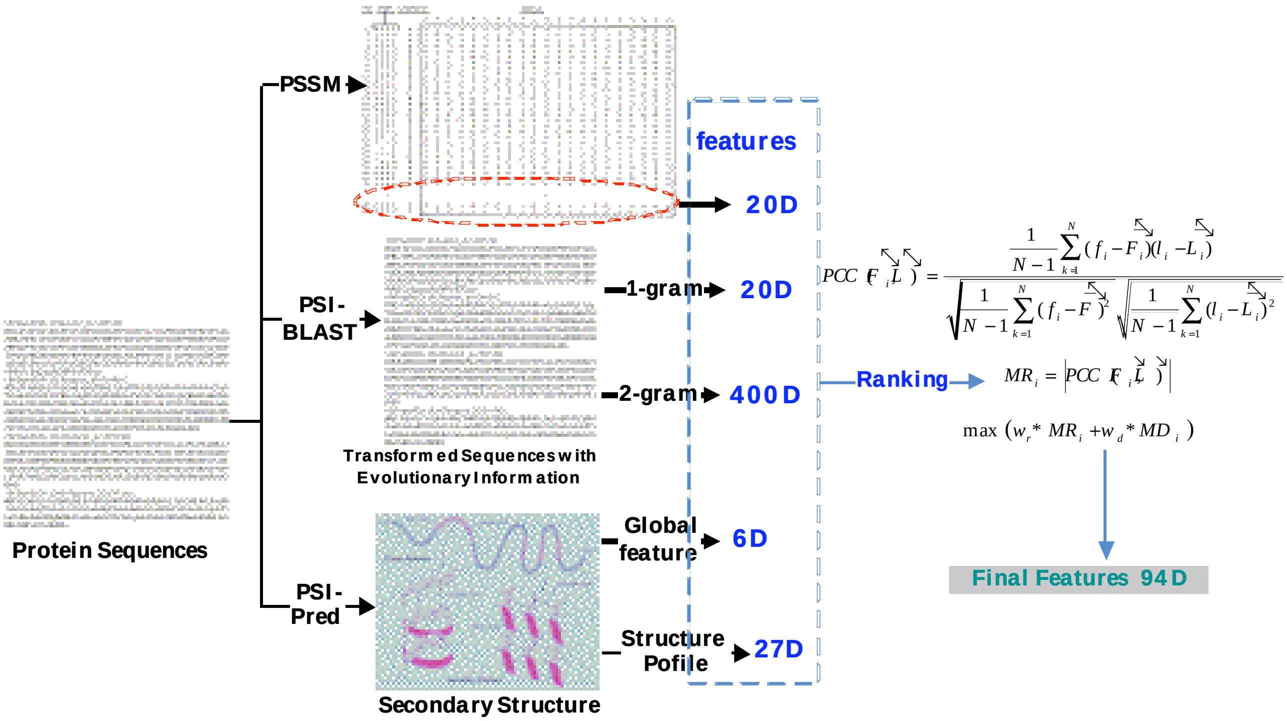
PSSM
features
20D
PSI-BLAST
1-gram
20D
2-gram
400D
Transformed Sequences with Evolutionary Information
Protein Sequences
PSI-Pred
Global feature
6D
Structure Pofile
27D
Secondary Structure
$PCC(\overleftrightarrow{F_i},\overleftrightarrow{L}) = \frac{\frac{1}{N-1}\sum_{k=1}^{N}(f_i-\overleftrightarrow{F_i})(l_i-\overleftrightarrow{L_i})}{\sqrt{\frac{1}{N-1}\sum_{k=1}^{N}(f_i-\overleftrightarrow{F})^2}\sqrt{\frac{1}{N-1}\sum_{k=1}^{N}(l_i-\overleftrightarrow{L_i})^2}}$
Ranking
$MR_i = |PCC(\overleftrightarrow{F_i},\overleftrightarrow{L})|$
$\max(w_r * MR_i + w_d * MD_i)$
Final Features 94D

Fig. (2). The process of the features extraction.

After we get the 473D features, they are ranked with a novel feature ranking strategy. A novel metric is proposed for the ranking, which combined the relationship to label and the independence of the feature. The relationship to label can be calculated by the Pearson Correlation Coefficient (PCC) between the feature and the label vectors. The independence can be computed by the sum of the distances between the feature and the other features. Then we added the PCC and the average distance, of which the sum can be viewed as the ranking metric. The whole features extraction is shown in (Fig. **2**).

3. HIERARCHICAL STRUCTURED SVM

Repeated information should not be reported in the text of an article. A calculation section must include experimental data, facts and practical development from a theoretical perspective.

Server classification methods have also been employed and subsequently improved for categorizing remote homology proteins. Many of these researchers have chosen to use artificial neural networks [40-46], spiking neural models [47-49], versus time-consuming profile features [50]. Considering Bio-inspired, the imbalance between the training data and the massive amount of negative samples, semi-supervised classification learning methods have also been used to solve this problem [51]. Support vector machines with modified kernels are the major classifiers used in recent the discriminative methods [52-54], since it can solve the "small samples" problem and avoid overfitting. Training and testing samples can be protein sequences instead of feature vectors, using the modified kernels. Similar to feature extraction methods, kernels can also be categorized into three groups. These are: sequence features kernels [55-57], alignment based kernels [58-60], and profile based kernels [61]. The sequence features kernels run fast, however, do not achieve adequate results. Alignment based kernels fully rely on the complete training protein sequences to improve performance. They are much slower than sequence features kernels. Profile based kernels involved all of the proteins' evolutionary information to compute similarity against the known proteins, so they are the most time consuming [62]. Sometimes, optimized parameters with RBF kernel may cause the overfitting. Therefore, we choose the linear kernel for our structured SVM.

It is known that there is the hierarchical structure among the different protein folds. However, the hierarchical structure of the labels is ignored by the common multi-classes classifiers. In this paper, we included the hierarchical structure information via revising the loss function for SVM. In the common SVM, the margin distances between different labels are all viewed as 1. In contrast, the margin distance is a value more than 0 and less than 1. The hierarchical structure includes different margin distances and different loss penalties.

For the common SVM, the optimized function is

$$arg\min_{w,\xi}\frac{1}{2}\left\|w\right\|^2+\frac{C}{N}\sum_{i}^{N}\xi_i \quad s.t.\ y_i(w^T x_i+b)\ge 1-\xi_i,\xi_i>0$$

We revised the constraints into:

$$\forall_i,\quad \forall\hat{y}\ne y_i\ :\ <w,\phi(x_i,y_i)>\ \ge\ \Delta(y_i,\hat{y})-\xi_i$$

where $\Delta(y_i,\hat{y})$ is the loss function, which can involve the hierarchical information of the labels. $\hat{y}$ is the real label for y_i. If they belong to the same protein class, the penalty would be less; while if they belong to different protein classes, the penalty would be more. The optimization prob-

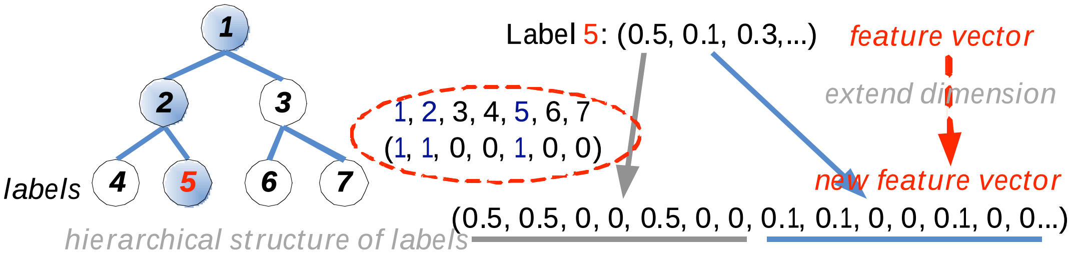

Fig. (3). Features extension for the hierarchical labels.

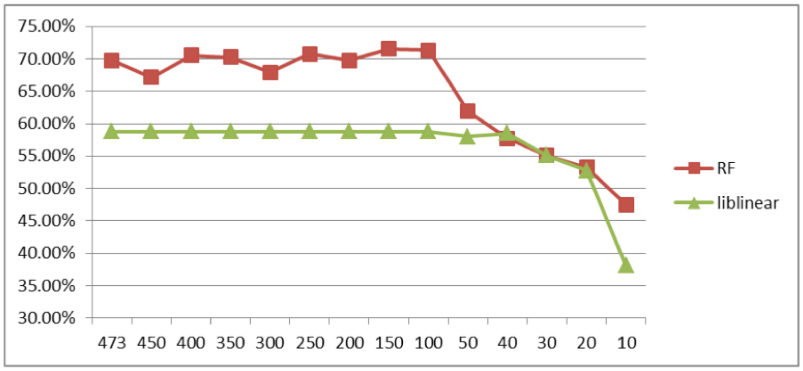

Fig. (4). Performance of the feature reduction on DD dataset.

lem is solved using the cutting plane algorithm in the SVM^{Struct} software package [63].

Feature vectors in the training set are also extended for involving the hierarchical labels information. If there are 4 labels with a hierarchical structure in (Fig. **3**), the feature vector would be extended as shown in (Fig. **3**). First, there is a path from root to the label 5. Therefore, for every feature with label 5, it would be extended to 7 times and the 1st, 2nd, and 5th ones are the features themselves, however, the other elements are 0. For an n-dimensional vector with the hierarchical structure labels as (Fig. **3**), the extended vector would be 7n-dimensional. After the extension, the distance between the vectors can reflect the distance in the hierarchical labels.

4. EXPERIMENTS

As mentioned above, DD dataset is always employed as benchmark. The training set in DD dataset includes 311 protein sequences, while the testing set includes 383 ones. They belong to 27 main protein folds in SCOP (Structural Classification of Proteins). All the sequences are filtered to ensure that each pairwise similarity is less than 0.4.

In order to test the performance of our method on big data, three more big datasets were employed, including EDD$_{new}$, F95$_{new}$, and F194$_{new}$. There datasets were extracted from the latest version of SCOP, and constructed similarly to DD dataset. EDD$_{new}$ dataset includes 3,625 protein sequences from the same 27 folds with the DD. F95$_{new}$ dataset contains 6,791 protein sequences from 95 folds. F194$_{new}$ dataset summarized 8,525 protein sequences from 194 folds. Moreover, all the protein sequences in the above three datasets also have less than 40% sequence similarity.

In order to test the performance of Structured SVM, we compared it with common SVM with linear kernel function. We tested the performance on the DD dataset with different features. First, we extract the 188D features as [64]. We also tested the performance of Wei's 473D features. Table **1** showed the comparison. We can see that structured SVM can outperform SVM with linear kernel.

Then we did the analysis of the features contribution, which was helpful for reducing the feature dimension. We ranked the features contribution and left the more contributed features from 10D to 473D. We tested the performance with random forest and liblinear (SVM with linear kernel) [65]. We can see from (Fig. **4**) that, the performance will be maintained until the feature is less than 90. In the experiments on structured SVM, we found that the 473D features performed the same as 90D. So we didn't reduce the features for structured SVM.

We also compared our methods with the state-of-the-art method. Other 6 methods are compared as shown in Table **2**. We can conclude that our methods outperform other state-of-the-art methods in the DD dataset. We should emphasize that our method employed SVM, which aims at maximum classification margin distance and suits for small sample sizes. Some other works employed ensemble classifier, which could cause overfitting for the small sample problem, especially for DD dataset.

Table 1. Comparison of different features and classifiers on DD dataset.

	188D	473D
Structured SVM	57.03%	66.63%
Liblinear	54.31%	61.36%

In order to avoid the overfitting and test our method on more big datasets, we also compared our method with Liblinear on other three bigger datasets, including EDD_{new}, $F95_{new}$, $F194_{new}$. EDD_{new} contains 27 classes which is the same as DD datasets. However, it contains more samples than DD. $F95_{new}$ contains 95 classes, while $F194_{new}$ contains 194 classes. The samples and classes of them are both much more than the DD dataset. From Table **3** we can see that, our features also work well on the bigger datasets. Structured SVM can outperform Liblinear since it involved the labels hierarchical information.

Table 2. Comparison with the state-of-the-art methods in DD dataset.

Methods	References	Overall Accuracy
Shamim	[66]	60.5%
HKNN	[67]	57.1%
Nanni *et al.*	[68]	61.1%
PFP-Pred	[69]	62.1%
ACCFold_AC	[70]	65.3%
ACCFold_ACC	[70]	66.6%
StructSVM	This paper	66.63%

Table 3. Performance of our method on different datasets.

Dataset	Liblinear	Structured SVM
DD	61.36%	66.63%
EDD_{new}	59.94%	64.53%
$F95_{new}$	63.25%	67.03%
$F194_{new}$	59.25%	62.73%

5. CONCLUSION

Protein folds prediction is still a basic and essential problem in proteomics research. Current related researches focused on the accuracy improvement, which has brought overfitting for folds prediction. In this paper, we employed SVM as the basic classifier, which aimed at the maximum margin distance rather than accuracy. SVM is viewed as the best solution to the small sample sizes and overfitting problem. Furthermore, we improved the basic SVM and involved the hierarchical label information, which just suits for the hierarchical folds relationship and can improve the prediction accuracy. We did several experiments on different benchmark datasets. The results showed that our structured SVM outperformed common linear kernel SVM in all the tested datasets. So we can conclude that structured SVM is more suitable than common SVM, as the classifier for protein folds prediction.

As a future research topic, it also deserves to the possibility that the bio-inspired computing models and algorithms are used for protein folds prediction, such as the P systems (inspired from the structure and the functioning of cells) [71-74], and evolutionary computation (motivated by evolution theory of Darwin) [75, 76].

Besides protein folds prediction, there are also some other multi-classes classification problems with hierarchical labels, including enzyme classification [77], microRNA family prediction [78-82] *etc*. Our works also yield insights for these researches. The extension of features in our work will expand the data, [83] which would be the extension works in the future.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

This work was supported by the Qinhuangdao Key Technology R&D Program (No.201502A217), and the Natural Science Foundation of China (No.61370010).

REFERENCES

[1] Andreeva, A.; Howorth, D.; Chandonia, J. M.; Brenner, S. E.; Hubbard, T. J. P.; Chothia, C.; Murzin, A. G. Data growth and its impact on the SCOP database: new developments. *Nucleic Acids Research*, **2007**, *36*(suppl 1): 419-425.

[2] Liu, X.; Suo, J.; Leung, S. C. H.; Liu, J.; Zeng, X. The power of time-free tissue P systems: Attacking NP-complete problems. *Neurocomputing*, **2015**, *159*: 151-156.

[3] Boratyn, G. M.; Camacho, C.; Cooper, P. S.; Coulouris, G.; Fong, A.; Ma, N.; Madden, T. L.; Matten, W. T.; McGinnis, S. D.; Merezhuk, Y.; Raytselis, Y.; Sayers, E. W.; Tao, T.; Ye, J.; Zaretskaya, I. BLAST: a more efficient report with usability improvements. *Nucleic Acids Research*, **2013**, *41*(W1): W29-W33.

[4] Qiwen, D.; Xiaolong, W.; Lei, L. Application of latent semantic analysis to protein remote homology detection. *Bioinformatics*, **2006**, *22*(3): 285-290.

[5] Liu, B.; Chen, J.; Wang, X. Application of Learning to Rank to protein remote homology detection. *Bioinformatics*, **2015**, *31*(21): 3492-3498.

[6] M. Muda, H.; Saad, P.; M. Othman, R. Remote protein homology detection and fold recognition using two-layer support vector machine classifiers. *Computers in Biology and Medicine*, **2011**, *41*: 687-699.

[7] Rangwala, H.; Karypis, G. Building multiclass classifiers for remote homology detection and fold recognition. *BMC Bioinformatics*, **2006**, *7*: 455.

[8] Lin, C.; Zou, Y.; Qin, J.; Liu, X.; Jiang, Y.; Ke, C.; Zou, Q. Hierarchical Classification of Protein Folds Using a Novel Ensemble Classifier. *PLoS One*, **2013**, *8*(2): e56499.

[9] Li, L.; Noble, W. S. Combining Pairwise Sequence Similarity and Support Vector Machines for Detecting Remote Protein Evolutionary and Structural Relationships. *Journal of Computational Biology*, **2003**, *10*(6): 857-868.

[10] Liu, B.; Xu, J.; Zou, Q.; Xu, R.; Wang, X.; Chen, Q. Using distances between Top-n-gram and residue pairs for protein remote homology detection. *BMC Bioinformatics* **2014**, *15*(Suppl 2): S3.

[11] Liu, B.; Wang, X.; Zou, Q.; Dong, Q.; Chen, Q. Protein Remote Homology Detection by Combining Chou's Pseudo Amino Acid Composition and Profile-Based Protein Representation. *Molecular Informatics*, **2013**, *32*: 775-782.

[12] Chen, W.; Liu, X.; Huang, Y.; Jiang, Y.; Zou, Q. Lin, C. Improved method for predicting protein fold patterns with ensemble classifiers. *Genetics and Molecular Research*, **2012**, *11*(1): 174-181.

[13] Zhao, X.; Zou, Q.; Liu, B.; Liu, X. Exploratory predicting protein folding model with random forest and hybrid features. *Current Proteomics*, **2014**, *11*(4): 289-299.

[14] Zakeri, P.; Jeuris, B.; Vandebri, R.; Moreau, Y. Protein fold recognition using geometric kernel data fusion. *Bioinformatics*, **2014**, *30*(13): 1850-1857.

[15] H. , D. C. . I. , D. Multi-class protein fold recognition using support vector machines and neural networks. *Bioinformatics*, **2001**, *17*: 349-358.

[16] Shen, H. -B.; Chou, K. -C. Ensemble classifier for protein fold pattern recognition. *Bioinformatics*, **2006**, *22*: 1717-1722.

[17] Nanni, L. A novel ensemble of classifiers for protein fold recognition. *Neurocomputing*, **2006**, *69*: 2434-2437.

[18] Wei, L.; Liao, M.; Gao, X.; Zou, Q. An Improved Protein Structural Prediction Method by Incorporating Both Sequence and Structure Information. *IEEE Transactions on Nanobioscience*, **2015**, *14*(4): 339-349.

[19] Yang, J. Y.; Chen, X. Improving taxonomy‐based protein fold recognition by using global and local features. *Proteins: Structure, Function, and Bioinformatics*, **2011**, *79*: 2053-2064.

[20] Wei, L.; Liao, M.; Gao, X.; Zou, Q. Enhanced Protein Fold Prediction Method through Novel Feature Extraction Technique. *IEEE Transactions on Nanobioscience*, **2015**, *14*(6): 649-659.

[21] Lin, C.; Chen, W.; Qiu, C.; Wu, Y.; Krishnan, S.; Zou, Q. LibD3C: Ensemble Classifiers with a Clustering and Dynamic Selection Strategy. *Neurocomputing*, **2014**, *123*: 424-435.

[22] Liu, B.; Wang, S.; Wang, X. DNA binding protein identification by combining pseudo amino acid composition and profile-based protein representation. *Scientific Reports*, **2015**, *5*: 15479.

[23] Zou, Q.; Chen, W.; Huang, Y.; Liu, X.; Jiang, Y. Identifying Multi-functional Enzyme with Hierarchical Multi-label Classifier. *Journal of Computational and Theoretical Nanoscience*, **2013**, *10*(4): 1038-1043.

[24] Zeng, X.; Yuan, S.; Huang, X.; Zou, Q. Identification of cytokine via an improved genetic algorithm. *Frontiers of Computer Science*, **2015**: Doi: 10. 1007/s11704-11014-14089-11703.

[25] Song, L.; Li, D.; Zeng, X.; Wu, Y.; Guo, L.; Zou, Q. nDNA-prot: Identification of DNA-binding Proteins Based on Unbalanced Classification. *BMC Bioinformatics*, **2014**, *15*: 298.

[26] Liu, B.; Xu, J.; Fan, S.; Xu, R.; Zhou, J.; Wang, X. PseDNA-Pro: DNA-Binding Protein Identification by Combining Chou's PseAAC and Physicochemical Distance Transformation. *Molecular Informatics*, **2015**, *34*(1): 8-17.

[27] Liu, B.; Xu, J.; Lan, X.; Xu, R.; Zhou, J.; Wang, X.; Chou, K. -C. iDNA-Prot|dis: Identifying DNA-Binding Proteins by Incorporating Amino Acid Distance-Pairs and Reduced Alphabet Profile into the General Pseudo Amino Acid Composition. *PLoS ONE*, **2014**, *9*(9): e106691.

[28] Yu, J.; Guo, M.; Needham, C. J.; Huang, Y.; Cai, L.; Westhead, D. R. Simple sequence-based kernels do not predict protein-protein interactions. *Bioinformatics*, **2010**, *26*(20): 2610-2614.

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

[30] Liu, B.; Liu, F.; Fang, L.; Wang, X.; Chou, K. -C. repRNA: a web server for generating various feature vectors of RNA sequences. *Molecular Genetics and Genomics*, **2015**, *DOI: 10. 1007/s00438-015-1078-7*.

[31] Liu, B.; Liu, F.; Fang, L.; Wang, X.; Chou, K. -C. repDNA: a Python package to generate various modes of feature vectors for DNA sequences by incorporating user-defined physicochemical properties and sequence-order effects. *Bioinformatics*, **2015**, *31*(8): 1307-1309.

[32] Saigo, H.; Philippe, J.; Nobuhisa, V.; Akutsu, T. Protein homology detection using string alignment kernels. *Bioinformatics*, **2004**, *20*(11): 1682-1689.

[33] Leslie, C.; Eskin, E.; A; Westopn, J.; Noble, W. Mismatch String Kernels for Discriminative Protein Classification. *Bioinformatics*, **2004**, *20*: 467-476.

[34] Ogul, H.; Mumcuoglu, E. A discriminative method for remote homology detection based on n-peptide compositions with reduced amino acid alphabets. *BioSystems*, **2007**, *87*: 75-81.

[35] Zou, Q.; Hu, Q.; Guo, M.; Wang, G. HAlign: Fast Multiple Similar DNA/RNA Sequence Alignment Based on the Centre Star Strategy. *Bioinformatics*, **2015**, *31*(15): 2475-2481.

[36] Kaushik, S.; Mutt, E.; Chellappan, A.; Sankaran, S.; Srinivasan, N.; Sowdhamini, R. Improved detection of remote homologues using cascade PSI-BLAST: Influence of Neighbouring Protein Families on Sequence Coverage. *PLoS One*, **2013**, *8*(2): e56449.

[37] Liu, X.; Zhao, L.; Dong, Q. Protein remote homology detection based on auto cross covariance transformation. *Computers in Biology and Medicine*, **2011**, *41*: 640–647.

[38] Liu, B.; Wang, X.; Chen, Q.; Dong, Q.; Lan, X. Using Amino Acid Physicochemical Distance Transformation for Fast Protein Remote Homology Detection. *PLoS ONE*, **2012**, *7*(9): e46633.

[39] Zou, Q.; Wang, Z.; Guan, X.; Liu, B.; Wu, Y.; Lin, Z. An Approach for Identifying Cytokines Based On a Novel Ensemble Classifier. *BioMed Research International*, **2013**, *2013*: 686090

[40] Song, T.; Pan, L. Spiking Neural P Systems with Rules on Synapses Working in Maximum Spiking Strategy. *IEEE Trans on Nanobioscience*, **2015**, *14*(4): 465-477.

[41] Zhang, X.; Pan, L.; Păun, A. On universality of axon P systems. *IEEE Transactions on Neural Networks and Learning Systems*, **2015**, *26*(11): 2816-2829.

[42] Song, T.; Liu, X.; Zeng, X. Asynchronous Spiking Neural P Systems with Anti-Spikes. *Neural Processing Letters*, **2014**: 1-15.

[43] Zhang, X.; Wang, B.; Pan, L. Spiking neural P systems with a generalized use of rules. *Neural Computation*, **2014**, *26*(12): 2925-2943.

[44] Zeng, X. Simulating Spiking Neural P Systems with Circuits. *Journal of Computational and Theoretical Nanoscience*, **2015**, *12*(9): 2023-2026.

[45] Zhang, X.; Zeng, X.; Luo, B.; Pan, L. On some classes of sequential spiking neural P systems. *Neural Computation*, **2014**, *26*(5): 974-997.

[46] Zeng, X.; Pan, L.; Pérez-Jiménez, M. J. Small universal simple spiking neural P systems with weights. *Science China Information Sciences*, **2014**, *57*(9): 1-11.

[47] Song, T.; Pan, L. On the Universality and Non-universality of Spiking Neural P Systems with Rules on Synapses. *IEEE Trans on Nanobioscience*, **2015**.

[48] Song, T.; Pan, L. Normal Forms for Some Classes of Sequential Spiking Neural P Systems. *IEEE Trans on Nanobioscience*, **2012**, *4*(11): 352-359.

[49] Song, T.; Pan, L. Spiking Neural P Systems with Rules on Synapses Working in Maximum Spikes Consumption Strategy. *IEEE Trans on Nanobioscience*, **2015**, *14*(1): 37-43.

[50] Hochreiter, S.; Heusel, M.; Obermayer, K. Fast model-based protein homology detection without alignment. *Bioinformatics*, **2007**, *23*(14): 1728-1736.

[51] Shah, A. R.; Oehmen, C. S.; Webb-Robertson, B. -J. SVM-HUSTLE-an iterative semi-supervised machine learning approach for pairwise protein remote homology detection. *Bioinformatics*, **2008**, *26*(4): 783-790.

[52] Jianlin, C.; Pierre, B. A machine learning information retrieval approach to protein fold recognition. *Bioinformatics*, **2006**, *22*(12): 1456-1463.

[53] Lingner, T.; Meinicke, P. Remote homology detection based on oligomer distances. *Bioinformatics*, **2006**, *22*(18): 2224-2231.

[54] Liu, B.; Fang, L.; Long, R.; Lan, X.; Chou, K. -C. iEnhancer-2L: a two-layer predictor for identifying enhancers and their strength by pseudo k-tuple nucleotide composition. *Bioinformaitcs*, **2015**, *DOI: 10. 1093/bioinformatics/btv604*.

[55] Handstad, T.; Hestnes, A. J.; Sætrom, P. Motif kernel generated by genetic programming improves remote homology and fold detection. *BMC Bioinformatics*, **2007**, *8*: 23.

[56] Ben-Hur, A.; Brutlag, D. Remote homology detection: A motif based approach. *Bioinformatics*, **2003**, *19*(Suppl 1): 26–33.

[57] Hou, Y.; Hsu, W.; Lee, M. L.; Bystroff, C. Efficient Remote Homology Detection Using Local Structure. *Bioinformatics* **2003**, *19*(17): 2294–2301.

[58] Webb-Robertson, B. -J.; Oehmen, C.; Matzke, M. SVM-BALSA: Remote homology detection based on Bayesian sequence alignment. *Computational Biology and Chemistry*, **2005**, *29*(6): 440–443.

[59] Saigo, H.; Vert, J.; Ueda, N.; Akutsu, T. Protein Homology Detection Using String Alignment Kernels. *Bioinformatics* **2004**, *20*: 1682–1689.

[60] Lingner, T.; Meinicke, P. Word correlation matrices for protein sequence analysis and remote homology detection. *BMC Bioinformatics*, **2008**, *9*: 259.

[61] Rangwala, H.; Karypis, G. Profile-based direct kernels for remote homology detection and fold recognition. *Bioinformatics*, **2005**, *21*(23): 4239-4247.

[62] Liu, B.; Zhang, D.; Xu, R.; Xu, J.; Wang, X.; Chen, Q.; Dong, Q.; Chou, K. -C. Combining evolutionary information extracted from frequency profiles with sequence-based kernels for protein remote homology detection. *Bioinformatics*, **2014**, *30*(4): 472-479.

[63] Joachims, T.; Finley, T.; Yu, C. Cutting-Plane Training of Structural SVMs. *Machine Learning*, **2009**, *77*(1): 27-59.

[64] Xu, J.; Zhang, J.; Han, B.; Liang, L.; Ji, Z. CytoSVM: an advanced server for identification of cytokine-receptor interactions. *Nucleic Acids Research*, **2007**, *35*: W538–W542.

[65] Fan, R. -E.; Chang, K. -W.; Hsieh, C. -J.; Wang, X. -R.; Lin, C. -J. LIBLINEAR: A Library for Large Linear Classification. *Journal of Machine Learning Research*, **2008**, *9*: 1871-1874.

[66] Shamim, M. T. A.; Anwaruddin, M.; Nagarajaram, H. A. Support Vector Machine-based classification of protein folds using the structural properties of amino acid residues and amino acid residue pairs. *Bioinformatics*, **2007**, *23*: 3320-3327.

[67] Oleg, O. Protein fold recognition with k-local hyperplane distance nearest neighbor algorithm. In: *Proceedings of the Second European Workshop on Data Mining and Text Mining in Bioinformatics, Pisa, Italy: 2004*; **2004**: 51-57.

[68] Nanni, L. A novel ensemble of classifiers for protein fold recognition. *Neurocomputing*, **2006**, *69*(16): 2434-2437.

[69] Shen, H. -B.; Chou, K. -C. Ensemble classifier for protein fold pattern recognition. *Bioinformatics*, **2006**, *22*(14): 1717-1722.

[70] Qiwen, D.; Shuigeng, Z.; Jihong, G. A new taxonomy-based protein fold recognition approach based on autocross-covariance transformation. *Bioinformatics*, **2009**, *25*(20): 2655-2662.

[71] Zhang, X.; Liu, Y.; Luo, B.; Pan, L. Computational power of tissue P systems for generating control languages. *Information Sciences*, **2014**, *278*(10): 285-297.

[72] Zeng, X.; Zhang, X.; Song, T.; Pan, L. Spiking neural P systems with thresholds. *Neural computation*, **2014**, *26*(7): 1340-1361.

[73] Zeng, X.; Xu, L.; Liu, X.; Pan, L. On languages generated by spiking neural P systems with weights. *Information Sciences*, **2014**, *278*: 423-433.

[74] Zhang, X.; Wang, B.; Ding, Z.; Tang, J.; He, J. Implementation of membrane algorithms on GPU. *Journal of Applied Mathematics*, **2014**, *2014*: 307617.

[75] Zhang, X.; Tian, Y.; Jin, Y. A knee point driven evolutionary algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation*, **2015**, *19*(6): 761-776.

[76] Zhang, X.; Tian, Y.; Cheng, R.; Jin, Y. An efficient approach to non-dominated sorting for evolutionary multi-objective optimization. *IEEE Transactions on Evolutionary Computation*, **2015**, *19*(2): 201-213.

[77] Cheng, X. -Y.; Huang, W. -J.; Hu, S. -C.; Zhang, H. -L.; Wang, H.; Zhang, J. -X.; Lin, H. -H.; Chen, Y. -Z.; Zou, Q.; Ji, Z. -L. A global characterization and identification of multifunctional enzymes. *PLoS One*, **2012**, *7*(6): e38979.

[78] Zou, Q.; Mao, Y.; Hu, L.; Wu, Y.; Ji, Z. miRClassify: An advanced web server for miRNA family classification and annotation. *Computers in Biology and Medicine*, **2014**, *45*: 157-160.

[79] Wang, Q.; Wei, L.; Guan, X.; Wu, Y.; Zou, Q.; Ji, Z. Briefing in family characteristics of microRNAs and their applications in cancer research. *BBA - Proteins and Proteomics*, **2014**, *1844*: 191-197.

[80] Wei, L.; Liao, M.; Gao, Y.; Ji, R.; He, Z.; Zou, Q. Improved and Promising Identification of Human MicroRNAs by Incorporating a High-quality Negative Set. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **2014**, *11*(1): 192-201

[81] Liu, B.; Fang, L.; Liu, F.; Wang, X.; Chen, J.; Chou, K. -C. Identification of real microRNA precursors with a pseudo structure status composition approach. *PLoS ONE*, **2015**, *10*(3): e0121501.

[82] Liu, B.; Fang, L.; Jie, C.; Liu, F.; Wang, X. miRNA-dis: microRNA precursor identification based on distance structure status pairs. *Molecular BioSystems*, **2015**, *11*: 1194-1204.

[83] Zou, Q.; Li, X.; Jiang, W.; Lin, Z.; Li, G.; Chen, K. Survey of MapReduce Frame Operation in Bioinformatics. *Briefings in Bioinformatics*, **2014**, *15*(4): 637-647.

Received: November 20, 2015 Revised: December 16, 2015 Accepted: December 20, 2015