Volume 4, Issue 7, July 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Modelling of Protein Sub-cellular Sites using Hidden Markov Model: A Review

Nivedita Rao
Master of Technology (Department of Computer Science and Engineering) GJUS&T Hisar, Haryana, India

Ms. Sunila Godara
Assistant Professor (Department of Computer Science and Engineering) GJUS&T Hisar, Haryana, India

Abstract— Predicting the location of the protein in the cell helps us to understand its structure and function better. Subcellular localization is a well-designed representation of proteins. We need a reliable method for predicting the protein subcellular locations. It would be significant in interpreting the original data produced by the large scale genome sequencing projects. The aim of this paper is to provide an overview of the use Hidden Markov Model for the prediction of protein subcellular sites. Using sequence data, HMM achieves higher prediction accuracy than any other methods in many cases. HMM adds some beneficial structural and parametric assumptions such as hidden state variables that are useful for prediction. Currently this method is limited to time series data.

Keywords— Protein subcellular location, Hidden Markov Model, Protein sequence analysis, Amino acid composition, sequence alignment.

I. INTRODUCTION

The advancements in genome has increased dramatically over the recent years, thus resulting in the explosive growth of biological data widening the gap between the number of protein sequences stored in the databases and the experimental annotation of their functions. Knowledge of the subcellular localization of a protein can significantly improve target identification during drug discovery process. A number of protein subcellular localization methods have been developed. Most of them have been developed in the past and can be divided into two categories: one is based on the identification of protein N-terminal sorting signals and other is based on amino acid composition [1].

Markov models are well known tools for analysing biological sequence data. Hidden Markov models are based around the idea of statistically modelling the signal or time series under consideration. The time series is characterized by a parametric statistical model which is configured so as to maximize the probability that the series could have been generated by that model. Since the model optimization can be done algorithmically, a HMM system can adapt its own model to describe the data well. That is, it can learn from structure of the time series from a set of training data, and describe that structure by way of a statistical model. The model can then be used to classify or recognize new data. Thus, a HMM performs a very similar task to a conventional supervised learning scheme, with the exception that it works on time-series data. The theory of HMMs first appeared in a series of classic papers by Baum and his colleagues during the late 1960's and early 1970's [2] [3]. Hidden Markov Models (HMMs) are an extremely versatile statistical representation that can be used to model any set of one-dimensional discrete symbol data. HMMs can model protein sequences in many ways, depending on what features of the protein are represented by the Markov states. For protein structure prediction, states have been chosen to represent homologous sequence positions [4], local or secondary structure types [5] [6], or transmembrane locality [7] [8]. HMMs can be applied in many fields where the goal is to recover a data sequence that is not immediately observable (but other data that depends on the sequence is). Applications include:

- Cryptanalysis
- Speech recognition
- Speech synthesis
- Part-of-speech tagging
- Machine translation
- Partial discharge
- Gene prediction
- Alignment of bio-sequences
- Time Series Analysis
- Human Activity recognition

In this paper we discuss how Hidden Markov Model (HMM) is used in the prediction of protein subcellular sites. In section 2, we discuss Protein subcellular sites. Section 3 discusses HMM, three problems of HMM used to represent real world problems, forward and backward algorithm, Viterbi algorithm and Baum-welch algorithm. Section 4 discusses the

ways HMM is used in the field of bioinformatics, HMM based profile, HMM based multiple sequence alignment and HMM to cluster sequences. Section 5 discusses the related work and section 6 contains concluding remarks.

II. PROTEIN SUBCELLULAR SITES

Subcellular localization of protein is one of the main functional characters as they must be localized correctly at the subcellular level to have a smooth and normal biological function. A protein's functional description is often indicative of its subcellular localizations.

Different cellular environments call for different biophysical properties of the proteins native to these environments such as inner membrane proteins are characterized by the presence of α- helical transmembrane regions [9] and the structure corroborates the concept that all outer membrane proteins consists of β-barrels [10].

Predictions of subcellular sites can be done on the basis of amino acid composition, by integrating various protein characteristics such as targeting motifs of different organelles [11], based on homology like proteome analyst. Many different protein subcellular sites prediction tools have been developed such as CELLO (sub**CEL**lular **LO**calization predictive system) [12], uses Support Vector Machine based on n-peptide composition to assign a Gram-negative protein to the cytoplasm, inner membrane, periplasm, outer membrane or extracellular space,SignalP [13] is used to predicts traditional N-terminal signal peptides in both prokaryotic and eukaryotic proteinsetc.

III. HIDDEN MARKOV MODEL

Hidden Markov model is a powerful statistical tool for modelling a wide range of sequential or time series data. A HMM can be visualized as a finite state machine composed of finite states. The HMM can generate a protein sequence by emitting symbols as it progresses through series of states. In a *hidden* Markov model, the state is not directly visible, but output, dependent on the state, is visible. Each state has a probability distribution over the possible output tokens as shown in figure 1.

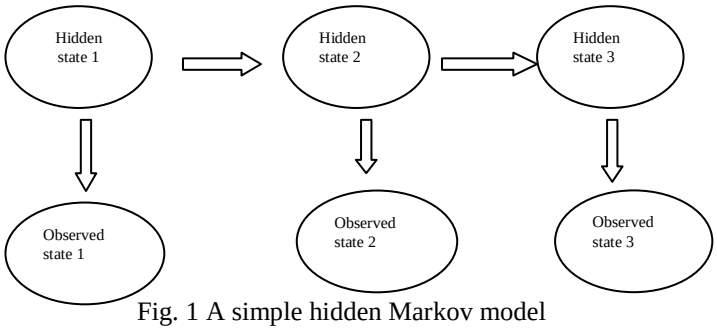

Fig. 1 A simple hidden Markov model

A. FORMAL DEFINITION OF HMM

Let the model be λ such that

$$\lambda = (A, B, \pi) \quad (1)$$

S is our state alphabet set, and V is the observation alphabet set:

$$S = (s_1, s_2, \ldots, s_N) \quad (2)$$

$$V = (v_1, v_2, \ldots, v_M) \quad (3)$$

We define Q to be a fixed state of length T, and corresponding observations O :

$$Q = q_1, q_2, \ldots, q_T \quad (4)$$

$$O = o_1, o_2, \ldots, o_T \quad (5)$$

A is the transition array, storing the probability of state j following state i. Note the state transition probabilities are independent of time:

$$A = [a_{ij}],\ a_{ij} = P(q_t = s_j | q_{t-1} s_i) \quad (6)$$

B is the observation array, storing the probability of observation k being produced from the state j, independent of t:

$$B = [b_i(k)],\ b_i(k) = P(x_t = v_k | q_t = s_i) \quad (7)$$

π is the initial probability array:

$$\pi = [\pi_i],\ \pi_i = P(q_1 = s_i) \quad (8)$$

Two assumptions are made by the model [14]. The first, called the Markov assumption, states that the current state is dependent only on the previous state, this represents the memory of the model:

$$P(q_t | q_1^{t-1}) = P(q_t | q_{t-1}) \quad (9)$$

The independent assumption states that the output observation at time t is dependent only on the current state; it is independent of previous states and observations:

$$P(o_t | o_1^{t-1}, q_1^{t}) = P(o_t | q_t) \quad (10)$$

B. FUNDAMENTAL HIDDEN MARKOV MODEL PROBLEMS AND SOLUTIONS

There are basically three problems which must be solved in order for HMMs to be used usefully in the real- world applications.

1. ***The model evaluation problem:*** Given a model along with sequence of observations, what will be the probability of the observations that were generated by the model? The solution to this can be used to find the unseen sequences, given a well-trained model. This problem can be solved by using forward and backward algorithm [15].

2. ***The decoding problem:*** Given a model along with sequence of observations, what will be the most likely the sequence of states to have produced the observations? This problem can be solved by using Viterbi algorithm [16].
3. ***Learning problem:*** Given a model along with sequence of observations, what will be the best set of parameters for the model produced by the observations? This problem can be solved by using Baum –Welch algorithm [17].

C. FORWARD ALGORITHM

It is the solution for evaluation problem [15][3]. It is a recursive algorithm for calculating $\alpha_t(i)$for the observation sequence of increasing length t. The probabilities for the single-symbol sequence is calculated as a product of initial *i*-th state probability and emission probability of the given symbol *o*(1) in the *i*-th state. A recursive procedure is applied.
The formal definition is as follows:
Initialization:

$$\alpha_1(i) = p_i\, b_i(o(1)) \text{ , } i = 1, \dots, N \quad (1)$$

Recursion:

$$\alpha_{t+1}(i) = \left[\sum_{j=1}^{N} \alpha_t(j) a_{ji}\right] b_i\left(o(t+1)\right) \quad (2)$$

here $i = 1, \dots, N$, $t = 1, \dots, T - 1$

Termination:

$$P(o(1)o(2)\ldots..o(T)) = \sum_{j=1}^{N} \alpha_T(j) \quad (3)$$

D. BACKWARD ALGORITHM

The Backward Algorithm calculates recursively backward variables going backward along the observation sequence [15]. We can add a symmetrical backward variable $\beta_t(i)$ as the conditional probability of the partial observation sequence from $o(t+1)$ to the end to be produced by all state sequences that start at *i*-th state.

$$\beta_t(i) = P(o(t+1), o(t+2), \dots, o(T) \mid q(t) = q_i) \quad (4)$$

Initialization:

$$\beta_T(i) = 1 \text{ , } i = 1, \dots, N \quad (5)$$

According to the above definition, $\beta_T(i)$ does not exist. This is a formal extension of the below recursion to t = T.
Recursion:

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij}\, b_j\left(o(t+1)\right)\beta_{t+1}(j) \quad (6)$$

here $i = 1, \dots, N$; $t = T - 1,\ T - 2, \dots, 1$

Termination:

$$P\left(o(1)o(2) \dots o(T)\right) = \sum_{j=1}^{N} p_j\, b_j\left(o(1)\right)\beta_1(j) \quad (7)$$

Obviously both Forward and Backward algorithms must give the same results for total probabilities

$$P(O) = P(o(1), o(2), \dots, o(T)) \quad (8)$$

E. VITERBI ALGORITHM

The forward algorithm gives the probability that an observation sequence was generated by a given HMM, but does not tell anything about the sequence of states. Knowledge of the state sequence is often useful, especially if the states have some relationship with physical events [16][3].
Because the state sequence is hidden in HMM, the best that can be done in practice to find the most probable state sequence to have generated the observed outputs. This can be achieved by *Viterbi algorithm*, a modification of forward algorithm. The Viterbi procedure remembers the most probable transition into the state at each time period.
Let $\delta_t(i)$ be the maximal probability of state sequences of the length *t* that end in state *i* and produce the *t* first observations for the given model.

$$\delta_t(i) = \max\{P(q(1), q(2), \dots, q(t\text{-}1) ; o(1), o(2), \dots, o(t) \mid q(t) = q_i).\} \quad (9)$$

Initialization:

$$\delta_1(i) = p_i\, b_i(o(1)) \quad (10)$$

$$\psi_1(i) = 0 \text{ , } i = 1, \dots, N \quad (11)$$

According to the above definition, $\beta_T(i)$ does not exist.

Recursion:

$$\delta_t(j) = \max_i [\delta_{t-1}(i)\, a_{ij}]\, b_j(o(t)) \quad (12)$$

$$\psi_t(j) = \arg\max_i [\delta_{t-1}(i)\, a_{ij}] \quad (13)$$

Termination:

$$p^* = \max_i [\delta_T(i)] \quad (14)$$

$$q^*_T = \arg\max_i [\delta_T(i)] \quad (15)$$

F. BAUM-WELCH ALGORITHM

Let us define $\xi_t(i, j)$, the joint probability of being in state q_i at time t and state q_j at time $t+1$, given the model and the observed sequence:

$$\xi_t(i, j) = P(q(t) = q_i, q(t+1) = q_j \mid O, \Lambda) \tag{16}$$

Therefore we get

$$\xi_t(i,j) = \frac{\alpha_t(i)a_{ij}\,b_j(o(t+1))\beta_{t+1}(j)}{P(O/\Lambda)} \tag{17}$$

The probability of output sequence can be expressed as [17][3]:

$$P(O/\Lambda) = \sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)a_{ij}\,b_j\big(o(t+1)\big)\beta_{t+1}(j) = \sum_{i=1}^{N}\alpha_t(i)\beta_t(i) \tag{18}$$

The probability of being in state q_i at time t:

$$\gamma_t(i) = \sum_{j=1}^{N}\xi_t(i,j) = \frac{\alpha_t(i)\beta_t(i)}{P(O/\Lambda)} \tag{19}$$

Estimates

Initial probabilities:

$$\overline{p_i} = \gamma_1(i) \tag{20}$$

Transition probabilities:

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1}\xi_t(i,j)}{\sum_{t=1}^{T-1}\gamma_t(i)} \tag{21}$$

Emission probabilities:

$$\bar{b}_{jk} = \frac{\sum_{t}^{*}\gamma_t(j)}{\sum_{t=1}^{T}\gamma_t(j)} \tag{22}$$

In the above equation Σ^* denotes the sum over t so that $o(t) = o_k$.

IV. HMM IN THE FIELD OF BIOINFORMATICS

The HMMs can be applied efficiently to well-known biological problems. That why HMMs gained popularity in bioinformatics, and are used for a variety of biological problems like protein secondary structure recognition, Multiple sequence alignment, Gene finding. Hidden Markov Models (HMMs) are an extremely versatile statistical representation that can be used to model any set of one-dimensional discrete symbol data. HMMs can model protein sequences in many ways, depending on what features of the protein are represented by the Markov states. For protein structure prediction, states have been chosen to represent homologous sequence positions, local or secondary structure types, or transmembrane locality. The resulting models can be used to predict common ancestry, secondary or local structure, or membrane topology by applying one of the two standard algorithms for comparing a sequence to a model.

Categorizing nucleotides within a genomic sequence can be interpreted as a classification problem with a set of ordered observations that posse's hidden structure, that is a suitable problem for the application of hidden Markov models.

The HMM is composed of a number of states, each state 'emits' symbols(residues) according to symbol-emission probabilities, these states are connected by state-transition probabilities. Starting from a certain initial state, a sequence of states is generated by moving from state to state according to the state transition probabilities until last state is reached. Each state then emits symbols according to the state's emission probability, creating an observable sequence of symbols. A simple HMM for heterogeneous DNA sequence is shown in fig. 2 [18].

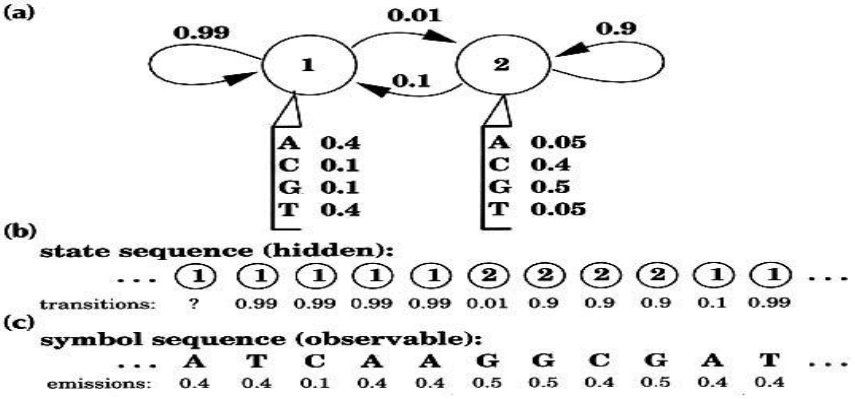

Fig. 2 A simple two state HMM describing DNA sequence

A. HMM-BASED PROFILES

The HMM-based profiles make two major contributions. First, HMMs can be trained from aligned as well as unaligned data. Second, HMM-based profiles use a justifiable statistical treatment of insertions and deletions. In standard profiles, it is impossible to determine insertion and deletion score except by trial and error method and since handling insertions and deletions is a major problem in recognizing highly divergent protein sequences, the recasting of profiles as HMMs promises a significant increase in the power of profiles to recognize distantly related structural homologs.

B. HMM- BASED MULTIPLE SEQUENCE ALIGNMENT

HMM's can be trained from a set of unaligned example sequences, producing a multiple alignment in the process. A well- studied training algorithm called the Baum-Welch algorithm [17] have described the use of an alternative HMM algorithm using gradient descent which seems equally effective. Both of these approaches find locally optimal alignments not global optima's and tends to get stuck in incorrect optima.

HMM-based multiple alignment is interestingly different other multiple alignment methods. In this the scoring parameters and alignments are initially unknown. Therefore, alignment does not require difficult priori choices for scoring parameters. This approach initially avoids the many to many multiple sequence alignment problem by recasting it to many to one sequence HMM alignment problem. Current HMM methods will outperform other multiple alignment algorithms in complicated cases involving many gaps and insertions [19].

C. HMM TO CLUSTER SEQUENCES AND DISCOVER SUBFAMILIES

When a relatively large number of sequences are present, it is sometimes possible to obtain good results by clustering these sequences according to similarity between them. Different HMM are trained for each cluster/subfamily. A simple extension of HMM enables us to use the EM (Expectation- maximizing) training algorithm to automatically partition the sequences into cluster of similar sequences. By iteratively splitting these clusters, we can build a phylogenetic tree in a "top-down" manner. Sometimes when the size of cluster becomes too small, it is hard to construct an accurate model due to insufficient number of sequences in each cluster. At this time we need to do some "bottom-up" processing.

V. RELATED WORK

Nakai and Kanehisa [20][11] initially developed a system for predicting protein subcellular locations using their N-terminal sorting signals. Later, a computational program based on the same approach, called PSORT [21], was presented. Several machine learning methods [22][23][24] have been proposed to detect these types of signals; the most popular one is SignalP [13]. Claros et al. [25] used neural networks for prediction. Method based on amino acid composition was proposed by Nakashima and Nishikawa [26]. From then onwards, many different approaches have been introduced to predict protein subcellular locations by amino acid composition or dipeptide composition. Reinhardt and Hunnard [27] used neural networks for predicting. Chou and Elrod [28] proposed a covariant discriminant algorithm. Yuan [29] constructed a Markov chain model using sequential data. Cedano et al. [30] gave a prediction program called ProtLock using Mahalanobis distance. Fujiwara and Asogawa [31] integrated hidden Markov model with neural network. Huang and Li [32] introduced a fuzzy k-nearest neighbor's algorithm. Furthermore Park and Kanehisa [33], Hua and Sun [34] adopted support vector machine (SVM). Gao et al. [35] used combined feature of sequences.

However, all the methods mentioned above might miss some information regarding sequence length or some sequence order. So in order to handle these problems new methods have been proposed in recent years. Chou [36] introduced a new concept of quasi-sequence order to reflect the sequence order effect based on the physic-chemical distance between amino acids, and a remarkable improvement was observed.in the prediction rate. Later, Chou [37] proposed a new concept called pseudo-amino acid composition. Chou and Cai [38] defined functional domain composition. There are other methods such as supervised locally linear embedding (SLLE) [39], complexity measure factor [40], LOCtarget [41], cellular automata [42], spectral analysis technique [43], Zp curve [44], lexical analysis [45], hybrid modules [46] and digital signal processing approach [47] were also suggested.

VI. DISCUSSION AND CONCLUSION

In this paper, various research papers and articles in the domain of prediction of subcellular sites have been studied. We have seen that a lot of work has been done in the field of bioinformatics. Many machine learning algorithms have been used to predict the protein subcellular sites. It has been observed that the performance of Hidden Markov model as compared to other machine learning techniques is higher if we have a sequential dataset in many cases. We can further improve the performance by extracting location signals and reducing noise. In case of non-sequential data, though HMM achieves lowest number of false negatives (highest specificity), it performs badly considering sensitivity. Therefore predicting entire sequence of a protein is much less sensitive than predicting with segments that are related with subcellular locations. We can also create a hybrid method in which HMM can be integrated with other machine learning technique in order to improve the predicting performance. For example Markov model tend to predict cytoplasmic proteins more accurately, while neural networks predict extracellular or other protein better. Properly combining all the methods should lead to more favourable predicting results like Fujiwara and Asogawa[31] integrated hidden Markov model with neural network.

REFERENCES

[1] Nakai, Kenta. "Protein sorting signals and prediction of subcellular localization."Advances in protein chemistry 54 (2000): 277-344.

[2] Baum, Leonard E., and Ted Petrie. "Statistical inference for probabilistic functions of finite state Markov chains." The annals of mathematical statistics(1966): 1554-1563.

[3] Rabiner, Lawrence. "A tutorial on hidden Markov models and selected applications in speech recognition." Proceedings of the IEEE 77.2 (1989): 257-286.

[4] Böhm, Siegfried, DmitrijFrishman, and H. Werner Mewes. "Variations of the C2H2 zinc finger motif in the yeast genome and classification of yeast zinc finger proteins." Nucleic acids research 25.12 (1997): 2464-2469.
[5] Frishman, Dmitrij, and Patrick Argos. "Seventy-five percent accuracy in protein secondary structure prediction." Proteins-Structure Function and Genetics 27.3 (1997): 329-335.
[6] Thompson, Michael J., and Richard A. Goldstein. "Predicting solvent accessibility: higher accuracy using Bayesian statistics and optimized residue substitution classes." (1996).
[7] Persson, Bengt, and Patrick Argos. "Prediction of transmembrane segments in proteins utilising multiple sequence alignments." Journal of molecular biology237.2 (1994): 182-192.
[8] Rost, Burkhard, PieroFariselli, and Rita Casadio. "Topology prediction for helical transmembrane proteins at 86% accuracy–Topology prediction at 86% accuracy." Protein Science 5.8 (1996): 1704-1718.
[9] Elofsson, Arne, and Gunnar von Heijne. "Membrane protein structure: prediction versus reality." Annu. Rev. Biochem. 76 (2007): 125-140.
[10] Pautsch, Alex, and Georg E. Schulz. "Structure of the outer membrane protein A transmembrane domain." Nature Structural & Molecular Biology 5.11 (1998): 1013-1017.
[11] Nakai, Kenta, and Minoru Kanehisa. "A knowledge base for predicting protein localization sites in eukaryotic cells." Genomics 14.4 (1992): 897-911.
[12] Yu CS, Lin CJ, Hwang JK. "Predicting subcellular localization of proteins for Gram-negative bacteria by support vector machines based on n-peptide compositions." Protein Science 2004, 13:1402-1406.
[13] DyrløvBendtsen, Jannick, et al. "Improved prediction of signal peptides: SignalP 3.0." Journal of molecular biology 340.4 (2004): 783-795.
[14] Blunsom, Phil. "Hidden markov models." Lecture notes, August 15 (2004): 18-19.
[15] Yu, Shun-Zheng, and Hisashi Kobayashi. "An efficient forward-backward algorithm for an explicit-duration hidden Markov model." Signal Processing Letters, IEEE 10.1 (2003): 11-14.
[16] Lou, Hui-Ling. "Implementing the Viterbi algorithm." Signal Processing Magazine, IEEE 12.5 (1995): 42-52.
[17] Welch, Lloyd R. "Hidden Markov models and the Baum-Welch algorithm." IEEE Information Theory Society Newsletter 53.4 (2003): 10-13.
[18] Churchill. "GA : Stochastic models for heterogeneous DNA sequences." Bull Math Biol 1989, 51: 79Ð94.
[19] Eddy, Sean R. "Multiple alignment using hidden Markov models." Ismb. Vol. 3. 1995.
[20] Nakai, Kenta, and Minoru Kanehisa. "Expert system for predicting protein localization sites in gram-negative bacteria." Proteins: Structure, Function, and Bioinformatics 11.2 (1991): 95-110.
[21] Nakai, Kenta, and Paul Horton. "PSORT: a program for detecting sorting signals in proteins and predicting their subcellular localization." Trends in biochemical sciences 24.1 (1999): 34-35.
[22] Nielsen, Henrik, et al. "Identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites." Protein engineering 10.1 (1997): 1-6.
[23] Nielsen, Henrik, and Anders Krogh. "Prediction of signal peptides and signal anchors by a hidden Markov model." Ismb. Vol. 6. 1998.
[24] Nielsen, Henrik, SørenBrunak, and Gunnar von Heijne. "Machine learning approaches for the prediction of signal peptides and other protein sorting signals." Protein engineering 12.1 (1999): 3-9.
[25] Claros, Manuel G., SørenBrunak, and Gunnar von Heijne. "Prediction of N-terminal protein sorting signals." Current opinion in structural biology 7.3 (1997): 394-398.
[26] Nakashima, Hiroshi, and Ken Nishikawa. "Discrimination of intracellular and extracellular proteins using amino acid composition and residue-pair frequencies." Journal of molecular biology 238.1 (1994): 54-61.
[27] Reinhardt, Astrid, and Tim Hubbard. "Using neural networks for prediction of the subcellular location of proteins." Nucleic acids research 26.9 (1998): 2230-2236.
[28] Chou, Kuo-Chen, and David W. Elrod. "Protein subcellular location prediction."Protein engineering 12.2 (1999): 107-118.
[29] Yuan, Zheng. "Prediction of protein subcellular locations using Markov chain models." FEBS letters 451.1 (1999): 23-26.
[30] Cedano, Juan, et al. "Relation between amino acid composition and cellular location of proteins." Journal of molecular biology 266.3 (1997): 594-600.
[31] Fujiwara, Yukiko, and Minoru Asogawa. "Prediction of subcellular localizations using amino acid composition and order." GENOME INFORMATICS SERIES(2001): 103-112.
[32] Huang, Ying, and Yanda Li. "Prediction of protein subcellular locations using fuzzy k-NN method." Bioinformatics 20.1 (2004): 21-28.
[33] Park, Keun-Joon, and Minoru Kanehisa. "Prediction of protein subcellular locations by support vector machines using compositions of amino acids and amino acid pairs." Bioinformatics 19.13 (2003): 1656-1663.
[34] Hua, Sujun, and Zhirong Sun. "Support vector machine approach for protein subcellular localization prediction." Bioinformatics 17.8 (2001): 721-728.
[35] Gao, Qing-Bin, et al. "Prediction of protein subcellular location using a combined feature of sequence." FEBS letters 579.16 (2005): 3444-3448.
[36] Chou, Kuo-Chen. "Prediction of protein subcellular locations by incorporating quasi-sequence-order effect." Biochemical and Biophysical Research Communications 278.2 (2000): 477-483.

[37] Chou, Kuo-Chen. "Prediction of protein cellular attributes using pseudo-amino acid composition." Proteins: Structure, Function, and Bioinformatics 43.3 (2001): 246-255.

[38] Chou, Kuo-Chen, and Yu-Dong Cai. "Using functional domain composition and support vector machines for prediction of protein subcellular location." Journal of Biological Chemistry 277.48 (2002): 45765-45769.

[39] Wang, Meng, et al. "SLLE for predicting membrane protein types." Journal of Theoretical Biology 232.1 (2005): 7-15.

[40] Xiao, X., et al. "Using complexity measure factor to predict protein subcellular location." Amino acids 28.1 (2005): 57-61.

[41] Nair, Rajesh, and BurkhardRost. "LOCnet and LOCtarget: sub-cellular localization for structural genomics targets." Nucleic acids research 32.suppl 2 (2004): W517-W521.

[42] Xiao, X., et al. "Using cellular automata to generate image representation for biological sequences." Amino Acids 28.1 (2005): 29-35.

[43] Wang, Meng, et al. "Weighted-support vector machines for predicting membrane protein types based on pseudo-
amino acid composition." Protein Engineering Design and Selection 17.6 (2004): 509-516.

[44] Feng, Zhi-Peng. "An overview on predicting the subcellular location of a protein."In silico biology 2.3 (2002): 291-303.

[45] Nair, Rajesh, and BurkhardRost. "Inferring sub-cellular localization through automated lexical analysis." Bioinformatics 18.suppl 1 (2002): S78-S86.

[46] P. S. Raghava. "ESLpred: SVM-based method for subcellular localization of eukaryotic proteins using dipeptide composition and PSI-BLAST." Nucleic acids research 32.suppl 2 (2004): W414-W419.

[47] Pan, Yu-Xi, et al. "Predicting protein subcellular location using digital signal processing." ActabiochimicaetbiophysicaSinica 37.2 (2005): 88-96.

