Journal of Bioinformatics and Computational Biology
Vol. 4, No. 6 (2006) 1181–1195

Imperial College Press
www.icpress.co.uk

PROTEIN SUBCELLULAR LOCALIZATION BASED ON PSI-BLAST AND MACHINE LEARNING

JIAN GUO*, XIAN PU†, YUANLIE LIN* and HOWARD LEUNG†

*Laboratory of Statistical Computation, Department of Mathematical Sciences
Tsinghua University

†Department of Computer Science, The City University of Hong Kong

Received 19 June 2006
Revised 2 August 2006
Accepted 2 August 2006

Subcellular location is an important functional annotation of proteins. An automatic, reliable and efficient prediction system for protein subcellular localization is necessary for large-scale genome analysis. This paper describes a protein subcellular localization method which extracts features from protein profiles rather than from amino acid sequences. The protein profile represents a protein family, discards part of the sequence information that is not conserved throughout the family and therefore is more sensitive than the amino acid sequence. The amino acid compositions of whole profile and the N-terminus of the profile are extracted, respectively, to train and test the probabilistic neural network classifiers. On two benchmark datasets, the overall accuracies of the proposed method reach 89.1% and 68.9%, respectively. The prediction results show that the proposed method perform better than those methods based on amino acid sequences. The prediction results of the proposed method are also compared with Subloc on two redundance-reduced datasets.

Keywords: Subcellular localization; probabillstic neural network; position-specific scoring matrix; multiple sequence alignment; PSI-BLAST.

1. Introduction

Subcellular location of the proteins is an important cue for inferring on their functional characteristic, interaction partners and potential roles in the cellular machinery. Determination of subcellular localization via experimental processes is often time-consuming and laborious, therefore, a number of *in-silico* subcellular localization methods have been proposed in the past decade. These methods can be generally categorized into the following groups. The first group locates the proteins based on the existence of the sorting signals,[37] which include signal peptides, mitochondrial targeting peptides, and chloroplast transit peptides.[21,39,40] The second group studies the whole sequence information such as the composition of the amino acid[7,9,17,18,31,38,44,58] and the composition of the amino acid pairs.[28,32,42,54] The third group uses the concept of pseudo amino acid composition

(PseAA) originally proposed by Chou[10] to extract information through a set of discrete correlation factors and various biochemical properties.[8,13,22–24,41,46,51,52,58] The fourth group[5,11,12,14,15] used the protein sample representation derived from a higher-level database, such as functional domain (FunD) database, gene ontology (GO) database, or their combination. The last group applied information fusion techniques to integrate different prediction methods. For example, PSORT-B[25,26] integrates the feature of the amino acid composition, the similarity to proteins of known location, the signal peptides, the transmembrane alpha-helices, and the motifs corresponding to specific localizations. Bhasin *et al.*[2,3] and Garg *et al.*[27] predicted subcellular locations by fusing the amino acid composition, the composition of residue pairs, the composition of physico-chemical properties, and direct BLAST search. With the development of human proteome project, subcellular localization of human proteins begins to abstract more attention and some pioneering study has been done by Garg *et al.*[27] and Chou and Shen.[19]

This paper introduces an approach for eukaryotic protein subcellular localization. The core idea of the proposed method (named as PNNSubPro) lies in the assumption that protein profile provides more information and results in more reliable prediction of subcellular localization. Compared with the amino acid sequence, protein profile derived from the multiple alignment program involves more common characters of a family of proteins. In other words, protein profile concerns about the conserved regions of this protein family and discarded the region not conserved. In this work, the probabilistic neural network classifier is used to train and test the features extracted from the protein profiles. The results show that the proposed method has better performances than those methods based on amino acid sequence.

2. Materials and Methods

2.1. *Data sets*

Two datasets were used to test the performance of the proposed method. The first one is Reinhardt and Hubbard's[44] eukaryotic protein dataset, which has been used extensively to evaluate some existing subcellular locations methods such as NNPSL,[44] Subloc,[31] Fuzzy k-NN,[32] and ESLpred.[3] The proteins in this database were extracted from SWISSPORT 33.0 and the sequences were filtered as follows:

(1) only those appeared to be complete and having reliable annotations were kept;
(2) transmembrane proteins were excluded[31,44] because reliable methods for predicting these proteins have been well developed;[6,16,30,33,45]
(3) plant proteins were also removed to ensure sufficient difference in composition.

The resulting dataset comprises 2427 eukaryotic proteins (684 cytoplasm, 325 extracellular, 321 mitochondrial, and 1097 nuclear proteins).

The second dataset, introduced by Huang and Li,[32] was created by selecting all eukaryotic proteins with annotated subcellular locations from SWISSPROT 41.0.

Similar to the construction process of Reinhardt and Hubbard's dataset, the transmembrane proteins were excluded. The remaining proteins were filtered by BLAST with identity cutoff set to 50%. The final dataset comprises 3572 proteins (622 cytoplasm, 1188 nuclear, 424 mitochondria, 915 extracellular, 26 golgi apparatus, 225 chloroplast, 45 endoplasmic reticulum, 7 cytoskeleton, 29 vacuole, 47 peroxisome, and 44 lysosome).

2.2. *Probabilistic neural network*

The probabilistic neural network (PNN)[48] is a powerful machine learning technique. The original PNN was designed to solve some drawbacks of the traditional backpropagation neural network, such as the long training time and the false minimum problem. The idea of PNN is based on the well-established statistical principles derived from Bayes Decision Rule and non-parametric kernel based estimators of probability density functions.

Consider a pattern vector $\mathbf{x} \in \mathcal{R}^m$ in a C-classification problem. Based on Bayes Decision Rule, $\mathbf{x}$ belongs to class $k, (1 \leq k \leq C)$ if and only if

$$h_k f_k(\mathbf{x}) > h_i f_i(\mathbf{x}), \quad 1 \leq i \leq C, \quad i \neq k \tag{1}$$

where h_k and h_i are the prior probability of the occurrence of the patterns from class k and class i, and f_k and f_i are the probabilistic density function of the samples in class k and class i, respectively. Usually the prior probability is known or can be assumed to be evenly. Therefore, the key point to apply Eq. (1) is how to estimate the probability density functions from the training samples.

The PNN is interpreted as a function which approximates the probability densities of the underlying distribution of the training samples. A nonparametric estimate method known as Parzen Window[43] is used to construct the class-dependent probability density functions for each class. Denote the jth training sample in the ith class as $\mathbf{x}_i^{(j)}$, then the Parzen estimate of the probability density function for the ith class is:

$$f_i(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{m}{2}} \sigma^m n} \sum_{j=1}^{n_i} \exp\left[-\frac{(\mathbf{x} - \mathbf{x}_i^{(j)})^{\mathrm{T}} (\mathbf{x} - \mathbf{x}_i^{(j)})}{2\sigma^2} \right] \tag{2}$$

where $n^{(i)}$ is the number of the training samples in the ith class, m is the dimension of the samples and σ is called "smooth parameter". To simulate the form of Eq. (2), the architecture of PNN is composed of four layers: input layer, pattern layer, summation layer, and output layer (see Fig. 1). The input comprises m (m equals the dimension of the feature vector) merely distributional units that supply the same input values to all of the pattern units in the pattern layer. The pattern layer comprises n_{T} neurons, where n_{T} is the number of the training samples. The pattern unit outputs the inner-product of each weight vector (feature vector) and the test

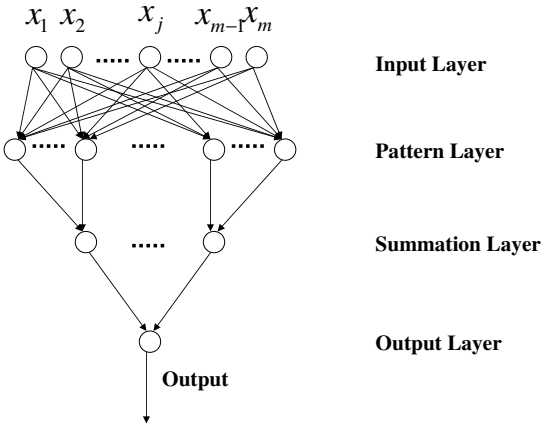

Fig. 1. The 4-layer structure of the probabilistic neural network.

example. After that, the product is transformed by the activation function:

$$g(\mathbf{x}) = \exp\left(\frac{\mathbf{x}^{\mathrm{T}} w_{k,i} - 1}{\sigma^2}\right) \tag{3}$$

where $w_{k,i}$ is the weights from the kth unit in the first layer to the ith unit in the second layer. The parameter determines the width of an area in the input space to which each neuron responds. A larger σ leads to a larger area around the input vector, where the radial basis function responds with significant output. In the summation layer, the ith unit ($1 \leq i \leq C$) simply sums the outputs of the units corresponding to the ith class. The output layer decides the predicted labels from the summation layer by a Max–Win-All strategy. Specifically, the test sample is classified to the class with maximal value of all units in the summation layer.

2.3. *Position-specific scoring matrix*

Each protein sequence (called query sequence) in the proposed dataset was used as a seed to search and align homogenous sequences from the SWISSPROT 46.0[4] protein database using the PSI-BLAST program[1] with parameters h and j set to 0.001 and 3, respectively. The aligned sequences are further converted into position-specific scoring matrices (PSSMs) to express their homogenous information. PSSM is a matrix with 20 rows and L columns, where L is the total number of amino acids in the query sequence. The (i, j)th entry of the matrix represents the chance of the amino acid in the jth position of the query sequence being mutated to amino acid type i during the evolution process.

For convenience, let us denote

$$\mathbf{P}^{(i)} = [\mathbf{p}_1^{(i)}, \mathbf{p}_2^{(i)}, \ldots, \mathbf{p}_{n_i}^{(i)}]$$

as the PSSM of the ith sequence, where

$$\mathbf{p}_j^{(i)} = [p_{j,1}^{(i)}, p_{j,2}^{(i)}, \ldots, p_{j,20}^{(i)}]^{\mathrm{T}}, \quad 1 \le j \le n_i,$$

and n_i is the total number of amino acids of the ith sequence.

2.3.1. *Features from PSSM*

Each protein in the proposed method is represented by two features extracted from its PSSM. The first feature is the amino acid composition of whole PSSM and the second one is the combination of the amino acid compositions of whole PSSM and the N-terminus of PSSM.

Feature 1

Feature 1 extracts the amino acid composition from whole PSSM. Denote

$$\mathbf{x}^{(i)} = [x_1^{(i)}, x_2^{(i)}, \ldots, x_{20}^{(i)}],$$

as the 20-dimensional feature vector of the ith protein. $x_k^{(i)} \quad (1 \le k \le n_i)$ is the composition of the kth amino acid in the PSSM of the ith protein and it is calculated as follows:

$$x_k^{(i)} = \frac{1}{n_i} \sum_{j=1}^{n_i} p_{j,k}^{(i)} \tag{4}$$

where $\mathbf{x}$ is input into a PNN classifier for training and testing.

The prediction method based on feature 1 and the PNN classifier is denoted as "PNNSubPro[1]".

Feature 2

Feature 2 uses the similar extraction approach as module 1 but it also computes the amino acid composition of N-terminus of the PSSM. Specifically, denote the amino acid composition of the N-terminus of the PSSM of the ith protein as

$$\mathbf{y}^{(i)} = [y_1^{(i)}, y_2^{(i)}, \ldots, y_{20}^{(i)}].$$

Here, $y_k^{(i)} \quad (1 \le k \le 20)$ is calculated as follows:

$$y_k^{(i)} = \frac{1}{L_N} \sum_{j=1}^{L_N} p_{j,k}^{(i)} \tag{5}$$

where L_{N} is the numbers of amino acids in the N-terminus of the ith protein. Then the feature vector extracted by this module is defined as:

$$\mathbf{x} \oplus \mathbf{y} = [x_1^{(i)}, \ldots, x_{20}^{(i)}, y_1^{(i)}, \ldots, y_{20}^{(i)}] \tag{6}$$

where $\oplus$ is the operator of the concatenation. In this paper, the length of the N-terminus L_N equals 30.

The prediction method based on feature 2 and the PNN classifier is denoted as "PNNSubPro^2".

2.4. *Assessment of performance*

This paper uses the leave-one-out cross validation (jackknife test) to evaluate the performance of a method on a dataset. The jackknife test is a rigorous and objective method which was elucidated in a comprehensive review[20] and a series of follow-up papers.[22,23,29,31,34,35,46,47,49–51,53,55–58] The overall accuracy (OA), the accuracy for each class (Acc), and the Matthews correlation coefficient (MCC)[36] were used to assess the prediction result.

Denote $\mathbf{M} \in \Re^{C\times C}$ as the confusion matrix of the prediction result, where C is the number of classes. Then $\mathbf{M}_{i,j}$ $(1 \le i, j \le C)$ represents the number of proteins that actually belong to class i but are predicted as class j. We further denote

$$\begin{aligned} p_c &= \mathbf{M}_{c,c}, \quad q_c = \sum_{i=1,i\neq c}^{C} \sum_{j=1,j\neq c}^{C} \mathbf{M}_{i,j}, \\ r_c &= \sum_{i=1,i\neq c}^{C} \mathbf{M}_{i,c}, \quad s_c = \sum_{j=1,j\neq c}^{C} \mathbf{M}_{c,j}, \end{aligned} \tag{7}$$

where c $(1 \le c \le C)$ is the index of a particular class. For class c, p_c is the number of true positive samples, q_c is the number of true negative samples, r_c is the number of false positive samples, and s_c is the number of false negative samples. Based on the notations above, the overall accuracy (OA), the accuracy of class c (Acc_c), and the Matthew's Correlation Coefficient of class c (MCC_c) can be calculated as:

$$\mathrm{OA} = \frac{\sum_{c=1}^{C} \mathbf{M}_{c,c}}{\sum_{i=1}^{C}\sum_{j=1}^{C} \mathbf{M}_{i,j}} \tag{8}$$

$$\mathrm{Acc}_c = \frac{\mathbf{M}_{c,c}}{\sum_{j=1}^{C} \mathbf{M}_{c,j}} \tag{9}$$

$$\mathrm{MCC}_c = \frac{p_c q_c - r_c s_c}{\sqrt{(p_c + s_c)(p_c + r_c)(q_c + s_c)(q_c + r_c)}}. \tag{10}$$

3. Result and Discussion

The parameter σ is optimized by maximizing the overall accuracy in the leave-one-out cross validation test and the prediction results on the two Reinhardt and

Hubbard's eukaryotic dataset and Huang and Li's dataset are listed in Tables 1 and 3, respectively. The parameter σ equals 0.087 and 0.081 for Tables 1 and 3, respectively.

3.1. *Result and comparison on Reinhardt and Hubbard's eukaryotic dataset*

In Table 1, the prediction results of the proposed method (PNNSubPro[1] and PNNSubPro[2]) are compared with the results of NNPSL,[44] EuPSI-BLAST,[3] Subloc,[31] Fuzzy k-NN,[32] and ESLpred.[3] The overall accuracy of PNNSubPro[1] reaches 88.3%, which is comparable with that of ESLpred (88.0%) but it is higher than that of NNPSL (66%), Subloc (79.4%), and Fuzzy k-NN (85.2%). The overall accuracy of PNNSubPro[2] (89.1%) is slightly higher than that of PNNSubPro[1]. For mitochondria, the accuracy and MCC of PNNSubPro[2] reaches 88.5% and 0.82, which is significantly higher than the corresponding results of PNNSubPro[1] and other methods in Table 1. The results imply that the N-terminus provides important information for localization of mitochondrial proteins.

The prediction results of PNNSubpro[1] and PNNSubpro[2] (Table 2) are also compared with that of EuPSI-BLAST,[3] which is a module of ESLpred. EuPSI-BLAST searches the training set to find the protein most similar to the test protein and classifies the test protein to the same class as the hit. Bhasin and Raghava[3] did not publish the overall accuracy and the MCC of EuPSI-BLAST, but we are still able to comfirm that PNNSubPro performs better than EuPSI-BLAST by comparing the accuracies of each location.

To straightly demonstrate the advantage of feature extraction from protein profiles rather than from amino acid sequences, we compared the performance of PNNSubpro[1] with PNNComp and Subloc.[31] PNNComp uses the same feature as Subloc (amino acid composition of sequence) and the same classifier as PNNSubpro[1] (PNN), so it can be regarded as a bridge between Subloc and PNNSubpro[1]. The results of the three methods are listed in Table 3. The difference between the performances of PNNSubpro[1] (88.3%) and PNNComp (81.2%) demonstrates that protein profile involves more positive information for the prediction. The overall accuracy of PNNSubpro[1] is slightly higher than that of Subloc, which implies that PNN performs better than SVM in this problem.

3.2. *Result and comparison on Huang and Li's dataset*

We also compared the results of PNNSubpro with the fuzzy k-NN method[32] on Huang and Li's dataset. To avoid overestimating, each pair of proteins in this dataset had an identity of less than 50%.[32] Huang and Li applied a fuzzy k-nearest neighbor (fuzzy k-NN) model and dipeptide frequency to predict the 11 locations in their dataset and achieved a overall accuracy of 58.1% by the leave-one-out cross validation test. The overall accuracies of PNNSubPro[1] and PNNSubPro[2] reaches 67.9% and 68.9%, which are about 10% higher than that of fuzzy k-NN (see Table 4).

Table 1. Comparison of different subcellular localization methods for Reinhardt and Hubbard's eukaryotic protein dataset.

Subcellular Location	NNPSL	EuPSI-BLAST	SubLoc		Fuzzy k-NN		ESLpred		PNNSubPro[1]		PNNSubPro[2]	
	Acc(%)	Acc(%)	Acc(%)	MCC	Acc(%)	MCC	Acc(%)	MCC	Acc(%)	MCC	Acc(%)	MCC
Cytoplasm	55	77.6	76.9	0.64	86.7	0.76	85.2	0.79	85.4	0.80	88.2	0.80
Extracellular	75	86.7	80.0	0.78	83.7	0.87	88.9	0.91	90.2	0.91	92.3	0.93
Mitochondria	61	54.8	56.7	0.58	60.4	0.63	68.2	0.69	75.4	0.73	88.5	0.82
Nuclear	72	84.5	87.4	0.75	92.0	0.83	95.3	0.87	93.3	0.87	88.9	0.86
Overall	66	–	79.4	–	85.2	–	88.0	–	88.3	–	89.1	–

NNPSL[44] and SubLoc[31] use amino acid composition as features; Fuzzy k-NN[32] extracts features by the dipeptide composition; EuPSI-BLAST[3] is one of the modules of ESLpred. It uses PSI-BLAST to search the training set and assigns the query sequence with the same subcellular location as the most similar protein in the training set. ESLpred[3] is a mixture method combining amino acid composition, dipeptide composition, physico-chemical properties, and PSI-BLAST searching; PNNSubPro[1] is the proposed method with feature 1; PNNSubPro[2] is the proposed method with feature 2. The results of SubLoc, fuzzy k-NN, PNNSubPro[1], and PNNSubPro[2] were obtained by leave-one-out cross validation. The results of NNPSL were obtained by 10-fold cross validation, and those of EuPSI-BLAST and ESLpred were obtained by fivefold cross validation. Acc: accuracy; MCC: Matthew's correlation coefficient.

Table 2. Comparison of the performance of two methods based on profiles and sequences, respectively.

Subcellular Location	Subloc		PNNComp		PNNSubPro[1]	
	Ac (%)	MCC	Acc (%)	MCC	Acc (%)	MCC
Cytoplasm	76.9	0.64	80.9	0.69	85.4	0.80
Extracellular	80.0	0.78	84.3	0.84	90.2	0.91
Mitochondria	56.7	0.58	60.1	0.59	75.4	0.73
Nuclear	87.4	0.75	86.7	0.79	93.3	0.87
Overall	79.4	–	81.2	–	88.3	–

PNNSubPro[1] is the method proposed in this paper which extracts features by amino acid composition of profiles. PNNComp uses the same PNN classifier as PNNSubPro[1] but it extracts the amino acid composition from protein sequences rather than profiles.

Table 3. Comparison of the overall accuracy of Subloc,[31] PNNSubPro,[1] and PNNSubPro[2] on redundance-reduced datasets.

Filter Threshold (%)	Number of Samples	Subloc (%)	PNNSubPro[1] (%)	PNNSubPro[2] (%)
100	2427	78.6	86.7	88.5
50	1137	66.2	72.2	78.8
20	597	59.3	62.0	71.2

Table 4. Comparison of Fuzzy k-NN with PNNSubPro[1] and PNNSubPro[2] on Huang and Li's eukaryotic protein dataset.

Subcellular Location	Fuzzy k-NN		PNNSubPro[1]		PNNSubPro[2]	
	Acc (%)	MCC	Acc(%)	MCC	Acc (%)	MCC
Cytoplasm	35.4	0.31	51.5	0.45	49.7	0.43
Nuclear	71.5	0.58	82.3	0.70	77.4	0.66
Mitochondria	36.6	0.30	57.6	0.53	66.8	0.62
Extracellular	81.6	0.54	77.8	0.77	81.3	0.78
Golgi apparatus	15.4	0.27	19.2	0.18	7.7	0.08
Chloroplast	32.4	0.36	45.8	0.42	68.0	0.62
Endoplasmic reticulum	11.1	0.22	40.0	0.35	37.8	0.37
Cytoskeleton	28.6	0.44	0.0	0.00	0.0	0.00
Vacuole	6.9	0.16	13.8	0.12	17.2	0.17
Peroxisome	14.9	0.27	46.8	0.40	29.8	0.29
Lysosome	20.5	0.31	45.5	0.41	31.8	0.33
Overall	58.1	–	67.9	–	68.9	–

Acc: accuracy; MCC: Matthew's correlation coefficient.

3.3. *Result on redundance-reduced datasets*

The two benchmark datasets used here were constructed by Reinhardt and Hubbard, and Huang and Li, respectively. The former covers only four subcellular locations allowing the inclusion of proteins with up to 90% sequence identity, and the latter covers 11 location sites allowing sequence identity up to 50%. To

Table 5. Comparison of the overall accuracy of Subloc,[31] PNNSubPro,[1] and PNNSubPro^2 on redundance-reduced datasets.

Filtering Threshold μ (%)	Number of Samples (%)	Subloc (%)	PNNSubPro^1 (%)	PNNSubPro^2 (%)
100	2427	78.6	86.7	88.5
50	1137	66.2	72.2	78.8
20	597	59.3	62.0	71.2

completely get rid of the homology or redundancy bias, an ideal dataset should be constructed according to the criterion that none of proteins has more than 35% (or better yet, 20%) to any others in a same subset (subcellular location). In addition, it is worthwhile to investigate whether the good performance of PNNSubPro is due to the similarity in the sequences. To answer this question, we constructed two redundance-reduced datasets by eliminating the homologous sequences from Reinhardt and Hubbard's eukaryotic dataset. Specifically, a redundance-reduced dataset should not involve any pair of sequences having an identity higher than μ, where μ is called filtering threshold. In this paper, m equals 50% and 20% for the two redudance-reduced datasets, respectively. The BLASTCLUST program in NCBI BLAST software was used to filter the homologous proteins from Reinhardt and Hubbard's eukaryotic dataset.

Table 5 shows the fivefold cross validation results of Subloc, PNNSubPro,[1] and PNNSubPro^2 on the original Reinhardt and Hubbard's eukaryotic dataset (μ = 100%) and the two redundance-reduced datasets with μ = 50% and μ = 20%, respectively.[a] When filtering threshold μ = 50%, less than a half (1137 out of 2427) of the proteins in the original dataset is remained. In this case, the overall accuracy of PNNSubPro^2 decreases from 88.5% to 78.8%, which is less significant than that of Subloc and PNNSubPro^1. The similar situation also occurs when μ further decreases to 20%. In summary, Subloc is more sensitive to homologous proteins than PNNSubPro^2 but less sensitive than PNNSubPro^1. This implies that the information from the N-terminus of the protein helps improve the robust to non-homologous proteins.

3.4. *Efficiency of PNNSubPro*

The efficiency of PNNSubPro is compared with Subloc on a PC with 2.8GHz CPU and 1GB memory. When the feature vectors have been generated, Subloc needs 183 minutes to finish the leave-one-out cross validation test while PNNSubPro^1 and PNNSubPro^2 needs 0.5 and 0.9 min, respectively. During the read-world application, however, PNNSubPro needs an additional 1–5 min to generate the PSSM of

[a]The results of Subloc, PNNSubPro,[1] and PNNSubPro^2 on the original Reinhardt and Hubbard's eukaryotic dataset are slightly different from those in Table 1. This is due to the results in Table 5 is obtained by fivefold cross validation and the results in Table 1 is obtained by leave-one-out cross validation.

each test sequence, so the actual prediction time of PNNSubPro is longer than that of Subloc. Nevertheless, we believe that the performance of a subcellular localization method is more important than its efficiency and the shortage of efficiency is easily compensated by improve the performance of computer.

3.5. *Future research*

Most existing *In-silico* subcellular localization methods (including PNNSubPro) are limited for predicting the single protein subcellular location only. As is well known, some proteins belong to multiplex subcellular locations, meaning that they can co-exist in several different location sites, or moving around among these sites. These proteins are particularly interesting and may carry some special important biological functions. Some pioneering work for predicting multiplex subcellular locations has been done recently[15] and we are attempting to extend PNNSubPro to predict proteins with multiplex subcellular locations. There are two direct ways to extend single subcellular location prediction to multiplex subcellular location prediction. The first way regards the proteins having multiplex subcellular locations as belonging to some new classes. The second way is to define a measure (e.g. likelihood) for each subcellular location and classify the protein to those subcellular locations with the measure larger than a threshold.

4. Conclusion

This paper proposed a method for eukaryotic protein subcellular localization based on protein profile, which is generated by using PSI-BLAST program to search the SWISSPROT database. The test on two benchmark datasets shows that the proposed method outperforms the methods based on the information of amino acid sequence. In addition, the prediction results on the two profile-based methods (PNNSubPro^1 and PNNSubPro^2) imply that utilizing the information of the N-terminus help improve the prediction performance and the robust to non-homologous proteins. Meanwhile, the proposed method can be easily involved in multi-predictor systems such as ESLpred or PSORT-B and can play a supplementary role to those experimental localization methods.

Acknowledgments

We would like to thank Prof. Xiangjun Liu who provided high performance workstation for our calculation. We also thank Dr. A. Reinhardt and Dr. Y. Huang who share their eukaryotic protein dataset. This work was supported by the human liver proteome project (2004BA711A21), the national nature science foundation (NSF) of China (10371063), and a grant from City University of Hong Kong (9360092).

References

1. Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ, Gapped BLAST and PSI-BLAST: A new generation of protein database search programs, *Nucleic Acids Res* **25**:3389–3402, 1997.
2. Bhasin M, Garg A, Raghava GPS, PSLpred: Prediction of subcellular localization of bacterial proteins, *Bioinformatics* **21**(10):2522–2524, 2005.
3. Bhasin M, Raghava GPS, ESLpred: SVM based method for subcellular localization of eukaryotic proteins using dipeptide composition and psi-blast, *Nucleic Acids Res* **32**(Webserver Issue):414–419, 2004.
4. Boeckmann B, Bairoch A, Apweiler R, Blatter MC, Estreicher A, Gasteiger E, Martin MJ, Michoud K, O'Donovan C, Phan I, Pilbout S, Schneider M, The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic Acids Res* **31**:365–37, 2003.
5. Cai YD, Chou KC, Predicting subcellular localization of proteins in a hybridization space, *Bioinformatics* **20**:1151–1156, 2004.
6. Cai YD, Chou KC, Predicting membrane protein type by functional domain composition and pseudo amino acid composition, *J Theor Biol* **238**:395–400, 2006.
7. Cedano J, Aloy P, Perez-Pons JA, Querol E, Relation between amino acid composition and cellular location of proteins, *J Mol Boil* **266**:594–600, 1997.
8. Chou KC, Prediction of protein subcellular locations by incorporating quasi-sequence-oder effect, *Biochem Biophys Res Commun* **278**:477–483, 2000.
9. Chou KC, Review: Prediction of protein structural classes and subcellular locations, *Curr Protein Peptide Sci* **1**:171–208, 2000.
10. Chou KC, Prediction of protein cellular attributes using pseudo amino acid composition, *PROTEINS: Structure, Function, and Genetics* **43**:246–255, 2001.
11. Chou KC, Cai YD, Using functional domain composition and support vector machines for prediction of protein subcellular location, *J Biol Chem* **277**:45765–45769, 2002.
12. Chou KC, Cai YD, A new hybrid approach to predict subcellular localization of proteins by incorporating gene ontology, *Biochem Biophys Res Commun* **311**:743–747, 2003.
13. Chou KC, Cai YD, Prediction and classification of protein subcellular location: Sequence-order effect and pseudo amino acid composition, *J Cell Biochem* **90**:1250–1260, 2003.
14. Chou KC, Cai YD, Prediction of protein subcellular locations by GO-FunD-PseAA predicor, *Biochem Biophys Res Commun* **320**:1236–1239, 2004.
15. Chou KC, Cai YD, Predicting protein localization in budding yeast, *Bioinformatics* **21**:944–950, 2005.
16. Chou KC, Cai YD, Using GO-PseAA predictor to identify membrane proteins and their types, *Biochem Biophys Res Commun* **327**:845–847, 2005.
17. Chou KC, Elord DW, Using discriminant function for prediction of subcellular location of prokaryotic proteins, *Biochem Biophys Res Commun* **252**:63–68, 1998.
18. Chou KC, Elord DW, Protein subcellular location prediction, *Protein Eng* **12**:107–118, 1999.
19. Chou KC, Shen HB, Hum-PLoc: A novel ensemble classifier for predicting human protein subcellular localization, *Biochem Biophys Res Commun* **347**:150–157, 2006.
20. Chou KC, Zhang CT, Review: Prediction of protein structural classes, *Crit Rev Biochem Mol Biol* **30**:275–349, 1995.
21. Emanuelsson O, Nielsen H, Brunak S, von Heijne G, Predicting subcellular localization of proteins based on their N-terminal amino acid sequence, *J Mol Biol* **300**:1005–1016, 1997.

22. Feng Z, Prediction of the subcellular location of prokaryotic proteins based on a new representation of the amino acid composition, *Biopolymers* **58**:491–499, 2001.
23. Feng Z, An overview on predicting the subcellular location of a protein, *In Silico Biol* **2**:291–303, 2002.
24. Gao Y, Shao SH, Xiao X, Ding YS, Huang YS, Huang ZD, Chou KC, Using pseudo amino acid composition to predict protein subcellular location: Approached with lyapunov index, bessel function, and chebyshev filter, *Amino Acids* **28**:373–376, 2005.
25. Gardy JL, Laird MR, Chen F, Rey S, Walsh CJ, Ester M, Brinkman FSL, PSORTb v.2.0: Expanded prediction of bacterial protein subcellular localization and insights gained from comparative proteome analysis, *Bioinformatics* **21**(5):617–623, 2005.
26. Gardy JL, Spencer C, Wang K, Ester M, Tusnady GE, Simon I, Hua SJ, deFays K, Lambert C, Nakai K, Brinkman FSL, PSORT-B: Improving protein subcellular localization prediction for Gram-negative bacteria, *Nucleic Acids Res* **31**(13): 3613–3617, 2003.
27. Garg A, Bhasin M, Raghava GPS, SVM-based method for subcellular localization of human proteins using amino acid compositions, their order and similarity search, *J Biol Chem* **280**:14427–14432, 2005.
28. Guo J, Lin YL, Sun ZR, A novel method for protein subcellular localization: Combining residue-couple model and SVM, *Proc APBC 2005*, pp. 117–129, 2005.
29. Guo YZ, Li M, Lu M, Wen Z, Wang K, Li G, Wu J, Classifying G protein-coupled receptors and nuclear receptors based on protein power spectrum from fast fourier transform, Amino Acids, DOI: 10.1007/S00726-006-0332-z, 2006.
30. Hirokawa T, Boon-Chieng S, Shigeki M, SOSUI: Classification and secondary structure prediction system for membrane proteins, *Bioinformatics* **14**:378–379, 1998.
31. Hua SJ, Sun ZR, Support vector machine approach for protein subcellular localization prediction, *Bioinformatics* **17**:721–728, 2001.
32. Huang Y, Li YD, Prediction of protein subcellular locations using fuzzy k-NN method, *Bioinformatics* **20**(1):21–28, 2004.
33. Lio P, Vannucci M, Wavelet change-point prediction of transmembrane proteins, *Bioinformatics* **16**:376–382, 2000.
34. Liu H, Wang M, Chou KC, Low-frequency fourier spectrum for predicting membrane protein types. *Biochem Biophys Res Commun* **336**:737–739, 2005.
35. Luo RY, Feng ZP, Liu JK, Prediction of protein strctural class by amino acid and polypeptide composition, *Eur J Biochem* **269**:4219–4225, 2002.
36. Matthews BW, Comparison of predicted and observed secondary structure of T4 phage lysozyme, *Biochim Biophys Acta* **405**:442–451, 1975.
37. Nakai K, Protein sorting signals and prediction of subcellular localization, *Adv Protein Chem* **54**(1):277–344, 2000.
38. Nakashima H, Nishikawa K, Discrimination of intracellular and extracellular proteins using amino acid composition and residue-pair frequencies, *J Mol* **238**:54–61, 1994.
39. Nielsen H, Brunak S, von Heijne G, Machine learning approaches for the prediction of signal peptides and other protein sorting signals, *Protein Eng* **12**:3–9, 1997.
40. Nielsen H, Engelbrecht J, Brunak S, von Heijne G, A neural network method for identification of prokaryotic and eukaryotic signal perptides and prediction of their cleavage sites, *Int J Neural Sys* **8**:581–599, 1997.
41. Pan YX, Zhang ZZ, Guo ZM, Feng GY, Huang ZD, He L, Application of pseudo amino acid composition for predicting protein subcellular location: Stochastic signal processing approach, *J Protein Chem* **22**:395–402, 2003.
42. Park KJ, Kanehisa M, Prediction of protein subcellular locations by support vector machines using compositions of amino acids and amino acid pairs, *Bioinformatics* **19**(13):1656–1663, 2004.

43. Parzen E, On estimation of a probability density function and mode, *Ann Math Stat* **33**:1065–1076, 1962.
44. Reinhardt A, Hubbard T, Using neural networks for prediction of the subcellular location of proteins, *Nucleic Acids Res* **26**:2230–2236, 1998.
45. Rost B, Fariselli P, Casadio R, Topology prediction for helical transmembrane proteins at 86% accuracy, *Protein Sci* **5**:1704–1718, 1996.
46. Shen HB, Chou KC, Predicting protein subnuclear location with optimized evidence-theoretic k-nearest classifier and pseudo amino acid composition, *Biochem Biophys Res Commun* **337**:752–756, 2005.
47. Shen HB, Yang J, Liu XJ, Chou KC, Using supervised fuzzy clustering to predict protein structural classes, *Biochem Biophys Res Commun* **334**:577–581, 2005.
48. Specht DF, Probabilistic neural network, *Neural Networks* **3**(1):109–118, 1990.
49. Sun XD, Huang RB, Prediction of protein structural classes using support vector machines, *Amino Acids* DOI: 10.1007/S00726-005-0239-0, 2006.
50. Wang M, Yang J, Xu ZJ, Chou KC, SLLE for predicting membrane protein types, *J Theor Biol* **232**:7–15, 2005.
51. Xiao X, Shao S, Ding Y, Huang Z, Huang Y, Chou KC, Using complexity measure factor to predict protein subcellular location, *Amino Acids* **28**:57–61, 2005.
52. Xiao X, Shao SH, Ding YS, Huang ZD, Chou KC, Using cellular automata images and pseudo amino acid composition to predict protein sub-cellular location, *Amino Acids* **30**:49–54, 2006.
53. Xiao X, Shao SH, Huang ZD, Chou KC, Using pseudo amino acid composition to predict protein structural classes: approached with complexity measure factor, *J Comput Chem* **27**:478–482,2006.
54. Yuan Z, Prediction of protein subcellular locations using markov chain models, *FEBS Letters* **451**:23–26, 1999.
55. Zhou GP, An intriguing controversy over protein structural class prediction, *J Protein Chem* **17**:729–738, 1998.
56. Zhou GP, Assa-Munt N, Some insights into protein structural class prediction, *PROTEINS: Struct Function Genet* **44**:57–59, 2001.
57. Zhou GP, Cai YD, Predicting protease types by hybridizing gene ontology and pseudo amino acid composition, *PROTEINS: Struct Function Bioinform* **63**:681–684, 2006.
58. Zhou GP, Doctor K, Subcellular location prediction of apoptosis proteins, *PROTEINS: Struct Function Genet* **50**:44–48, 2003.

Jian Guo is now the Research Assistant at Laboratory of Statistical Computation and Bioinformatics, Department of Mathematical Sciences, Tsinghua University. He received his Bachelor Degree from Tsinghua University, majoring in Pure and Applied Mathematics. His research interest includes: protein sequence analysis, computational biology and biostatistics.

Xian Pu is a research assistant at Department of Computer Science, City University of Hong Kong. She obtained her B.S. degree from Xiamen University. Her research interests include: machine learning and its application in finance and bioinformatics.

Yuanlie Lin is a Professor at the Department of Mathematical Sciences, Tsinghua University. He is the Laboratory of Statistical Computation and Bioinformatics, Department of Mathematical Sciences, Tsinghua University. His research interest includes: Markov random process, hidden Markov model, statistical learning theory and bioinformatics. He has published many papers in different fields, including theory statistics, stochastic process and bioinformatics.

Howard Leung is currently an Assistant Professor in the Department of Computer Science at City University of Hong Kong. He received the B.E. degree in Electrical Engineering from McGill University, Canada, in 1998, the M.Sc. degree and the Ph.D. degree in Electrical and Computer Engineering from Carnegie Mellon University in 1999 and 2003 respectively. Currently he is working on several research projects along the area of multimedia signal processing and pattern recognition. He has been developing a web-based Chinese handwriting education system with an intelligent analysis tool to provide instant feedback to students. Moreover, he is working on re-synthesizing the dynamic writing from static Chinese calligraphy images by applying image processing and novel model parameter estimation techniques. In addition, he is experimenting with novel approaches for processing, indexing and retrieving 3D human motions captured by Motion Capture system. In terms of professional activities, he is the Organization Chair for the 5th International Conference on Web-Based Learning (ICWL 2006) and the Finance Chair of the 10th IEEE International EDOC Conference (EDOC 2006). He is currently a member of the IEEE Signal Processing Society and is the Treasurer of the Hong Kong Web Society. More information about Howard Leung can be obtained from his homepage: http://www.cs.cityu.edu.hk/∼howard.