A Comparative Study on Feature Extraction from Protein Sequences for Subcellular Localization Prediction

Wen-Yun Yang [1], Bao-Liang Lu [1,2], and Yang Yang [1]
[1] Department of Computer Science and Engineering, Shanghai Jiao Tong University
[2] Laboratory for Computational Biology, Shanghai Center for Systems Biomedicine
800 Dong Chuan Rd., Shanghai 200240, China
{bluesky; bllu; alayman}@sjtu.edu.cn

***Abstract*— One of the central problems in computational biology is to identify the protein function in an automated and high-throughput fashion. A key step in this process is to predict subcellular compartment the protein belongs to, since the protein localization closely correlates with its function. A wide variety of methods for protein subcellular localization has been proposed over recent years. They fall into two categories, sequence-based and database-based. The first one is to extract useful features from amino acid sequences and strives to discover the principles behind protein localization process. The second one is more apt to conduct data mining from existing public annotation databases.**

This paper focuses on the sequence-based approach and exploits the discriminative ability contained in amino acid sequences for protein subcellular localization. We conducted comparisons among amino acid composition approach, amino acid tuple approach, voting scheme, and a new characteristic representation of proteins proposed in this paper. Our experiments are carried out on 7579 eukaryotic protein sequences from 12 subcellular locations. The highest accuracy, 82.8% across 5-fold cross validation is obtained by voting scheme using five predictors. This is the best performance achieved in this dataset using sequence-based approach. Our experiments demonstrate that there are considerable potentials on improving prediction accuracy by exploiting protein sequences, which have not been fully utilized so far, and more explorations are still needed in this direction.

I. Introduction

One of the fundamental goals in cell biology and proteomics is to identify the function of new proteins. Since experimental determination is expensive and time consuming, as well as the amount of internet available protein sequences are exploding dramatically, computational methods aiming to predict protein function in an automated and high-throughput fashion are increasingly becoming an appealing complement to experimental techniques.

Protein subcellular localization closely relates to the protein function. Therefore, predicting the subcellular location of protein sequences is a key step to understand the biological functions of protein sequences. Various methods for predicting subcellular localization of protein sequences have been extensively studied in the last decades, and researchers have developed increasingly more new models to acquire better prediction performance. Typically, the development in this area follows two trends: sequence-based and database annotation-based.

The sequence-based discriminative prediction attempts to extract increasingly more characteristic subsequence features from protein sequences and performs prediction based on these features. The methods in sequence-based trend can be further divided into three sub-categories: (a) prediction based on amino acid composition, (b) prediction with known targeting sequences, and (c) prediction based on other novel extracted features.

The pioneering work on amino acid composition discriminative capability was done by Nishikawa [1], Reinhardt and Hubbard [2]. Then extensive studies upon it have been conducted [3] [4] [5] [6]. Furthermore, an extensional exploration on amino acid pair composition and voting scheme was proposed by Park and Kanehisa [7]. The underlying biological model in this category is fairly simple since it only depends on 20 amino acid features. Although the discriminative ability is limited by the lack of features, it is nonetheless still a good choice when very little annotation information is known about the query protein sequence.

Another way to predict subcellular localization of protein based on sequences is to identify targeting sequences [8] [9] [10] [11]. These methods strive towards mimicking the biological protein sorting process with computational simulation. However, these methods based on targeting sequences inevitably have a common drawback: it is hard to determine the presence of a targeting sequences. As protein sequences from draft genomes are often incomplete, lacking N-terminal region, the prediction methods in this category will be inaccurate when the targeting signals are missing or partially included. On the other hand, these methods based on targeting sequences such as [8] [11] generally predict three or four locations, and have relatively high prediction accuracies. In practical application sense, the coverage scope is fairly narrow.

Besides amino acid composition and targeting sequence, many other novel methods for feature extraction from protein sequences have been developed. The spectrum kernel and mismatch kernel by Leslie [12] [13], and weight decomposition kernel by Menchetti *et al.* [14], take the context of each amino

acid residue into account. Yang and Lu developed a method, in which Chinese language segmentation techniques are used to extract features from protein sequences [15]. Toh and his colleagues adopted N-terminal sorting signals by using the information derived from amino acid index database [16].

The second trend of prediction of protein subcellular location arises from the protein annotation databases. This kind of method depends on the fact that the annotation databases are becoming increasingly more capable to supply reliable clues for protein identity or homology analysis, such as motifs, gene ontology (GO) [17] and function domain of proteins. Chou and his colleagues [18] [19] [20] proposed a Go-FunD-PseAA hybridized method, which performs better than others. In fact, the prior domain knowledge derived from database query involves abundant information relating to protein profiles of subcellular localization and can be used to further improve the prediction accuracy by annotation matching. However, this kind of methods faces a deficiency. When the protein to be predicted is a newly discovered one, there is no existing annotation in the database. As a result, the prediction performance of this kind of methods will be degraded. However, along with the growing coverage of public annotation databases, further improvement can be expected. Recently, Höglund and colleagues [21] [22] [23] proposed a hybrid method which combines targeting sequences, amino acid composition, and sequence motifs for predicting protein subcellular localization. Their predicting model is an integrated system of four or five different classifiers, SVMTarget, SVMSA, SVMaac, MotifSearch and/or text-based method, some of which partially depend on specific databases such as PROSITE and NLSdb.

In this paper, we carry out in-depth exploration on the discriminative ability of different sequence-based methods using amino acid composition and amino acid k-tuples. Furthermore, we apply the feature selection methods derived from text classification to amino acid sequences to select the subsequences with the most statistical characteristic. And then we propose a new hybrid method for incorporating characteristic tuples into amino acid features to improve the prediction performance.

This paper is organized as follows. Section 2 describes several existing methods used in our comparative study and our new prediction method. Section 3 presents the experiments and the simulation results. Section 4 discusses the benefit and cost of these methods. Section 5 summarizes the conclusions.

II. Methods

Protein sequences are consecutive amino acid residues, and we regard them as text strings with an alphabet $\mathcal{A}$ of size $|\mathcal{A}| = 20$. Many feature extraction methods have been developed in the past several years. Typically, these methods can be classified into two categories. One is based solely on amino acid composition [1] [2]. The other one is an extension of the atomic length from only one amino acid to k amino acid tuple, where k is an integer and larger than one. We refer to it as 'k-tuple', such as 2-tuple in [7].

The rest of this section consists of three parts. Firstly, 20 amino acid features are adopted as our initial representative features, as it is based on the assumption which is simple and effective. Secondly, k-tuple features are introduced to take the place of 20 amino acid as feature set. Against the high computational cost of k-tuple prediction caused by large feature amount, three feature selection methods which are commonly used for text classification are introduced to eliminate irrelevant k-tuples. Thirdly, we propose a hybrid prediction method with all r (for $r = 1, \ldots, k$) length amino acid tuples together as a mixed feature set. The experiment shows that our hybrid method appears to be an alternative choice.

A. Amino Acid Composition

In amino acid composition prediction model, each protein sequence i in the dataset of size N is represented by an input vector $\vec{x}_i$ of 20 dimensions and a location label y_i for $i = 1, \ldots, N$. The prediction procedure can be understood within a 20 dimensional space and each protein sequence represents a point in it. What we need to do is to classify the points to their corresponding labels.

Intuitively, we consider amino acid composition (AAC-I in short) to be amino acid residue occurrence times.

$$\begin{gathered} x_{ij} = count_i(j) \\ \text{for } i = 1, \ldots, N \text{ and } j = 1, \ldots, 20 \end{gathered} \tag{1}$$

where x_{ij} is the jth element of $\vec{x}_i$, and $count_i(j)$ denotes the times that amino acid j occurs in protein sequence i.

For normalization purpose, the following equation is always satisfied for any protein sequence whether it is longer or shorter than others.

$$\sum_{j=1}^{20} x_{ij} = 1. \tag{2}$$

So we consider amino acid composition to be amino acid residue occurrence probability (AAC-II in short)

$$x_{ij} = \frac{count_i(j)}{\sum_{j=1}^{20} count_i(j)}. \tag{3}$$

It is reported that better performance could be obtained by normalizing each $\vec{x}_i$ to $\vec{a}_i$ [6], where $|\vec{a}_i| = 1$ for $i = 1, \cdots, N$. So each $\vec{a}_i$ will be the unit length vector in 20 dimensional Euclidean space. The following relation (AAC-III in short) between $\vec{x}_i$ and $\vec{a}_i$ can be easily proven

$$\begin{gathered} a_{ij} = \sqrt{x_{ij}} \\ \text{for } i = 1, \ldots, N \text{ and } j = 1, \ldots, 20. \end{gathered} \tag{4}$$

B. k-tuple Subsequence

It should be pointed out that all of the prediction algorithms based on amino acid composition do not take the sequence order effect into account. To improve the prediction accuracy, it is necessary to incorporate some order information. Intuitively, we use amino acid tuples to partially represent the sequence order. For example, the sequences "AIC" and "CIA" have the same representation by the 20 amino acid features. But if we use 2-tuple features, "AIC" is represented by "AI" and "IC", and "CIA" is represented by "CI" and "IA".

Since the experimental results show that ACC-I defined in (1) give the best performance (see Table II), we accordingly modify it to be the k-tuple feature vector for each protein sequence as follows. The length of k-tuple feature vector would be 20^k and each element is the corresponding k-tuple occurrence time, i.e.,

$$\begin{aligned} x_{ij} &= count_i(j) \\ \text{for } i &= 1, \ldots, N \text{ and } j = 1, \ldots, 20^k \end{aligned} \tag{5}$$

where $count_i(j)$ denotes the times that jth amino acid tuple occurs in protein sequence i.

It should be noted that the dimensionality of k-tuple space increases exponentially with k. So if k is assigned to an arbitrary number, such as 10 or larger, the dimensionality of feature space will be $20^{10} \approx 10^{13}$. It is too large a feature space for learning.

In this paper we choose the following two different strategies for feature extraction. (a) *Without dimension reduction*: we take prediction based on full k-tuple space without any dimension reduction, and the maximum value of k is set to 5. As a result, at most $20^5 = 3.2 \times 10^6$ features are extracted. (b) *With dimension reduction*: when the proteins are represented in a high dimensional space, the occurrences of many k-tuples will be very scarce. The occurrence distributions of k-tuples for $k = 2, 3, 4$, and 5 are shown in Fig. 1. Note that some k-tuples just occur only once or even never occur in the dataset. Thus lots of them must be irrelevant to the subcellular localization since they are too sparse. Motivated by this phenomenon, we adopted the feature selection techniques from text classification to filter the k-tuple feature set.

Three feature selection methods derived from text categorization are adopted in this study, each of which uses a term-goodness criterion and a predefined threshold to achieve a desired degree of term elimination from the full k-tuple feature set. We try to find the most significant k-tuples from these selection procedures to improve the prediction accuracy. These feature ranking criteria are term frequency, Fisher linear discriminant criterion, and χ^2 statistics.

1) Term frequency: Term frequency [24] is the occurrence time of the specific term. We calculate the term frequency for each unique term in the training set and preserve a predefined proportion of the most high frequency terms. Those terms whose frequency rankings are lower than a given threshold are removed from the feature space. The basic assumption behind this selection criterion is that the rare terms are either non-informative for category prediction, or not influential in global performance. It is also possible that improvement of performance will be acquired if the rare terms are more likely to be noise terms.

2) Fisher linear discriminant criterion: Fisher linear discriminant analysis [25] is based on finding the direction that is the most efficient for discrimination. The original analysis and term criterion formula are both for two-category case, but they can be modified to a generalized form and extended to multi-category case.

For two-category case, let us consider the problem of projecting data from d dimensions onto a line, which is one dimension. The direction of this line is denoted by a vector $\mathbf{w}$. The Fisher linear discriminant analysis employs the linear function $\mathbf{w^t x}$, for which the following criterion function is maximum.

$$\mathcal{J}(\mathbf{w}) = \frac{|\widetilde{m}_1 - \widetilde{m}_2|^2}{\widetilde{s}_1^2 + \widetilde{s}_2^2} \tag{6}$$

where $\widetilde{m}_1$ denotes the sample mean for the projected points of category $\mathcal{C}_1$, $\widetilde{m}_2$ for points of $\mathcal{C}_2$, $\widetilde{s}_1^2$ denotes the variance for projected points of category $\mathcal{C}_1$, and $\widetilde{s}_2^2$ for points of $\mathcal{C}_2$.

Let $\mathcal{J}(\mathbf{w})$ represents the dimension-goodness. So we can use the criterion defined in (6) to rank the dimensions, providing that $\mathbf{w}$ is assigned to specific dimension parallel vector. Note that in feature space, one term is represented by a dimension, so dimension ranking is in fact the term ranking that we would like to acquire.

To naturally extend (6) to multi-category case, we firstly modify this criterion to a generalized form as follows,

$$\mathcal{J}(\mathbf{w}) = \frac{|\widetilde{m}_1 - \widetilde{m}|^2 + |\widetilde{m}_2 - \widetilde{m}|^2}{\widetilde{s}_1^2 + \widetilde{s}_2^2} \tag{7}$$

where $\widetilde{m}$ denotes the sample mean for all the projected points of category $\mathcal{C}_1$ and $\mathcal{C}_2$.

As a result, the criterion function for multi-category case can be formulated as

$$\mathcal{J}(\mathbf{w}) = \frac{\sum_{i \in \mathcal{Y}} |\widetilde{m}_i - \widetilde{m}|^2}{\sum_{i \in \mathcal{Y}} \widetilde{s}_i^2} \tag{8}$$

where $\mathcal{Y}$ is the label set of the dataset. Usually, we call the numerator, sum of $|\widetilde{m}_i - \widetilde{m}|^2$, the *between-category scatter*. Likewise the divisor, sum of $\widetilde{s}_i^2$, is called the *within-category scatter*. This modified form of Fisher discriminant criterion is also adopted for tumor classification [26].

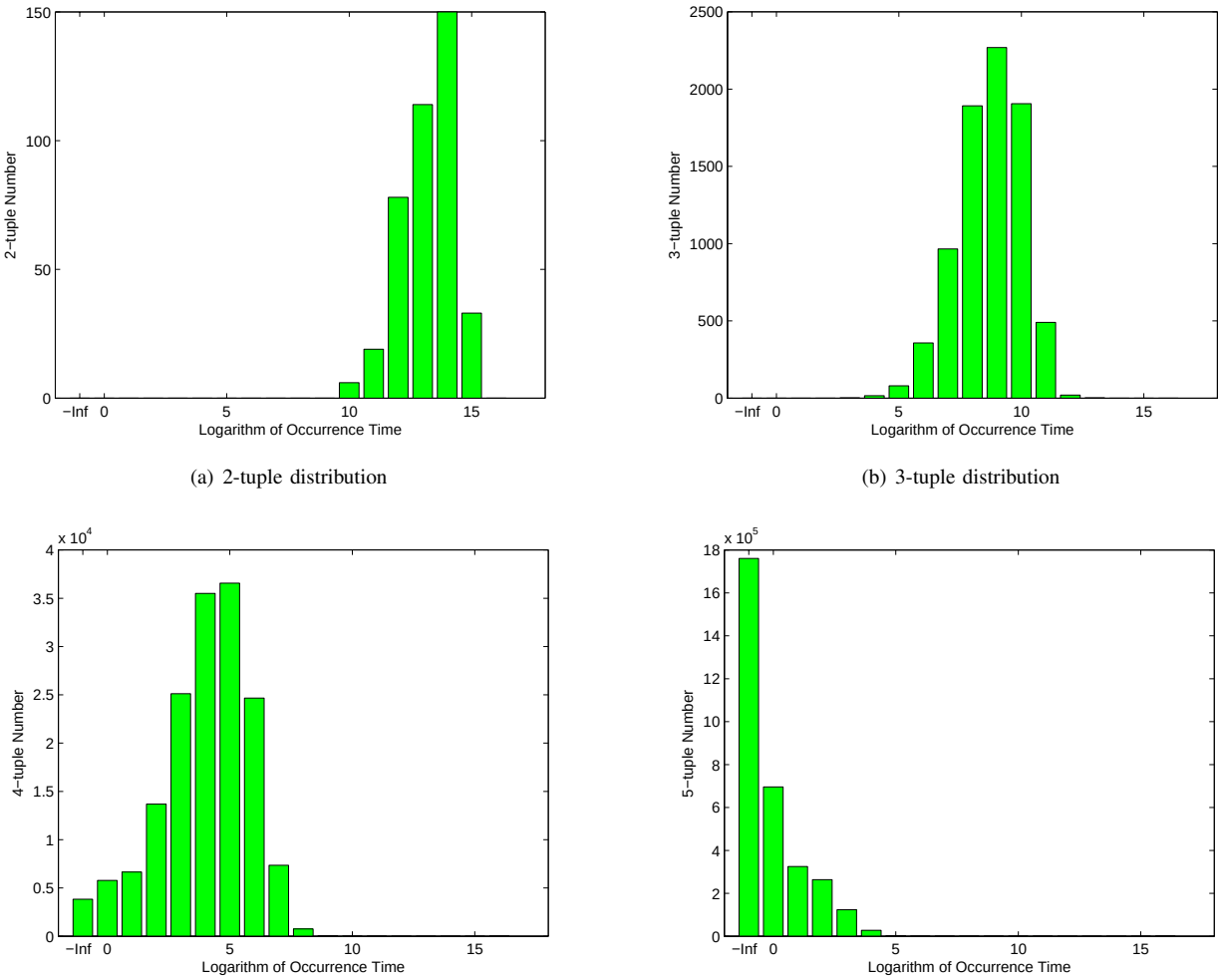
3) χ^2 statistics: A χ^2 statistics [24] is used to measure the lack of independence between term *t* and category *c* and can be compared to the χ^2 distribution with one degree of freedom to judge extremeness. By using the two-way contingency table of a term *t* and a category *c*, the term-goodness criterion is defined by

$$\chi^2(t, c) = \frac{N \times (AD - CB)^2}{(A + C) \times (B + D) \times (A + B) \times (C + D)} \tag{9}$$

where A denotes the number of times that *t* and *c* co-occur, B denotes the number of times the *t* occurs without *c*, C denotes the number of times the *c* occurs without *t*, D denotes the number of times neither *t* nor *c* occurs, and N is the total number of proteins.

The χ^2 statistic has a natural value of zero if *t* and *c* are independent. The higher χ^2 statistic value, the less independence between *t* and *c* holds. To compute the χ^2 statistic value for term *t*, we first compute for each category *c* and *t*, then combine the category specific scores for the term *t* into the following two scores,

$$\chi^2_{avg}(t) = \sum_{i \in \mathcal{C}} Pr(c_i) \chi^2(t, c_i) \tag{10}$$



(a) 2-tuple distribution

(b) 3-tuple distribution

(c) 4-tuple distribution

(d) 5-tuple distribution

Fig. 1. Distributions of k-tuples. Here, x-axis represents the logarithm to base 2 of the k-tuple occurrence times, that is, the occurrence times can be computed by 2^x, y-axis represents the number of k-tuples with greater than 2^{x-1} and less than or equal to 2^x occurrence times. For example, '-Inf' in x-axis denotes the k-tuples never occurrence, '0' denotes those with one occurrence time, and '5' denotes those with occurrence times between 2^4+1 and 2^5.

and

$$\chi^2_{max}(t) = \max_{i \in \mathcal{C}} \{\chi^2(t, c_i)\}. \tag{11}$$

Since the prior location distribution is not known to us, χ^2_{avg} statistic value can not be estimated accurately if the $Pr(c_i)$ is not accurate. In the experiment, we use the χ^2_{max} as our term-goodness criterion to rank the terms and then eliminate the terms of lower χ^2 statistic values.

C. Hybrid Features

Since the discriminative ability of amino acid composition is limited by only 20 features, and the improved performance (see Table III) using k-tuple space consumes too high computational resource, we try to make a tradeoff between these two types of feature sets.

In our hybrid model, we define each protein i (for $i = 1, \ldots, N$) as

$$\mathbf{P_i} = \begin{bmatrix} a_1 \\ \vdots \\ a_{20} \\ a_{k_1}(1) \\ \vdots \\ a_{k_1}(N_1) \\ \vdots \\ a_{k_M}(1) \\ \vdots \\ a_{k_M}(N_M)) \end{bmatrix} \tag{12}$$

where for $a_1, a_2, \ldots, a_{20}$, we use the amino acid composition form of (1). And we also incorporate the characteristic k-tuple subsequence by their occurrence time. So every element of the

vector $\mathbf{P_i}$ in (12) is given by

$$a_j = count_i(j) \quad \text{for } j = 1, \ldots, 20 \tag{13}$$

and

$$a_{k_m}(n) = count_i(k_m\text{-tuple}(n)) \quad \text{for } m = 1, \ldots, M \tag{14}$$

where k_m-tuple(n) denotes the nth highest score k_m-tuple in k_m-tuple space. The number of dimensions of $\mathbf{P_i}$ defined in (12) is given by

$$D = 20 + \sum_{m=1}^{M} N_m. \tag{15}$$

In our experiment, to simplify the hybrid model, we assign all N_m to a fixed value N, i.e.

$$N_m = N \quad \text{for } m = 1, \cdots, M. \tag{16}$$

Thus we get a feature vector whose dimensionality is

$$D = 20 + N \times M. \tag{17}$$

III. Experiment

A. Dataset and Evaluation

To have a critical comparison in this study, we use the dataset created by Park and Kanehisa [7]. The dataset consists of 7579 protein sequences (sequence similarity less than 80%, by ALIGN), all of which are eukaryotic proteins of 12 subcellular locations and collected from SWISS-PROT database release 39.0 [27]. Table I describes the location distributions of the dataset.

TABLE I
THE NUMBER OF PROTEINS USED IN THE DATASET

Subcellular locations	Number of proteins
Chloroplast	671
Cytoplasmic	1241
Cytoskeleton	40
Endoplasmic reticulum	114
Extracellular	861
Golgi apparatus	47
Lysosomal	93
Mitochondrial	727
Nuclear	1932
Peroxisomal	125
Plasma membrane	1674
Vacuolar	54
Total	7579

To evaluate our approach, we adopt 5-fold cross validation test, in which the dataset is divided into five subsets of approximately equal size. Then five rounds of training and test are carried out, and each time four subsets are used as training set and the other one as test set. Every performance measure is obtained by calculating the mean value of the results of five rounds of training and test. To be consistent with the previous work of other researchers, we use the same 5 folds as in [7].

We use five measures to assess our approach performance. They are standard precision (P), recall (R), F_1, total accuracy (TA) and location accuracy (LA). Three measures, P, R and F_1, are used to measure the prediction quality of each location. Two measures, TA and LA, are used to measure the overall prediction quality across all locations. These five measures can be defined by the following equations.

For each location l, the precision, recall and F_1 can be defined by true positive (TP), false positive (FP), false negative(FN) [28].

$$P_l = \frac{TP_l}{TP_l + FP_l} \tag{18}$$

$$R_l = \frac{TP_l}{TP_l + FN_l} \tag{19}$$

$$F_{1,l} = \frac{2 \times P_l \times R_l}{P_l + R_l}. \tag{20}$$

For the overall prediction, we use the total accuracy and location accuracy defined by the following equation. It is the same as defined in [7].

$$TA = \frac{\sum_{l=1}^{L} TP_l}{N} \tag{21}$$

$$LA = \frac{\sum_{l=1}^{L} ACC_l}{L} \tag{22}$$

where N is the total number of proteins in the dataset ($N = 7579$), L is the number of subcellular locations ($L = 12$), n_l is the number of proteins in each location l in Table I, and ACC_l is the accuracy for each location, defined by

$$ACC_l = \frac{TP_l}{n_l} = R_l. \tag{23}$$

B. Prediction and Result

Since support vector machine (SVM) is regarded as the state-of-the-art classifier, we adopt it as our predictor to testify our method. The SVM used in our experiment is partially based on the implementation of LibSVM version 2.82 [29]. We adopt one-versus-others strategy and RBF kernel, because this configuration is reportedly the best in this dataset [7].

The parameters used in the training process are selected from grid search procedure, which can be standardized as follows. γ is selected from $\{2^{-15}, 2^{-14}, \ldots, 2^{10}\}$ and C is selected from $\{2^{-2}, 2^{-1}, \ldots, 2^{12}\}$. The combination of γ and C which gives the highest total accuracy was used as the training parameter across the 5-fold cross validation.

1) Amino acid composition: According to (1), (3) and (4), three predictors based on different representations of amino acid composition were compared in our experiment. The best parameters (γ and C) for AAC-I, AAC-II and AAC-III are [2^{-10}, 2^3], [2^8, 2^2], and [2^6, 2^1], respectively. The detailed accuracy is shown in Table II.

From the experimental results, we can observe that the representation form of AAC-I gives the best prediction performance among these three representations although it is the roughest one. Surprisingly, the form AAC-III proposed by [6] could not give better performance than AAC-II even with the best parameters. It may be due to Bayes classifier that they used is different from the SVMs we used.

TABLE II

COMPARISON BETWEEN THREE REPRESENTATION OF AMINO ACID COMPOSITION (%)

Location	AAC-I			AAC-II			AAC-III		
	R	P	F_1	R	P	F_1	R	P	F_1
Chloroplast (671)	69.5	64.7	67.0	65.7	67.6	66.7	65.1	71.4	**68.1**
Cytoplasmic (1241)	66.8	65.4	66.1	65.8	64.2	65.0	64.7	67.1	**65.9**
Cytoskeleton (40)	48.0	91.0	60.2	76.0	88.7	**81.0**	70.7	93.3	79.3
ER (114)	56.2	71.6	62.8	49.0	76.3	58.5	55.1	77.3	**63.5**
Extracellular (861)	77.6	77.6	**77.6**	75.4	79.6	77.4	72.8	79.7	76.1
Golgi apparatus (47)	29.6	54.5	**37.7**	19.3	50.7	26.7	19.1	46.7	25.8
Lysosomal (93)	64.2	75.4	**68.8**	65.5	70.4	67.5	65.5	70.0	67.5
Mitochondrial (727)	45.4	60.1	**51.7**	45.2	57.2	50.4	40.7	59.3	48.2
Nuclear (1932)	86.1	73.4	**79.2**	85.6	72.7	78.6	88.8	67.7	76.8
Peroxisomal (125)	37.6	68.8	**48.1**	28.0	60.1	37.4	28.0	62.5	38.4
Plasma membrane (1674)	88.3	92.9	**90.5**	89.5	89.1	89.3	88.3	89.6	88.9
Vacuolar (54)	41.1	68.4	**49.0**	24.4	47.3	30.2	22.2	55.9	29.6
Total accuracy		**74.7**			73.8			73.4	
Location accuracy		**59.2**			57.5			56.8	

2) k-tuple subsequence: We compare the SVM prediction performance based on k-tuple feature set for different k from 2 to 5. The detailed values of performance measures are given in Table III. To make comparison with the amino acid composition features, we also include the best performance of prediction based on amino acid composition. We selected the best kernel parameter γ and penalty parameter C from grid search procedure. When k equals to 2, 3, 4 and 5, the parameter values are $[2^{-10}, 2^{4}]$, $[2^{-10}, 2^{2}]$, $[2^{-13}, 2^{5}]$ and $[2^{-15}, 2^{10}]$, respectively.

From the experimental result, we can observe that about 5.8% to 6.5% accuracy improvement can be obtained when we use k-tuple space with k equal to 4 or 5. In previous sequence-based prediction, the discriminative ability of k-tuples has never been solidly studied. Therefore our extensive experiments on those tens of thousands of k-tuples demonstrate that considerable potential discriminative abilities in the protein sequence have not been utilized so far. the amount of information of subcellular location encoded in the protein sequence is still open for biology research until now.

In order to further utilize the potential discriminative abilities of k-tuple spaces, we adopt the voting scheme. The scheme can be described as follows. For a protein to be predicted, the five classifiers, based on amino acid composition (ACC-I), 2-tuple, 3-tuple, 4-tuple and 5-tuple respectively, can be organized as a classifier committee. Then majority voting is adopted as the decision-making strategy. In case of a tie, the final decision will be made by randomly select one location from the highest vote getters. The result of voting gives the highest performance in our comparative study, which is compared with reported prediction results in Table VI.

Apparently, this voting scheme integrates all the k-tuple space information together, and achieves better performance while maintaining robust prediction, that is, it alleviates the influence by little mutation or missing in protein sequence.

3) Hybrid feature: In order to explore the k-tuple discriminative ability from another aspect, we introduced the hybrid feature set in Section II part C. Then we carried out experiments to testify the effectiveness. Concerned about the computational complexity, we assign some of the parameters in our new representation to fixed values, $M = 4$, $k_1 = 2$, $k_2 = 3$, $k_3 = 4$ and $k_4 = 5$. Table IV presents the total accuracy (TA) values of this hybrid approach using different feature selection methods. Table V compares performance of this feature extraction method with others. The comparison demonstrates that our feature extraction method makes a trade-off between accuracy and computational cost. The accuracy is 2.4% lower than 4-tuple prediction but the feature amount is reduced by about 99.5% from 20^4 to 820.

TABLE IV

TOTAL ACCURACY (%) COMPARISON BETWEEN THREE FEATURE SELECTION METHODS WITH VARIANT N

Method	Total accuracy (%)			
	$N = 20$	$N = 50$	$N = 100$	$N = 200$
Frequency	75.4	76.0	77.3	78.2
Fisher discriminant	75.7	76.4	77.6	78.6
χ^2 statistics	75.3	76.2	77.8	78.8
Number of Features	100	220	420	820

IV. DISCUSSIONS

A. Discriminative ability of k-tuple features

The exponentially increasing feature amount along with k inhibits us to further explore the predicting performance in k-tuple space. But we have obtained preliminary results on the relation of prediction accuracy to different k for ($k = 1, \ldots, 5$) as shown in Fig. 2. From the figure, we can observe that prediction accuracy increases with tuple length when k is less than 4, however, suffers from a little drop when k increases from 4 to 5. This observation is important. Does it hint that 4-tuple is the best choice? It will be very useful to have the knowledge about what performance can be ultimately achieved in k-tuple space.

TABLE III

COMPARISON OF SVM PREDICTION ACCURACY(%) USING k-TUPLE SPACES IN DIFFERENT k VALUES AND AMINO ACID COMPOSITION

Location	AAC-I			k-tuple space											
				k = 2			k = 3			k = 4			k = 5		
	R	P	F_1	R	P	F_1	R	P	F_1	R	P	F_1	R	P	F_1
Chloroplast (671)	69.5	64.7	67.0	72.3	74.2	73.2	71.1	80.5	75.4	76.8	90.5	83.0	76.9	93.6	**84.3**
Cytoplasmic (1241)	66.8	65.4	66.1	70.1	69.3	69.7	72.6	66.7	69.5	78.4	71.1	74.6	74.1	79.8	**76.8**
Cytoskeleton (40)	48.0	91.0	60.2	53.1	100.0	68.6	43.0	100.0	58.9	63.1	100.0	75.7	66.0	100.0	**77.9**
ER (114)	56.2	71.6	62.8	58.7	88.6	70.4	59.7	94.5	73.1	65.8	96.3	**78.1**	64.1	94.8	76.3
Extracellular (861)	77.6	77.6	77.6	78.5	77.8	78.1	77.9	75.0	76.4	81.0	82.4	81.6	76.0	89.8	**82.2**
Golgi apparatus (47)	29.6	54.5	**37.7**	25.6	60.0	35.8	10.9	80.0	18.9	12.9	80.0	21.3	6.4	60.0	11.6
Lysosomal (93)	64.2	75.4	68.8	59.1	74.2	65.2	54.7	89.0	66.5	59.1	92.6	71.3	60.2	93.6	**72.5**
Mitochondrial (727)	45.4	60.1	51.7	56.8	62.8	59.6	48.4	64.5	55.2	50.6	79.9	61.8	59.0	70.8	**64.1**
Nuclear (1932)	86.1	73.4	79.2	87.9	77.6	82.5	87.5	76.0	81.3	90.7	79.4	**84.6**	92.8	74.8	82.8
Peroxisomal (125)	37.6	68.8	48.1	33.5	70.3	45.0	33.4	91.3	47.7	41.5	90.7	**56.3**	40.0	91.3	55.0
Plasma membrane (1674)	88.3	92.9	90.5	92.0	91.8	**91.9**	94.0	89.8	91.8	95.9	86.0	90.7	93.0	82.6	87.4
Vacuolar (54)	41.1	68.4	49.0	35.6	63.1	45.0	29.8	90.5	42.8	54.0	97.5	**68.7**	52.0	94.4	65.9
Total accuracy		74.7			77.8			77.4			**81.2**			80.5	
Location accuracy		59.2			60.3			56.9			**64.1**			63.4	

TABLE V

COMPARISON OF F_1 VALUES (%) OBTAINED BY HYBRID FEATURE SET, AMINO ACID COMPOSITION, AND 4-TUPLE FEATURES

Locations	AAC-I	Hybrid features	4-tuple features
Chloroplast (671)	67.0	74.7	**83.0**
Cytoplasmic (1241)	66.1	69.9	**74.6**
Cytoskeleton (40)	60.2	65.1	**75.7**
ER (114)	62.8	73.4	**78.1**
Extracellular (861)	77.6	**81.8**	81.6
Golgi apparatus (47)	37.7	**42.5**	21.3
Lysosomal (93)	68.8	71.0	**71.3**
Mitochondrial (727)	51.7	61.4	**61.8**
Nuclear (1932)	79.2	82.8	**84.6**
Peroxisomal (125)	48.1	54.3	**56.3**
Plasma membrane (1674)	90.5	**91.7**	90.7
Vacuolar (54)	49.0	51.5	**68.7**
Total accuracy	74.7	78.8	**81.2**
Location accuracy	59.2	63.6	**64.1**

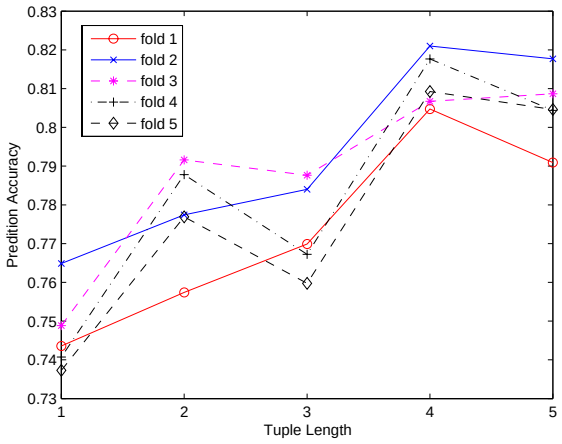


Fig. 2. Accuracy variance with tuple length across 5 folds

B. Comparison with other methods

In order to demonstrate the effectiveness of our proposed methods including k-tuple prediction and voting scheme on different k-tuple spaces, we made comparisons with the method developed by Park and Kanehisa [7] because we used the same dataset. The comparison results are summarized in Table VI.

To the best of our knowledge, our method exhibits the best performance among all existing prediction methods that do not use the external information extracted from any database. The improvement should owe to the discriminative ability of k-tuple space which has not been found so far. On the other hand, our experimental results also indicate that protein sequence itself contains important information for subcellular localization. This property is very useful when the predicting protein is newly discovered and no function annotation and other related information can be extracted from databases.

In the database annotation-based trend, some promising

TABLE VI

COMPARISON BETWEEN OUR METHOD AND PREVIOUS METHODS (%)

Location	Previous Method by Park and Kanehisa	Our Method		
	R	R	P	F_1
Chloroplast (671)	72.3	**79.7**	89.8	84.4
Cytoplasmic (1241)	72.2	**77.8**	75.0	76.3
Cytoskeleton (40)	**58.5**	55.9	100.0	69.7
ER(114)	46.5	**68.4**	94.3	79.1
Extracellular (861)	78.0	**84.0**	86.8	85.4
Golgi apparatus (47)	14.6	**17.3**	100.0	28.5
Lysosomal (93)	**61.8**	61.1	89.7	71.6
Mitochondrial (727)	57.4	**58.3**	78.4	66.8
Nuclear (1932)	89.6	**92.6**	78.0	84.7
Peroxisomal (125)	25.2	**39.9**	89.5	54.4
Plasma membrane (1674)	92.2	**95.6**	90.6	93.0
Vacuolar (54)	25.0	**46.5**	93.5	61.1
Total accuracy	78.2		**82.8**	
Location accuracy	57.9		**64.8**	

prediction accuracies were also achieved, such as Chou and Cai [18]. However, their method incorporated gene ontology and function domain information for prediction. As a result, improvement could be obviously obtained, since the gene ontology [17] was partially comprised of the subcellular localization annotation. However, it is not a general method for large-scale and newly discovered protein sequences.

V. Conclusions

This paper focuses on a study of exploring the potential discriminative ability of protein sequences. We carried out the prediction without external information but only using the amino acid sequences. The experimental results show that amino acid tuples have even more information related to subcellular localization than amino acid composition. This fact has not been solidly validated by previous works. Furthermore, we proposed a hybrid approach which combines different length tuple together, that is, amino acid composition features and characteristic tuple features obtained with feature selection methods of text categorization. The experimental results show that this hybrid method makes a tradeoff between feature amount and prediction accuracy. It is also a good choice since it retains the accuracy with a slight drop while significantly reducing the number of features.

We conducted experiments on a dataset consisting of 7579 protein sequences [7]. The highest performance we obtained using voting scheme among five classifiers is about 8.1% higher than amino acid composition features, and 4.6% higher than the previous highest accuracy achieved by methods based only on protein sequences.

Acknowledgement

The authors thank Ke Wu for his valuable advices. This research was partially supported by the National Natural Science Foundation of China via the grants NSFC 60375022 and NSFC 60473040.

References

[1] K. Nishikawa, Y. Kubota, and T. Ooi, "Classification of proteins into groups based on amino acid compostion and other characters," *J. Biochem.*, vol. 94, pp. 997–1007, 1983.

[2] A. Reinhardt and T. Hubbard, "Using neural networks for prediction of subcellular location of proteins," *Nucleic Acids Research*, vol. 26, pp. 2230–2236, 1998.

[3] K. C. Chou and D. W. Elrod, "Protein subcellular location prediction," *Protein Engineering*, vol. 12, pp. 107–118, 1999.

[4] K. C. Chou, "Prediction of protein cellular attributes using pseudo-amino acid composition," *Proteins: Structure, Function, and Genetics*, vol. 43, pp. 246–255, 2001.

[5] S. Hua and Z. Sun, "Support vector machine approach for protein subcellular localization prediction," *Bioinformatics*, vol. 17, pp. 721–728, 2001.

[6] Z. P. Feng, "Prediction of the subcellular location of prokaryotic proteins based on a new representation of the amino acid composition," *Biopolymers*, vol. 58, pp. 491–499, 2001.

[7] K. J. Park and M. Kanehisa, "Prediction of protein subcellular locations by support vector machines using compositions of amino acids and amino acid pairs," *Bioinformatics*, vol. 19, pp. 1656–1663, 2003.

[8] O. Emanuelsson, H. Nielsen, S. Brunak, and G. Heijne, "Predicting subcellular localization of proteins based on their N-terminal amino acid sequence," *J. Mol. Biol.*, vol. 300, pp. 1005–1016, 2000.

[9] H. Nielsen, J. Engelbrecht, S. Brunak, and G. Heijne, "Identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites," *Protein Engineering*, vol. 10, pp. 1–6, 1997.

[10] O. Emanuelsson, H. Nielsen, and G. Heijne, "ChloroP, a neural network-based method for predicting chloroplast transit peptides and their cleavage site," *Protein Science*, vol. 8, pp. 978–984, 1999.

[11] H. Bannai, Y. Tamada, O. Maruyama, K. Nakai, and S. Miyano, "Extensive feature detection of N-terminal protein sorting signals," *Bioinformatics*, vol. 18, pp. 298–305, 2002.

[12] C. Leslie, E. Eleazar, and W. S. Noble, "The spectrum kernel: a string kernel for SVM protein classification," in *Proceedings of the Pacific Symposium on Biocomputing*, vol. 7, 2002, pp. 566–575.

[13] C. Leslie, E. Eskin, J. Weston, and W. S. Noble, "Mismatch string kernel for SVM protein classification," in *Advances in Neural Information Processing Systems 15*. MIT Press, 2003, pp. 1417–1424.

[14] S. Menchetti, F. Costa, and P. Frasconi, "Weighted decomposition kernels," in *Proceedings of International Conference on Machine Learning*, 2005, pp. 585–592.

[15] Y. Yang and B. L. Lu, "Extracting features from protein sequences using chinese segmentation techniques for subcellular localization," in *Proceedings of the 2005 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB 2005)*, 2005, pp. 288–295.

[16] K. S. Toh, M. N. Nguyen, and J. C. Rajapakse, "LVQ approach using AA indices for protein subcellular localization prediction," in *Proceedings of the 2005 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB 2005)*, 2005, pp. 296–302.

[17] "the Gene Ontology website," available at http://www.geneontology.org/, 2006.

[18] K. C. Chou and Y. D. Cai, "Prediction of protein subcellular locations by GO-FunD-PseAA predictor," *Biochemical and Biophysical Research Communications*, vol. 320, pp. 1236–1239, 2004.

[19] Y. D. Cai and K. C. Chou, "Predicting 22 protein localizations in budding yeast," *Biochemical and Biophysical Research Communications*, vol. 323, pp. 425–428, 2004.

[20] K. C. Chou and Y. D. Cai, "Predicting protein localization in budding yeast," *Bioinformatics*, vol. 21, pp. 944–950, 2005.

[21] A. Höglund, P. Donnes, T. Blum, H. Adoplh, and O. Kohlbacher, "Using N-terminal targeting sequences, amino acid composition, and sequence motifs for predicting protein subcellular localization," in *Proceedings of the German Conference on Bioinformatics (GCB '05)*, 2005, pp. 45–59.

[22] A. Höglund, T. Blum, S. Brady, P. Donnes, J. S. Miguel, M.Rocheford, and et al., "Significantly improved prediction of subcellular localization by integrating text and protein sequence data," in *Proceedings of Pacific Symposium on Biocomputing*, vol. 11, 2006, pp. 16–27.

[23] A. Höglund, P. Donnes, T. Blum, H. Adoplh, and O. Kohlbacher, "MultiLoc: prediction of protein subcellular localization using N-terminal targeting sequences, sequence motifs and amino acid composition," *Bioinformatics*, vol. 22, pp. 1158–1165, 2006.

[24] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in *Proceedings of International Conference on Machine Learning*, 1997, pp. 412–420.

[25] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, pp. 179–188, 1936.

[26] S. Dudoit, J. Fridlyand, and T. P. Speed, "Comparison of discrimination methods for the classification of tumors using gene expression data," *Journal of the American Statistical Association*, vol. 97, pp. 77–87, 2002.

[27] A. Bairoch and R. Apweiler, "The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000," *Nucleic Acids Research*, vol. 28, pp. 45–48, 2000.

[28] D. D. Lewis, "Evaluating and optimizing autonomous text classfication systems," in *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '95)*, 1995, pp. 246–254.

[29] C.-C. Chang and C.-J. Lin, *LIBSVM: a library for support vector machines*, 2001, software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.