Using Decision Templates to Predict Subcellular Localization of Protein

Jianyu Shi[1], Shaowu Zhang[2], Quan Pan[2], and Yanning Zhang[1]

[1] School of Computer Science and Engineering,
[2] School of Automation, Northwestern Polytechnical University
710072 Xi'an, China
snake5947@msn.com, {zhangsw,quanpan,ynzhang}@nwpu.edu.cn

Abstract. Theoretical and computational methods for the prediction of protein subcellular localization have been proposed and are developing continuously. Many representations of protein sequence are proposed but a new problem arises: how to organize them together to improve prediction. It is an available solution to serialize multiple representations to single bigger one, but is still hard to avoid calculation error derived from greatly different feature values and causes huge computational burden natively because of high dimensional feature vector. We present a novel method based on decision templates(DT) for such problems in this paper. First, a protein sequence is represented as three new types of feature vectors. Then, the feature vectors are further taken as the inputs of individual SVM classifiers respectively. Finally, the outputs of these classifiers are aggregated by decision templates. The results demonstrate that DT is superior to other methods of subcellular localization prediction.

Keywords: decision templates, subcellular localization prediction, multi-scale energy, moment descriptor, amino acid composition distribution, support vector machines.

1 Introduction

As one of the most important areas in post-genome era, proteomics aims to understand proteins' potential roles, elucidate their interaction in a cellular context, and further make the corresponding functional annotation. Determination of subcellular location of proteins is of essence and importance to their functional annotation. However, the biological experiment of protein subcellular localization will be hard to meet the demands due to both time-consuming and expensive cost. Therefore, to bridge this gap, there is a need to develop more effective methods.

During the last decade, many theoretical and computational methods were developed in an attempt to predict subcellular localization of protein. Originally, Nakashima and Nishikawa represented protein sequence with amino acid composition (AAC) and indicated that intracellular and extracellular proteins are significantly different in this representation[1]. The subsequent studies showed that AAC is closely related to protein subcellular localizations[2-4]. Although AAC can represent the major

J.C. Rajapakse, B. Schmidt, and G. Volkert (Eds.): PRIB 2007, LNBI 4774, pp. 71–83, 2007.

information of sequence, it always ignores the sequence-order and structure information of protein. Hence two sequences, different in function and localization but similar in AAC, may be predicted as the same localization. To represent protein sequence better, some improved representations have been proposed[5-13].

So many representation methods give us lots of choices to predict subcellular localization of protein, but also push a new problem out: how to organize them together to achieve better prediction than what any single method can do.

One available solution is to serialize multiple representations to single bigger one, in other words, combined feature. In this way, it is proved that the prediction obtained can be better[5]. However, there are still two problems need to be solved. First, it is hard to avoid calculation error because feature values derived from multiple representations are always greatly different. Secondly, the combined feature vector is always of high dimension which bring out huge computational burden natively. Chou's pseudo amino acid composition (PseAA) provides a good way to the former problem[5,6,14]. However, the latter one is still the obstacle to build the online application of subcellular localization prediction which is already the tendency of this research area[15].

To solve above problems, some methods are presented by performing majority vote algorithm to the outputs of several classifiers which use different feature representations as inputs respectively. In this paper, we introduce three types of feature representation methods, and then use a multiple classifier fusion scheme, decision templates (DT), to perform the prediction of protein subcellular localization.

2 Database

In this paper, we use several databases which are presented in [5] and [15] respectively. These databases vary with the version of SWISS-PROT, the type of locations, the count of subcellular localization and total number of sequences. The similarities between sequences are all less than 80%.

Table 1. The database used in this paper

Database	CH01-J[5]	CH01-I[5]	HO06-P[15]
SWISS-PROT	Release 35.0	Release 35.0	Release 42.0
Location			
chloroplast(ch)	145	112	449
cytoplasm(cy)	571	761	1411
cytoskeleton(cs)	34	19	—
endoplasmic reticulum (er)	49	106	198
extracellular (ex)	224	95	843
Golgi apparatus (go)	25	4	150
lysosome (ly)	37	31	—
mitochondria (mi)	84	163	510
nucleus (nu)	272	418	837
peroxisome (pe)	27	23	157
plasma membrane (pm)	699	762	1238
vacuole (va)	24	—	63
Total	2191	2494	5856

Remarkably, the former database is composed of a training set and a testing set; the latter one contains three databases of which many locations and sequences overlap, therefore only the plant database are used here as a result of holding the most locations. Consequently, there are three databases used in these papers, which are shortly denoted by CH01-J, CH01-I, and HO06-P and listed in Table 1.

3 Representation Methods

Without loss of generality, we assume that there are N protein sequences in the dataset, let L_k be the length of the k th sequence p_k, and α_i be the i th element of 20 natural amino acids represented by English letters A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W and Y respectively.

3.1 Multi-scale Energy

According to amino acid composition, the protein sequence p_k can be characterized as a 20-D feature vector:

$$AAC_k = \left[c_1^k, \cdots, c_i^k, \cdots, c_{20}^k \right], \quad k = 1, \cdots, N, \tag{1}$$

where $c_i^k = n^i / L_k$ is the normalized occurrence frequency of amino acid α_i, and n^i is the count of α_i appearing in sequence p_k.

However, it is not sufficient to represent a specific protein sequence only based on AAC. Consequently, there is a need to improve AAC or develop other representations of protein sequence to deal with such case.

Using discrete wavelet transform (DWT) and Mallat fast algorithm[16], we proposed the multi-scale energy (MSE) representation to improve AAC[8]. Each protein sequence can be firstly coded into digital signal by mapping all amino acid residues of protein sequence to the corresponding numerical value according to one of amino acid indices[17]. Here we choose the hydrophilicity index HOPT810101 from amino acid index database. Hence, such a coded protein sequence can be treated as a digital signal and further processed by DWT.

According to DWT and Mallat fast algorithm, the fine-scale and large-scale information of a protein hydrophilicity signal can be simultaneously investigated by projecting the mapped digital signal onto a set of wavelet basis functions with various scales. Here, the wavelet basis function used is symlet wavelet. The features extracted from the wavelet-based multi-resolution information, can discriminate different types of protein signals. Consequently, sequence p_k can be characterized as a (m+1)-D feature vector of multi-scale energy(MSE):

$$MSE_k = \left[d_1^k, \cdots, d_j^k, \cdots, d_m^k, a_m^k \right] . \tag{2}$$

Here m is the coarsest scale of decomposition, d_j^k is the root mean square energy of the wavelet detail coefficients in the corresponding j th scale, and a_m^k is the root

mean square energy of the wavelet approximation coefficients in the scale m. The energy factors d_j^k and a_m^k are defined as

$$d_j^k = \sqrt{\frac{1}{N_j}\sum_{n=0}^{N_j-1}[u_j^k(n)]^2}\ ,\ a_m^k = \sqrt{\frac{1}{N_m}\sum_{n=0}^{N_m-1}[v_m^k(n)]^2}\ ,\ j=1,2,\cdots,m\,, \tag{3}$$

where N_j is the number of the wavelet detail coefficients, N_m is the number of the wavelet approximation coefficients, $u_j^k(n)$ is the nth detail coefficient in the corresponding jth scale, and $v_m^k(n)$ is the nth approximation coefficient in the scale m. For the protein sequence p_k with length L_k, m equals $INT\left(\log_2\left(L_k\right)\right)$.

Obviously, MSE contains the approximation and detail information of protein signal which reflect sequence-order effects. In order to get better representation, we combine MSE with AAC and construct the following (20+m+1)-D feature vector $\vec{x}_k$ to represent sequence p_k.

$$\vec{x}_k = [c_1^k,\cdots,c_i^k,\cdots,c_{20}^k,\lambda_1^k,\cdots,\lambda_j^k,\cdots,\lambda_m^k,\lambda_{m+1}^k]^T\,, \tag{4}$$

where $\lambda_j^k = d_j^k, \lambda_{m+1}^k = a_m^k, j=1,\cdots,m$.

3.2 Moment Descriptor

Considering the order amino acid, we proposed a new feature representation method, called moment descriptor (MD)[12].

Firstly, instead of using the direct definition of AAC in (1), we calculate c_i^k by introducing position indicator $\Delta_{i,j}^k$ as follows:

$$c_i^k = \frac{1}{L_k}\sum_{j=1}^{L_k}\Delta_{i,j}^k\,, \tag{5}$$

$$\Delta_{i,j}^k = \begin{cases} 1 & \text{if } \alpha_i \text{ is present at position } j \text{ in } p_k \\ 0 & \text{if } \alpha_i \text{ is NOT present at position } j \text{ in } p_k \end{cases}. \tag{6}$$

Obviously, (5) is the sampled statistical mean (raw moment) of position indicator. Hence, we choose it as the first MD of protein sequence.

Secondly, considering the position of amino acid α_i in sequence p_k, we define another feature for amino acid α_i:

$$m_i^k = \frac{1}{L_k}\sum_{j=1}^{L_k}\left(\Delta_{i,j}^k \cdot j\right). \tag{7}$$

where m_i^k represents mean of position of α_i. We choose it as the second MD.

Thirdly, the sampled variance v_i^k of position of amino acid α_i in sequence p_k is considered:

$$v_i^k = \frac{1}{L_k}\sum_{j=1}^{L_k}\left(\Delta_{i,j}^k \cdot j - m_i^k\right)^2 . \tag{8}$$

where v_i^k represents the second-order central moment of position of amino acid α_i in sequence p_k. We choose it as the third MD of protein sequence.

Eventually, we get a combined feature vector for sequence p_k by serializing above three moment descriptors:

$$\bar{x}_k = \left[c_1^k,\cdots,c_i^k,\cdots,c_{20}^k, m_1^k,\cdots,m_i^k,\cdots,m_{20}^k, v_1^k,\cdots,v_i^k,\cdots,v_{20}^k\right]^T, k = 1,\cdots,N . \tag{9}$$

3.3 Amino Acid Composition Distribution

As we know, the tertiary structure of protein is always composed of several secondary structure units, such as α-helix or β-sheet. Considering this fact, we present a new representation, amino acid composition distribution (AACD) which divide a protein sequence p_k equally into multiple segments and then calculate AAC of each segment in series[13]. So the sequence p_k can be represented as the following formula:

$$AACD_n^k = \begin{Bmatrix} c_{1,1}^k & \cdots & c_{1,m}^k & \cdots & c_{1,n}^k \\ \cdots & \cdots & c_{i,m}^k & \cdots & \cdots \\ c_{20,1}^k & \cdots & c_{20,m}^k & \cdots & c_{20,n}^k \end{Bmatrix}_{20\times n} , \tag{10}$$

where n is the count of segments, $\left[c_{1,m}^k,\cdots,c_{i,m}^k,\cdots c_{20,m}^k\right]^T$ is the AAC of the m th segment of p_k, and $c_{i,m}^k$ is define as:

$$c_{i,m}^k = n \cdot t_{i,m}^k \big/ L_k , \quad m = 1,\cdots,n, \quad i = 1,\cdots,20 , \tag{11}$$

where $t_{i,m}^k$ is the count of α_i appearing in the m th segment of sequence p_k.

In order to be the input of classifier, this representation of protein sequence p_k is turned to a feature vector as the following:

$$\bar{x}_k = \left[c_{1,1}^k,\cdots,c_{1,n}^k,\cdots,c_{i,1}^k,\cdots,c_{i,n}^k,\cdots,c_{20,1}^k,\cdots,c_{20,n}^k\right]^T , \quad k = 1,\cdots,N . \tag{12}$$

4 Classification and Assessment

4.1 Support Vector Machines

When the representation of protein sequence is set, next step is just to choose a classifier to perform the prediction of subcellular localization. Many types of classifiers which have been applied to such prediction, such as neural network[2], covariant discriminant algorithm[5], fuzzy KNN[18] and support vector machines(SVM)[9,10,15].

In these classifiers, SVM has been more broadly applied to such prediction due to its good performance of classification. SVM was originally designed for binary classification[19] while such prediction is M-class classification. Usually, we can construct M-class SVMs to solve such problem based on the binary class SVM. That is an ongoing research issue. Extensive experiments have shown that ''One-Versus-Rest'' (OVR)[19], ''One-Versus-One'' (OVO)[20]and ''Directed Acyclic Graph'' (DAG)[21]are practical[8,12,22,23]. Because of its convenient usage, OVO is used in this paper.

To perform the prediction, the SVM software, LIBSVM, is used, and can be freely downloaded from http://www.csie.ntu.edu.tw/~cjlin/libsvm/ for academic research[22]. In addition, we do the training only with the RBF kernel in all experiments.

4.2 Multiple Classifier System and Decision Template

If we have different feature sets, different training sets, different classification methods or different training sessions, then multiple classifier system (MCS) is proposed to improve classification or prediction accuracy by combining the outputs of a set of classifiers[24,25]. Various combined schemes for MCS can be grouped into three basic main categories according to their architecture: serial (cascading), hierarchical and parallel architectures[25]. Most combination schemes in the literatures belong to the parallel MCS which involves two kinds of aggregated schemes. The one is known as selection rule[26] of which clustering and selection, dynamic classifier selection with local accuracy are always used. The other is referred to as fusion rule which includes lots of related algorithms, for example, majority vote[24].

If the set of classifiers is fixed, the problem focuses on the aggregated function or rule. It is also possible to use a fixed combiner and optimize the set of input classifiers. We only consider the former one in this paper. Due to the higher efficiency and flexibility, we select parallel MCS and use the fusion rule decision templates (DT). DT is non-sensitive to poorly trained individual classifiers, and can achieve good and stable performance without strict probabilistic conditions[27].

Let $\bar{x} \in \Re^n$ be a feature vector (a representation of a protein sequence), $\{\omega_1,...,\omega_j,...,\omega_M\}$ be the label set of M classes, and $\{e_1,...,e_i,...,e_L\}$ be the set of L classifiers. We denote the output of the i-th classifier as $D_i(\bar{x}) = \left[d_{i,1}(\bar{x}),...,d_{i,j}(\bar{x}),...,d_{i,M}(\bar{x})\right]^T$, where $d_{i,j}(\bar{x})$ is the degree of "support" given by classifier e_i to the hypothesis that $\bar{x}$ comes from class ω_j. The outputs of L classifiers can be organized in a decision profile (DP) as the matrix [27]:

$$DP(\vec{x}) = \begin{bmatrix} d_{1,1}(\vec{x}) & \cdots & d_{1,j}(\vec{x}) & \cdots & d_{1,M}(\vec{x}) \\ & & \vdots & & \\ d_{i,1}(\vec{x}) & \cdots & d_{i,j}(\vec{x}) & \cdots & d_{i,M}(\vec{x}) \\ & & \vdots & & \\ d_{L,1}(\vec{x}) & \cdots & d_{L,j}(\vec{x}) & \cdots & d_{L,M}(\vec{x}) \end{bmatrix}, \tag{13}$$

where the components $d_{i,j}(\vec{x})$ can be regarded as an estimate of the posterior probability $P_i(\omega_j \mid \vec{x})$ produced by classifier e_i for class ω_j and the given $\vec{x}$, $i = 1, 2, ..., L$, $j = 1, 2, ..., M$.

Let Z be the crisp labeled training dataset. The decision template for class ω_j denoted DT_j can be regarded as the expected $DP(\vec{x})$ for class ω_j:

$$DT_j = \frac{1}{N_j} \sum_{\substack{\vec{z}_k \in \omega_j \\ \vec{z}_k \in Z}} DP(\vec{z}_k), j = 1, \cdots M. \tag{14}$$

where $\vec{z}_k \in \Re^n$ is a feature vector of the training dataset, N_j is the number of samples of Z from class ω_j, and DT_j is an $L \times M$ matrix.

For a tested feature vector $\vec{x} \in \Re^n$, we can calculate the squared Euclidean distance between $DP(\vec{x})$ and each DT_j

$$d_E\left(DP(\vec{x}), DT_j\right) = \frac{1}{L \cdot c} \sum_{i=1}^{M} \sum_{k=1}^{L} \left(d_{k,i}(\vec{x}) - dt_j(k,i)\right)^2, \tag{15}$$

where $dt_j(k,i)$ is the k, i-th element in decision template DT_j.

Then, we can get the find predicted class label of $\vec{x}$ by:

$$j^*(\vec{x}) = \arg\max_j \left(1 - d_E\left(DP(\vec{x}), DT_j\right)\right), j = 1, \cdots, M. \tag{16}$$

4.3 Prediction of Assessment

To make a fair and full comparison, jackknife test are used to CH01-J and independent test to CH01-I and 5-fold cross validation (5CV) to HO06-P respectively.

During the process of jackknife test, each protein in training dataset is singled out in turn as a test sample, and the remaining are used as training samples. For independent test, proteins in training dataset are used as training samples, and those in independent test dataset are used as test samples. The quality of independent test indicates the ability of generalization of prediction system. To assess the quality of

jackknife and independent test, the total prediction accuracy is always used and defined as:

$$Q = \frac{1}{N}\sum_{\omega=1}^{\Omega} p(\omega)\,. \tag{17}$$

According to 5CV procedure, the dataset is split randomly and equally into 5 subsets. In turn, we take each subset as the testing set to evaluate the prediction, and use the rest subsets to build classification modal, in other words, to do the training. The average and the standard deviation of the accuracies of all evaluations are used to indicate the performance of prediction, and are defined respectively as:

$$\bar{Q} = \sum_{i=1}^{k} Q_i \Big/ k\,,\ S = \sqrt{\sum_{i=1}^{k}\left(Q_i - \bar{Q}\right)^2 \Big/ (k-1)},\ i = 1,\cdots,k,\quad k = 5\,. \tag{18}$$

where Q_i is the accuracy of the i th evaluation and k is the count of cross-validation.

Besides the total accuracy or average accuracy, the sensitivity ($Sens^i$), the specificity ($Spec^i$) and the Matthews correlation coefficient (MCC^i) of each location are also used to assess the wider performance of prediction:

$$\begin{aligned} Sens^i &= \frac{p_i}{(p_i + u_i)}\,,\ Spec^i = \frac{p_i}{(p_i + o_i)}\,, \\ MCC^i &= \frac{p_i n_i - u_i o_i}{\sqrt{(p_i + u_i)(p_i + o_i)(n_i + u_i)(n_i + o_i)}} \end{aligned} \tag{19}$$

5 Experiment and Discussion

Each of protein sequences is firstly represented as three types of feature vectors which are corresponding to MSE, MD, and AACD respectively. Then, the proposed feature representations are taken as the inputs of three multi-class SVM classifiers respectively. Finally, the prediction is performed by fusing the outputs of these individual SVM classifiers with decision templates rule.

5.1 Comparison with the Former Methods

In order to validate the effectiveness of decision templates and make fair comparisons, we perform DT on all selected databases, use jackknife test to CH01-J, independent test to CH01-I, and 5CV test to HO06-P respectively. The results are shown in Table 2 and Table 3 respectively.

We can see that DT wins the best accuracies and achieves the lowest standard deviations. To dataset CH01-J, the accuracy achieved by DT is 83.75%, and 10.72%, 16.07% and 10.18% higher than those achieved by [5,6,14], respectively. To dataset CH01-I, the accuracy achieved by DT is 88.41%, and 7.54%, 14.55% and 8.62% higher than those achieved by [5,6,14], respectively. To dataset HO06-P, the accuracy

Table 2. The results of the comparison with the former methods for database Chou

Methods	CH01-J	CH01-I
	Jackknife(%)	Independent(%)
Chou[5]	73.03	80.87
Pan[6]	67.68	73.86
Xiao[14]	73.57	79.79
Our(DT)	83.75	88.41

Table 3. The results of the comparison with the former methods for database HO06-P by 5CV

Loc	PORST[28]			MultiLoc[15]			DT		
	Sens	Spec	MCC	Sens	Spec	MCC	Sens	Spec	MCC
ch	0.49	0.58	0.50	0.88	0.85	0.85	0.85	0.89	0.86
cy	0.40	0.70	0.42	0.68	0.85	0.70	0.87	0.75	0.73
er	0.21	0.11	0.11	0.72	0.54	0.61	0.61	0.86	0.71
ex	0.74	0.70	0.67	0.68	0.81	0.70	0.86	0.86	0.83
go	0.02	0.13	0.04	0.75	0.41	0.54	0.71	0.82	0.76
mi	0.65	0.53	0.54	0.85	0.81	0.81	0.74	0.77	0.73
nu	0.59	0.60	0.53	0.82	0.75	0.75	0.80	0.82	0.78
pe	0.47	0.16	0.24	0.71	0.34	0.47	0.39	0.79	0.55
pm	0.81	0.75	0.72	0.74	0.89	0.77	0.93	0.90	0.89
va	0.13	0.06	0.07	0.70	0.20	0.36	0.48	0.75	0.59
Q(%)	57.50			74.60 ± 0.80			82.68 ± 0.38		

achieved by DT is 82.68%, and 25.18% and 8.08% higher than those achieved by [28] and [15], respectively. In addition, DT achieves the lowest standard deviation.

5.2 The Performance Analysis of DT

In order to analyze the performance of DT, we make a contrast with the "single best" classifier and the "oracle". The "single best" classifier is referred to as the member classifier which achieves the best accuracy. The "oracle" is such procedure which assign the correct class label to $\vec{x}$ as long as one individual classifier produces the correct class label of $\vec{x}$ [27]. The result of "oracle" can reflect the maximum bound of the classification ability of MCS. The results of Chou-J, Chou-I and HO06-P are listed in Table 4, 5, and 6 respectively.

Firstly, these results show that DT fusion rule can achieve better total accuracy and lower standard deviation than the "single best" classifier. Secondly, the sensitivity, the specificity and MCC of each location are always improved in most cases. Especially, the performances of prediction on those locations which have minority protein sequences are improved greatly. Finally, the results of "oracle" show that the fusion of MSE, MD and AACD can represent protein sequence better than any single feature representation because the maximum bound of the classification ability of MCS for three dataset are 89.14%, 93.42 and 90.86%, respectively.

Furthermore, we also compare DT with other popular aggregated rules which are majority vote (MV) [24] and dynamic classifier selection with local accuracy (DCS_LA) [26]. The results are shown in Table 7.

Table 4. The result of multi-classifier fusion for the Chou-J dataset

Loc	Single Best			Decision Template			Oracle
	Sens	Spec	MCC	Sens	Spec	MCC	Sens
ch	0.79	0.82	0.79	0.75	0.88	0.80	0.86
cy	0.91	0.76	0.76	0.93	0.77	0.78	0.97
cs	0.44	0.94	0.64	0.47	1.00	0.68	0.56
er	0.39	0.83	0.56	0.41	0.95	0.62	0.55
ex	0.76	0.77	0.74	0.75	0.80	0.74	0.86
go	0.20	0.83	0.40	0.28	0.78	0.46	0.40
ly	0.54	0.80	0.65	0.59	0.73	0.65	0.76
mi	0.32	0.71	0.46	0.43	0.82	0.58	0.48
nu	0.88	0.72	0.76	0.88	0.74	0.78	0.93
pe	0.26	1.00	0.51	0.26	0.44	0.33	0.26
pm	0.94	0.94	0.91	0.96	0.96	0.94	0.98
va	0.29	1.00	0.54	0.29	0.88	0.50	0.46
Q(%)	82.15			83.75			89.14

Table 5. The result of multi-classifier fusion for the Chou-I dataset

Loc	Single Best			Decision Template			Oracle
	Sens	Spec	MCC	Sens	Spec	MCC	Sens
ch	0.67	0.78	0.71	0.74	0.81	0.77	0.86
cy	0.91	0.87	0.84	0.93	0.91	0.89	0.96
cs	1.00	0.95	0.97	1.00	0.95	0.97	1.00
er	0.89	0.96	0.92	0.93	0.98	0.95	0.94
ex	0.88	0.74	0.80	0.86	0.81	0.83	1.00
go	0.50	1.00	0.71	0.50	1.00	0.71	0.50
ly	0.94	0.97	0.95	1.00	0.97	0.98	1.00
mi	0.18	0.85	0.37	0.29	0.89	0.49	0.53
nu	0.84	0.85	0.82	0.87	0.87	0.84	0.94
pe	0.39	1.00	0.62	0.61	0.78	0.69	0.65
pm	0.98	0.84	0.86	0.99	0.87	0.89	0.99
va	-	-	-	-	-	-	-
Q(%)	85.49			88.41			93.42

Table 6. The result of multi-classifier fusion for the HO06-P dataset

Loc	Single Best			Decision Template			Oracle
	Sens	Spec	MCC	Sens	Spec	MCC	Sens
ch	0.84	0.83	0.82	0.85	0.89	0.86	0.90
cy	0.81	0.71	0.66	0.87	0.75	0.73	0.94
er	0.55	0.81	0.65	0.61	0.86	0.71	0.69
ex	0.84	0.79	0.77	0.86	0.86	0.83	0.95
go	0.57	0.83	0.68	0.71	0.82	0.76	0.76
mi	0.72	0.75	0.71	0.74	0.77	0.73	0.93
nu	0.75	0.79	0.72	0.80	0.82	0.78	0.89
pe	0.32	0.85	0.51	0.39	0.79	0.55	0.45
pm	0.87	0.84	0.81	0.93	0.90	0.89	0.97
va	0.40	0.81	0.56	0.48	0.75	0.59	0.48
Q(%)	78.09 ± 1.01			82.68 ± 0.38			90.86

Table 7. The results of the comparison with other aggregated rules

Methods	CH01-J	CH01-I	HO06-P
	Jackknife(%)	Independent(%)	5CV(%)
MV	83.02	88.05	80.28 ± 0.53
DCS_LA (k=10)	83.20	88.01	80.67 ± 0.49
DT	83.75	88.41	82.68 ± 0.38

These results also show that DT fusion rule can achieve better total accuracy and lower standard deviation than both MV and DCS_LA rules.

As above described, DT is an effective and robust method to subcellular localization prediction.

6 Conclusion

In this paper, we have presented three types of representation methods, MSE, MD and AACD, and then performed prediction of protein subcellular localization by using the parallel MCS. Instead of serializing the proposed representations of protein sequence to single bigger one, the presented method integrates them together by DT fusion rule. DT aggregates the outputs of three individual SVM classifiers which take those representations as the inputs respectively. Compared with other prediction methods, other aggregated rules and the single best classifier, the results show that DT achieves better prediction of subcellular localization and is more effective and robust to subcellular localization prediction. In addition, DT also avoids huge computational burden increased by high dimension derived from the serialization of multiple representations. Consequently, DT can be applied to develop the application of subcellular localization prediction.

Acknowledgments. The authors would like to thank Prof. Guo-Ping Zhou for his critical and constructive comments and suggestions. This paper was supported in part by the National Natural Science Foundation of China (No. 60372085 and 60634030) and the Technological Innovation Foundation of Northwestern Polytechnical University (No. KC02).

References

1. Nakashima, H., Nishikawa, K.: Discrimination of Intracellular and Extracellular Proteins Using Amino Acid Composition and Residue-Pair Frequencies. J. Mol. Biol. 238, 54–61 (1994)
2. Reinhardt, A., Hubbard, T.: Using Neural Networks for Prediction of the Subcellular Localization of Proteins. Nucleic Acids Research 26, 2230–2236 (1998)
3. Chou, K.C., Elrod, D.: Protein Subcellular Localization Prediction. Protein Eng. 12, 107–118 (1999)
4. Hua, S.J., Sun, Z.R.: Support Vector Machine Approach for Protein Subcellular Localization Prediction. Bioinformatics 17, 721–728 (2001)
5. Chou, K.C.: Prediction of Protein Cellular Attributes Using Pseudo-Amino Acid Composition. Proteins: Struct. Funct. Genet. 43, 246–255 (2001)

6. Pan, Y.X., Zhang, Z.Z., Guo, Z.M., Feng, G.Y., Huang, Z., He, L.: Application of Pseudo Amino Acid Composition for Predicting Protein Subcellular Location: Stochastic Signal Processing Approach. Journal of Protein Chemistry 22, 395–402 (2003)
7. Gao, Y., Shao, S.H., Xiao, X., Ding, Y.S., Huang, Y.S., Huang, Z.D., Chou, K.C.: 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)
8. Shi, J.Y., Zhang, S.W., Pan, Q., Cheng, Y.M., Xie, J.: Prediction of Protein Subcellular Localization by Support Vector Machines Using Multi-Scale Energy and Pseudo Amino Acid Composition. Amino Acids 33, 69–74 (2007)
9. Park, K.J., Kanehisa, M.: Prediction of Protein Subcellular Locations by Support Vector Machines Using Compositions of Amino Acids and Amino Acid Pairs. Bioinformatics 19, 1656–1663 (2003)
10. Cui, Q., Jiang, T., Liu, B., Ma, S.: Esub8: A Novel Tool to Predict Protein Subcellular Localizations in Eukaryotic Organisms. BMC Bioinformatics 5, 66–72 (2004)
11. Bhasin, M., Raghava, G.P.S.: Eslpred: SVM-Based Method for Subcellular Localization of Eukaryotic Proteins Using Dipeptide Composition and Psi-Blast. Nucl. Acids Res. 32, W414–W419 (2004)
12. Shi, J.Y., Zhang, S.W., Liang, Y., Pan, Q.: Prediction of Protein Subcellular Localizations Using Moment Descriptors and Support Vector Machine. In: Rajapakse, J.C., Wong, L., Acharya, R. (eds.) PRIB 2006. LNCS (LNBI), vol. 4146, pp. 105–114. Springer, Heidelberg (2006)
13. Shi, J.Y., Zhang, S.W., Pan, Q., Zhou, G.-P.: Amino Acid Composition Distribution: A Novel Sequence Representation for Prediction of Protein Subcellular Localization. In: The 1st IEEE International Conference on Bioinformatics and Biomedical Engineering, pp. 115–118. IEEE Computer Society Press, Los Alamitos (2007)
14. Xiao, X., Shao, S.H., Ding, Y.S., Huang, Z.D., Huang, Y., Chou, K.C.: Using Complexity Measure Factor to Predict Protein Subcellular Location. Amino Acids 28, 57–61 (2005)
15. Höglund, A., Dönnes, P., Blum, T., Adolph, H.-W., Kohlbacher, O.: Multiloc: Prediction of Protein Subcellular Localization Using N-Terminal Targeting Sequences, Sequence Motifs and Amino Acid Composition. Bioinformatics 22, 1158–1165 (2006)
16. Mallat, S.: A Wavelet Tour of Signal Processing, 2nd edn. Academic Press, London (1999)
17. Kawashima, S., Ogata, H., Kanehisa, M.: AAindex: Amino Acid Index Database. Nucleic Acids Research 27, 368–369 (1999)
18. Huang, Y., Li, Y.D.: Prediction of Protein Subcellular Locations Using Fuzzy K-NN Method. Bioinformatics 20, 21–28 (2004)
19. Vapnik, V.: Statistical Learning Theory. Wiley, New York (1998)
20. Kreßel, U.H.: Pairwise Classification and Support Vector Machines. In: Schölkopf, B., Burges, C.J., Smola, A.J. (eds.) Advances in Kernel Methods: Support Vector Learning, pp. 255–268. MIT Press, Cambridge, MA (1999)
21. Platt, J., Cristianini, N., Shawe-Taylor, J.: Large Margin Dags for Multiclass Classification. Advances in Neural Information Processing Systems 12, 547–553 (2000)
22. Hsu, C., Lin, C.J.: A Comparison of Methods for Multi-Class Support Vector Machines. IEEE Transactions on Neural Networks 13, 415–425 (2002)
23. Rifin, R., Klautau, A.: In Defense of One-Vs-All Classification. Journal of Machine Learning Research 5, 101–141 (2004)
24. Kittler, J., Hatef, M., Duin, R., Matas, J.: On Combining Classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence 20, 226–239 (1998)

25. Jain, A.K., Duin, R.P.W., Mao, J.: Statistical Pattern Recognition: A Review. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 4–37 (2000)
26. Kuncheva, L.I.: Switching between Selection and Fusion in Combining Classifiers: An Experiment. IEEE Transactions on Systems, Man, and Cybernetics, Part B 32, 146–156 (2002)
27. Kuncheva, L.I., Bezdek, J.C., Duin, R.: Decision Templates for Multiple Classifier Fusion: An Experimental Comparison. Pattern Recognition 34, 299–314 (2001)
28. Nakai, K., Horton, P.: Psort: A Program for Detecting the Sorting Signals of Proteins and Predicting Their Subcellular Localization. Trends Biochem. Sci. 24, 34–36 (1999)