Prediction of Protein Subcellular Localization Based on Primary Sequence Data*

Mert Özarar[1], Volkan Atalay[1], and Rengül Çetin Atalay[2]

[1] Dept. of Computer Eng., Middle East Technical University, Ankara, TURKEY
{ozarar,volkan}@ceng.metu.edu.tr,
[2] Dept. of Molecular Biology and Genetics, Bilkent University, Ankara, TURKEY
rengul@bilkent.edu.tr,

Abstract. This paper describes a system called *prediction of protein subcellular localization (P2SL)* that predicts the subcellular localization of proteins in eukaryotic organisms based on the amino acid content of primary sequences using amino acid order. Our approach for prediction is to find the most frequent motifs for each protein (class) based on clustering and then to use these most frequent motifs as features for classification. This approach allows a classification independent of the length of the sequence. Another important property of the approach is to provide a means to perform reverse analysis and analysis to extract rules. In addition to these and more importantly, we describe the use of a new encoding scheme for the amino acids that conserves biological function based on *point of accepted mutations (PAM)* substitution matrix. We present preliminary results of our system on a two class (dichotomy) classifier. However, it can be extended to multiple classes with some modifications.

1 Introduction

Eukaryotic cells are subdivided into functionally separate membrane enclosed compartments. Each compartment and its vicinity contain functionally linked proteins related to the activity of that cell compartment [1]. In an eukaryotic cell, each protein is targeted to its specific cell localization where it is functionally active. Large scale genome analysis provides high number of putative genes to be characterized. Therefore, prediction of the subcellular localization of a newly identified protein is invaluable for the characterization of its function. Furthermore, studying subcellular localization is useful for understanding the disease mechanisms and developing novel drugs. If the rules for the prediction were biologically interpretable, this knowledge could help in designing artificial proteins with desired properties. Hence, a fully automatic and reliable prediction system for protein subcellular localization would be very useful.

* This work was supported by the Turkish Academy of Sciences to R.Ç.A. (in the framework of the Young Scientist Award Program-RCA/TÜBA-GEBİP/2001-2-3).

A. Yazici and C. Şener (Eds.): ISCIS 2003, LNCS 2869, pp. 611–618, 2003.

The aim of this work is to design and develop a system called *prediction of protein subcellular localization (P2SL)* that predicts the subcellular localization of proteins in eukaryotic organisms based on the amino acid content of primary sequences. The amino acid composition in the full or partial sequences can be taken as a global feature and the order may represent the local features such as the sequence order of amino acids that are found in protein sequence motifs [2]. In this paper, we are interested in the prediction using only local features. Our approach for prediction is to find the most frequent (hopefully the most significant) motifs for each protein (class) based on clustering and then to use these most frequent motifs as features for classification. This approach allows a classification independent of the length of the sequence. Another important property of the approach is to provide a means to perform reverse analysis and analysis to extract rules. In addition to these and more importantly, we describe the use of a new encoding scheme for the amino acids that conserves biological function based on *point of accepted mutations (PAM)* substitution matrix [3]. *PAM* is used to score aligned peptide sequences to determine the similarity of these sequences. The scores in *PAM* are derived by comparing aligned sequences of proteins with known homology and determining the observed. By using *PAM* substitution matrix, we believe that we are able to represent the chemical differences of each amino acid in protein sequences. In the literature, each amino acid is traditionally represented in binary form independent of their chemical properties. In this study, we present preliminary results of our system on a two class (dichotomy) classifier. However, it can be extended to multiple classes with some modifications.

The manuscript is organized as follows. In Section 2, we indicate the related studies. The data and computational methods used in this study are presented in Section 3. Experiments and comments on the results are then explained in the subsequent section. Finally, the eventual improvements are indicated together with conclusions and future work.

2 Related Work

Several attempts have been made to predict protein subcellular localization. Most of these prediction methods can be classified into two categories: one is based on the recognition of protein N-terminal sorting signals and the other is based on amino acid composition.

PSORT was developed based on rules for various sequence features of known protein sorting signals [4]. iPSORT was developed based on a decision tree with an amino acid index rule and an alphabet indexing and pattern rule [5]. TargetP was developed based on neural networks and achieved high prediction accuracy [6]. There are several studies to predict the specific localizations. MitoProt was developed for analyzing mitochondrial proteins and MitoProt II was for predicting the mitochondrial ones [7]. MTS prediction was based on hidden Markov models for mTPs [8]. SignalP was developed based on neural networks for SP prediction [9]. ChloroP was developed based on neural networks for chloroplast

targeting peptide prediction [10]. SortPred was developed for using both neural networks and hidden Markov models for four class problem of subcellular localization [11]. The prediction accuracy of SortPred is 86.4% for plant and 91.3% for non-plant, while the accuracy of TargetP and iPSORT is 85.3% and 84.5%, respectively for plant and 90% and 88%, respectively for non-plant. Among them, PSORT and iPSORT use global features, on the other hand MitoProt and MTS use local features. TargetP, SignalP, ChloroP and SortPred use both global and local features.

A combination of clustering followed by classification seems to be the most important aspect of this study. *P2SL* dichotomizer prediction results are very promising compared to the experimental results indicated in the literature. Our ultimate goal is to design a system that performs predictions only on human proteins yielding comparable results with TargetP.

When compared to the studies in the literature,

- we use windows (motifs) similar to the work by Emanuelsson et.al. [6],
- we use self organizing maps (SOM) similar to the work by Cai et.al. [12],
- we use a novel encoding scheme based on PAM [13].

3 Methods

The main idea for the prediction of protein subcellular localization using local features is based on finding the substrings which are common for a protein class and infrequent for the other classes. Such substrings are called as motifs. We use a self organizing map for this purpose. For an unknown input sequence, we determine which motifs exist and the sequence is classified according to this information. The flow diagram of *P2SL* is illustrated in Figure 1. The input to the system is amino acid sequences. The sequences are extracted from this data. The primary sequence is then decomposed into substrings. Each substring is encoded with *PAM250* [13] substitution matrix. We apply clustering on the encoded substring via a self organizing map. During the training phase, motifs for each class are determined. Throughout the test phase, when the substrings of an unknown input sequence is given, according to the winning nodes in the self organizing map, we form a binary vector which indicates the existence of motifs of a particular class in the input. k-nearest neighborhood classifier is then applied to this binary vector to determine the label of the unknown protein.

3.1 Data Representation

Protein sequences are strings of arbitrary size and amino acids correspond to the letters in a protein sequence. Let $\hat{X}$ represent a protein sequence whose length is $len(\hat{X})$. $\hat{X}$ can be decomposed into substrings of some fixed length, κ. If $\kappa < len(\hat{X})$, there are exactly $(len(\hat{X}) - \kappa + 1)$ substrings in $\hat{X}$. $\hat{X}(j : m + j)$ then denotes j^{th} substring in a protein sequence $\hat{X}$. In order to perform further computational analysis, we need to encode the amino acids. Although, the most

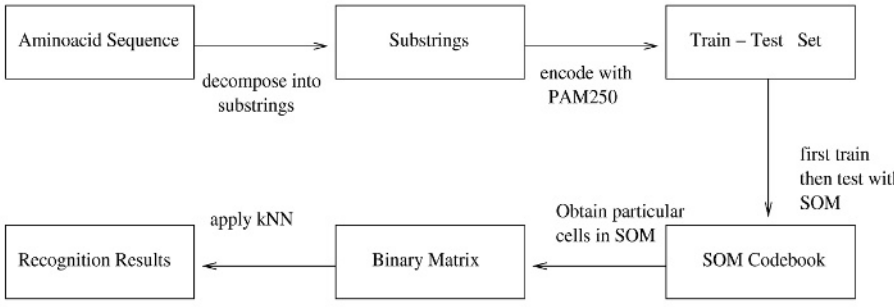

Fig. 1. Flow diagram of *P2SL*.

popular way of encoding reported in the literature is to represent each amino acid in binary form, in this study, we make use of substitution matrices. While aligning two protein sequences, certain methods are used to score the alignment of one residue against another. Substitution matrices indicate score values for this purpose. We employ *PAM250* scoring matrix to encode an amino acid. In the rest of the manuscript, X denotes a *PAM* encoded protein sequence $\hat{X}$.

The frequencies of these mutations are in this table as a "log odds-matrix" where:

$$M_{ij} = 10(\log 10\ R_{ij}),$$

In this equation, M_{ij} is the matrix element and R_{ij} is the probability of that substitution as observed in the database, divided by the normalized frequency of occurence for amino acid i. All of the number are rounded to the nearest integer. The base 10 log is used so that the numbers can be added to determine the score of a compared set of sequences, rather than multiplied.

3.2 Clustering

We use a self organizing map (SOM) for clustering. SOM is an unsupervised artificial neural network model that relates similar input vectors to the same region of a map of nodes or neurons [14]. Topological organization of neurons in SOM is its essential feature, since it does a topological ordering of the input. SOM is often used to categorize and interpret data, by mapping a high-dimensional input data to a lower dimensional space which is usually 2. Each input data is composed of a vector of elements. The map is an array of nodes which are also called neurons and it is often laid out in a rectangular or hexagonal lattice. Each node has an associated reference vector of the same size as input feature vectors. Input vectors are compared with these reference vectors. There are two phases of a SOM: training and testing. For training, all the input vectors, one at a time are presented to the network. Each input vector is compared to weight vectors associated with every neuron. The neuron having the weight vector with

the smallest difference to the current input vector becomes the winning neuron. The weight vector of this winning neuron and those of the neighboring neurons as well are then updated in the direction of the input vector. Note that in this training scheme similar input vectors are mapped to nodes that are close together on the map. This, in fact generates both the topological ordering and clustering properties of a SOM. In the training phase, there is no update in the weight vectors and therefore we are only interested to determine to which of the nodes (cluster) the input data is mapped.

In our system, clustering occurs during training and in this phase substrings are topologically grouped. The problem of finding motifs of a protein class turns out to finding the nodes specific to a class. At the end of training, these nodes are found by simply looking at the difference of the number of substrings of two classes assigned to each node. "assigned" is used in the sense that for an input vector X, SOM cell i is the winning node.

Let $c_i^{\mathcal{X}}$ and $c_i^{\mathcal{Y}}$ denote the cardinal of the set of substrings of class $\mathcal{X}$ and class $\mathcal{Y}$, respectively assigned to the cell i. If the size of the two dimensional SOM is m by n, then there are $m \times n$ nodes. The difference of substrings of the two classes assigned to cell i is then

$$\Delta c_i^{\mathcal{X}} = (c_i^{\mathcal{X}} - c_i^{\mathcal{Y}}) \text{ and } \Delta c_i^{\mathcal{Y}} = -\Delta c_i^{\mathcal{X}} = (c_i^{\mathcal{Y}} - c_i^{\mathcal{X}})$$

Let the sets of SOM nodes to which motifs of class $\mathcal{X}$ and $\mathcal{Y}$ are assisgned during the training phase be $P^{\mathcal{X}}$ and $P^{\mathcal{Y}}$, respectively. $P^{\mathcal{X}}$ and $P^{\mathcal{Y}}$ are determined as follows. If $\Delta c_i^{\mathcal{X}} > \tau^{\mathcal{X}}$ then $i \in P^{\mathcal{X}}$, otherwise i is not in $P^{\mathcal{X}}$ where $\tau^{\mathcal{X}}$ is a threshold value emprically determined. Similarly, If $\Delta c_j^{\mathcal{Y}} > \tau^{\mathcal{Y}}$ then $j \in P^{\mathcal{Y}}$, otherwise j is not in $P^{\mathcal{Y}}$. Remark that we choose $\tau^{\mathcal{X}}$ and $\tau^{\mathcal{Y}}$ such that there are equal number s of elements in $P^{\mathcal{X}}$ and $P^{\mathcal{Y}}$.

3.3 Classification

The k nearest neighbor (kNN) classification method is one of the most popular classification methods in pattern recognition. In kNN, the class label of an unknown sample is determined by the majority of class labels of the k nearest training samples to the unknown sample. Nearest neighbor method using only one nearest training sample is called 1-Nearest Neighbor (1NN) method. On the other hand, that using $k > 1$ nearest training samples is called k-nearest neighbor (kNN) method. Compared to linear or quadratic classifers, the nearest neighbor method is more effective for a complex distribution of samples. On the other hand, kNN method requires more computational cost than 1NN method.

After clustering, the next step is classification. For this purpose, for each training input X a binary vector Z of $2s$ elements is formed as follows. The elements $0, \ldots, s-1$ represent class $\mathcal{X}$ while those $s, \ldots, 2s-1$ represent class $\mathcal{Y}$. If the winning node corresponding to input substring $X(j : j+\kappa)$ is l^{th} element of $P^{\mathcal{X}}$, then $Z(l) = 1$. Similarly, for class $\mathcal{Y}$, if the winning node corresponding to input substring $X(j : j+\kappa)$ is l^{th} element of $P^{\mathcal{Y}}$, then $Z(s+l) = 1$. Note that final form of Z is obtained after processing all of the available substrings in a protein sequence X.

Assume that Z' represents the binary vector for a protein sequence in the training set and Z for that in the test set. For each protein i in the test set, Hamming distance between corresponding Z_i and Z'_j (for all of the elements in the training set) is calculated. Then, k proteins of the training set having the least Hamming distance to Z_i is checked. Suppose that among those k protein sequences, there are q proteins belonging to class $\mathcal{X}$ and r proteins belonging to class $\mathcal{Y}$. Hence, q+r=k. Then, a voting mechanism takes place: if $q > r$, then the i^{th} element of the test set is classified as of class $\mathcal{X}$. If $r > q$, then the i^{th} element of the test set is classified as class of $\mathcal{Y}$. Since, the k is chosen as an odd value, there is no chance of $q = r$.

4 Material, Results, and Discussions

In our experiments, we first employ our method to a previously published data set [6]. By using this data set, we have generated four data subsets (classes), signal peptide containing proteins (SP class), nuclear proteins (NP class), cytosolic proteins (CP class) and mitochondrial proteins (MP class), of which only two (SP and NP) are used in this study. Both classes are represented in *Fasta* format.

From each class, 40 proteins are randomly chosen for training from the input data. Randomly chosen 20 proteins from each class exclusive of those in the training set are used for testing. Substrings of size $\kappa = 30$ are extracted from the protein sequences. Therefore, 18815 SP substrings and 23290 NP substrings are formed. For the self organizing map, SOM-PAK [15] is used as a computational tool. We have tried different map sizes, topologies and neighborhood functions. Experiments yield that a SOM with map size 25x25, randomly initialization, rectangular topology and Gaussian neighborhood function gives better results for our problem. Both ordering and fine-tuning training is used with 10000 and 50000 epochs, respectively. Figure 2 shows the difference of histograms of SOM nodes after the training phase. The peaks are obvious for both classes, but the important nodes are the ones in which a high number of samples exists for one class yet not a significant value exists for the other class. For example, the difference value for node (0,0) is approximately 40. This shows that unless there are cells with lower difference values, it shall not be treated as a particular node for both classes. However, for instance, the frequency difference between SP and NP is approximately 90 for the node (24,0) and it is one of the specific cells for the SP class.

By keeping the thresholds ($\tau^{\mathcal{X}} = \tau^{\mathcal{Y}}$) as 20, (16,7), (14,5), (16,8), (17,4), (17,8), (12,8), (17,5), (18,9), (19,9), (19,6), (15,8), (17,7), (13,8), (19,8), (18,7), (14,4), (17,11), (18,3), (15,7), (0,19) represent the particular set of cells for class NP and (24,0), (15,0), (0,3), (7,19), (6,21), (16,0), (9,4), (24,19), (24,21), (3,10), (6,22), (21,0), (5,2), (8,0), (5,3), (9,24), (4,9), (4,24), (6,24), (9,0) represent the specific set of nodes for class SP. The k value is taken as 5, for kNN and 100% accuracy is obtained from the test results. The test set ratio over training set ratio is 1/4 which is quite large for a pattern recognition experiment. This shows that our methodology is meaningful even though the sample size is small.

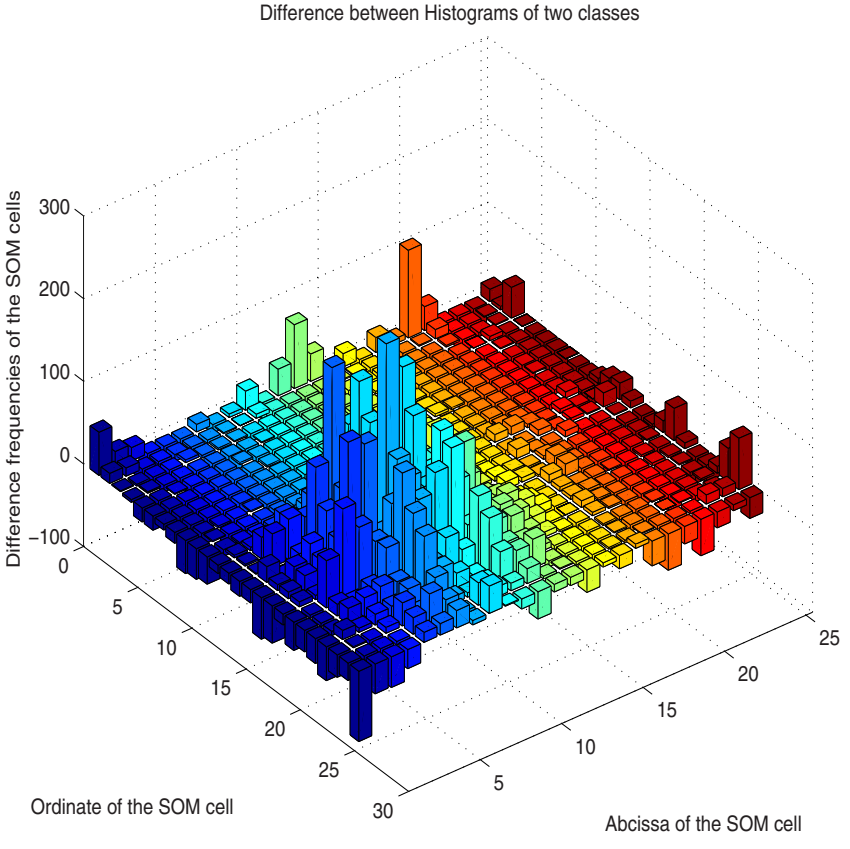

Fig. 2. Difference of histograms of SOM cells for SP and NP classes.

5 Conclusion

We describe *P2SL* for the prediction of protein subcellular localization sites using amino acid order. It achieves higher prediction accuracy than the previous two class classifiers. Our clustering strategy is training self organizing maps and using kNN for classification purposes. In order to extract features from amino acid sequences, *PAM250* scoring matrix is used. This preserves the biological meaning of each independent amino acid found in protein subcellular targeting sequence motifs. Our classification strategy is based on choosing the dominant vectors which are extracted by clustering. As the next step, we intend to increase the number of classes to 4 rather than 2. Cytoplasmic and mitochondrial classes shall be added. More samples should be used for training the SOM. Futhermore, another classifier such as multilayer perceptrons can be employed together with kNN to obtain better results. In addition to these, SOM clustering can be useful for rule extraction and reverse analysis.

References

1. C. van Vliet, E.C. Thomas, A. Merino-Trigo, R.D. Teasdale, P.A. Gleeson: Intracellular sorting and transport of proteins. *Prog Biophys Mol Biol.*, 83(1)(2003) 1–45.
2. F. Corpet, F. Servant, J. Gouzy and D. Kahn: ProDom and ProDom-CG: tools for protein domain analysis and whole genome comparisons. *Nucleic Acids Res.*, 28(2000) 267–269.
3. M.O. Dayhoff, R.M. Schwartz and B.C. Orcutt: A model of evolutionary change in proteins. *Atlas of protein sequence and structure. Vol. 5, Suppl. 3. National Biomedical Research Foundation, Washington, D.C.*, 1979 345–352.
4. K. Nakai and M. Kanehisa: A knowledge base for predicting protein localization sites in the eukaryotic cells, *Genomics*, 14(1992) 897–991.
5. iPSORT is available at: *http://hypothesiscreator.net/iPSORT*
6. O. Emanuelsson, H. Nielsen, S. Brunak and G. von Heijne: Predicting subcellular localization of proteins based on their N-terminal amino acid sequence, *J.Mol.Biol.*, 300(2000) 1005–1016.
7. M.G. Claros: MitoProt: a Macintosh application for studying mitochondrial proteins, *Computer Applications in the Biosciences* 11(4)(1995) 441–447.
8. Y. Fujiwara, H. Asogawa and K. Nakai: Prediction of mitochondrial targeting signals using hidden Markov models, *Genome Informatics*, 8(1997) 53–60.
9. H. Nielsen, J. Engelbrecht, S. Brunak, G. von Heijne: A neural network method for identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites, *International Journal of Neural Systems*, 8(5–6)(1997) 581–599.
10. O. Emanuelsson, H. Nielsen and G. von Heijne: ChloroP, a neural network-based method for predicting chloroplast transit peptides and their cleavage sites, *Protein Sci.*, 8(1999) 978–984.
11. Y. Fujiwara and M. Asogawa: Prediction of Subcellular Localization Using Amino Acid Composition and Order, *Genome Informatics*, 12(2001) 103–112.
12. Y. Cai, X. Liu, K. Chou: Artificial neural network model for predicting protein subcellular location, *Computers and Chemistry*, 26(2002) 179–182.
13. S.F. Altschul: Amino acid substitution matrices from an information theoretic perspective, *J. Mol. Biol.*, 219(1991) 555–565.
14. T. Kohonen: The self-organizing map. *Proceedings of the IEEE*, 78(9)(1990) 1464–1480.
15. The SOMPAK package is available at: *http://www.cis.hut.fi/nnrc/papers/*