RESEARCH ARTICLE

GNBSL: A new integrative system to predict the subcellular location for Gram-negative bacteria proteins

Jian Guo[1], Yuanlie Lin[1] and Xiangjun Liu[2]

[1] Department of Mathematical Sciences, Laboratory of Statistical Computing & Bioinformatics, Tsinghua University, Beijing, P. R. China
[2] Bioinformatics Laboratory, School of Medicine, Tsinghua University, Beijing, P. R. China

This paper proposes a new integrative system (GNBSL – Gram-negative bacteria subcellular localization) for subcellular localization specified on the Gram-negative bacteria proteins. First, the system generates a position-specific frequency matrix (PSFM) and a position-specific scoring matrix (PSSM) for each protein sequence by searching the Swiss-Prot database. Then different features are extracted by four modules from the PSFM and the PSSM. The features include whole-sequence amino acid composition, N- and C-terminus amino acid composition, dipeptide composition, and segment composition. Four probabilistic neural network (PNN) classifiers are used to classify these modules. To further improve the performance, two modules trained by support vector machine (SVM) are added in this system. One module extracts the residue-couple distribution from the amino acid sequence and the other module applies a pairwise profile alignment kernel to measure the local similarity between every two sequences. Finally, an additional SVM is used to fuse the outputs from the six modules. Test on a benchmark dataset shows that the overall success rate of GNBSL is higher than those of PSORT-B, CELLO, and PSLpred. A web server GNBSL can be visited from http://166.111.24.5/webtools/GNBSL/index.htm.

Received: February 8, 2006
Revised: June 5, 2006
Accepted: June 7, 2006

Keywords:
Pairwise profile alignment / Position-specific frequency matrix / Position-specific scoring matrix / PSI-BLAST / Subcellular localization

1 Introduction

Subcellular location is a key functional characteristic of potential gene products such as proteins. At present, a number of subcellular localization methods have been introduced. These methods can be grouped into three categories. One category is based on the existence of N-terminal sorting signals [1] such as signal peptides, mitochondrial targeting peptides, and chloroplast transit peptides [2, 3]. Emanuelsson *et al.* [4] proposes an integrative system based on individual sorting signal predictions. This system can be used to find cleavage sites in sorting signals and simulate the real sorting process to a certain extent. Nevertheless, the prediction accuracy of the methods in this is highly dependent on the quality of the N-terminal sequence assignment. The second category studies those whole-sequence features such as amino acid composition [5–10], pseudoamino acid composition [11–18, 19–25], dipeptide composition [26, 27], residue-couple composition [28, 29], Fourier transform feature [30], cellular automata image [19], physical and chemical properties [31, 32], functional domain [21–23, 33–42], and n-Gram [43]. The third category integrates different features to improve the

Correspondence: Jian Guo, Department of Mathematical Sciences, Laboratory of Statistical Computing & Bioinformatics, Tsinghua University, Beijing 100084, P. R. China
E-mail: genovo@126.com
Fax: +86-10-62786651

Abbreviations: MCC, Matthew's correlation coefficient; **PNN**, probabilistic neural network; **PSFM**, position-specific frequency matrix; **PSSM**, position-specific scoring matrix; **SVM**, support vector machine

performance and robustness. For example, PSORT-B [44] integrated amino acid composition, similarity to proteins of known localization, presence of a signal peptide, transmembrane alpha-helices and motifs corresponding to specific localizations. Bhasin and coworkers [45–47] developed other kinds of synthesis methods which infused amino acid composition, composition of physicochemical properties, dipeptide composition, residue couples, and EuPSI-BLAST.

This paper proposes a new integrative system, GNBSL (Gram-negative Bacteria Subcellular Localization), to predict the protein sucbellular location for Gram-negative bacteria, which has been studied by Gardy *et al.* [44], Yu *et al.* [43], and Bhasin *et al.* [47]. Different from PSORT-B [44], CELLO [43], and PSIPRED [47], GNBSL extracts features from both sequences and profiles. First, it generates a position-specific frequency matrix (PSFM) and a position-specific scoring matrix (PSSM) for each protein sequence by searching the Swiss-Prot database. Then a number of features are extracted from the PSFM and the PSSM. The features include whole-sequence amino acid composition, N- and C-terminus amino acid composition, dipeptide composition, and segment composition. To further improve the performance, two other modules are added in this system. One module extracts the residue-couple features from the protein sequence; the other module uses a local pairwise profile alignment kernel to train a support vector machine (SVM) classifier. The probabilistic neural network (PNN) and SVM are applied for classification in different modules. Finally, an additional SVM is employed to fuse the results from different modules and output the final decision. We tested the performance of GNBSL on a benchmark dataset and compared the results from different modules and from existing methods. The webserver can be visited at http://166.111.24.5/webtools/GNBSL/index.htm.

2 Materials and methods

2.1 Datasets

In this work, Gardy *et al.* [44] Gram-negative bacteria protein dataset was used as a benchmark dataset to test the performance of GNBSL. This dataset includes 1302 proteins distributed in five subcellular locations: cytoplasm, inner-membrane, outer-membrane, periplasm, and extracellular. This dataset is convenient for comparing our work with existing methods because it is also used by PSORT-B, CELLO, and PSLpred.

2.2 Features and modules

In this paper, each protein sequence is used as a seed to search the Swiss-Prot 46.0 protein database to find out the homogenous sequences using PSI-BLAST program [48] and generates two profiles: PSSM and PSFM. Both PSSM and PSFM are matrices with 20 rows and L columns. The elements of PSSM profiles represent the log-likelihood of the residue substitutions at all positions in the template (query sequence) while PSFM contains both the sequence-weighted observed frequency as well as the pseudocounts derived from the substitution matrix. In the following step, we will construct five modules to extract different features from PSSM and PSFM and an additional module to extract residue-couple features from the amino acid sequence.

2.2.1 Module 1

The first module extracts amino acid composition from PSSM. Denote as the PSSM matrix of the protein sequence, where the elements of $M = (a_{x,y})_{20xL}$ as the PSSM matrix of the protein sequence, where a_{ij} the alements of M and denote $V = (v_1, v_2, \ldots, v_{20})$ as a 20-dimensional vector representing the occurrence frequency from 20 types of amino acids. The components of the vector, v_i, can be calculated as follows:

$$v_{\mathrm{i}} = \frac{\sum_{\mathrm{j}=1}^{\mathrm{L}} a_{\mathrm{ij}}}{L},\ i = 1, 2, \ldots, 20 \tag{1}$$

Obviously, v_i is the mean value of the elements in the ith row of M and V are the feature vectors representing the amino acid composition of the entire PSSM.

2.2.2 Module 2

Unsually, the N- and the C-terminus of the protein contain important signal peptides, which determine the subcellular location of the protein. It is not a easy thing to directly identify these signal peptides from the sequence. Instead, this module calculates the amino acid composition from the whole PSSM, the N-terminus of PSSM and the C-terminus of PSSM. For each part, a 20-D vector is extracted using the same method as module 1, so the feature vector of module 2 has 60 dimensions.

2.2.3 Module 3

This module extracts dipeptide composition from PSFM. A dipeptide comprises two consecutive residues. Obviously, there are totally $20 \times 20 = 400$ possible types of dipeptide. Therefore, the feature vector in this module is set to 400 dimensions, corresponding to the 400 possible dipeptide types. Denote by A_1 the front residue of the dipeptide and by A_2 the back residue of the dipeptide, where both A_1 and A_2 represent to 20 different amino acid types (denoted by numbers 1 to 20). Denote $N = (b_{x,y})_{20xL}$ as the PSSM. Then the occurrence frequency of the dipeptide A_1A_2 in the sequence can be calculated as

$$f_{\mathrm{i,j}} = P(A_1 = i, A_2 = j) = \frac{1}{L-1}\sum_{\mathrm{s}=1}^{\mathrm{L}-1} b_{\mathrm{i,s}} \times b_{\mathrm{j,s}} \tag{2}$$

where $i = 1, \ldots 20$, $j = 1, \ldots, 20$; L is the length of the sequence.

2.2.4 Module 4

This module assumes that different segments of a sequence can provide complementary information. It divides the query sequence into several fragments with equal length and calculates the amino acid composition from the corresponding fragments of the PSFM. Denote by S_i the ith segment, i = 1,2,. . .,m, where m is the number of all fragments. The amino acid composition (20-D vector) of S_i $(I = 1,2,. . .,m)$ is calculated and all the 20-D vectors from different segments are concatenated to the feature vector.

2.2.5 Module 5

In this module , a local profile alignment kernel is designed to train an SVM classifier. The local profile alignment algorithm [49] is an extension of the Smith–Waterman local sequence alignment algorithm. Unlike the latter one, local profile alignment attempts to find the local similarity between two profiles. It shares the same dynamic programming process as the Smith–Waterman algorithm but uses a different scoring method for aligned pairs. In Smith–Waterman algorithm, the score of a pair of aligned amino acids is defined by the substitution matrix, such as BLOSUM62, PAM50, *etc*. In the local profile alignment algorithm, the score should be calculated from two aligned 20-D vectors (the column of PSFM, representing the distribution of amino acids in a specific site of the sequence). Let us denote $M_1 = (m^1_{\mathrm{i,j}})_{20\times \mathrm{L}_1}$ and $M_2 = (m^2_{\mathrm{i,j}})_{20\times \mathrm{L}_2}$ as two PSFMs to be aligned, where the column number of the two matrices are L_1 and L_2, respectively. Denote $C^{\mathrm{u}}_1 = (m^1_{1,\mathrm{u}}, m^1_{2,\mathrm{u}}, \ldots, m^1_{20,\mathrm{u}})^{\mathrm{T}}$ as the column u of M_1 and $C^{\mathrm{v}}_1 = (m^2_{1,\mathrm{v}}, m^2_{2,\mathrm{v}}, \ldots, m^2_{20,\mathrm{v}})^{\mathrm{T}}$ as the column v of M_2, respectively. Then the pairwise profile alignment score of C^{u}_1 and C^{v}_2 is defined as

$$\varepsilon(C^{\mathrm{u}}_1, C^{\mathrm{v}}_2) = \sum_{\mathrm{h}=1}^{20}\sum_{\mathrm{k}=1}^{20} m^1_{\mathrm{h,u}} \cdot m^2_{\mathrm{k,v}} \cdot \tau(h,k)$$

where $\tau(h, k)$ is the h, k element of a specific substitution matrix. A dynamic programming algorithm is employed to trace back the optimal local alignment. The local pairwise profile alignment score of two PSFMs, $\rho(M_1, M_2)$, is defined as the sum of the scores from each pair of aligned columns. Then a pairwise alignment kernel for SVM is defined as

$$K(M_1, M_2) = \left(1 + \frac{\varrho(M_1, M_2)}{\sqrt{\varrho(M_1, M_1)}\sqrt{\varrho(M_2, M_2)}}\right)^{\mathrm{d}}$$

where d is a parameter. This kernel can be regarded as an extension of the traditional polynomial kernel by replacing the inner product with the local pairwise profile alignment score. An SVM classifier is trained using the above kernel on the training set to obtain the support vectors, their corresponding weights α and the bias b. For a binary classification problem, a protein sequence S can be predicted by the following decision formula

$$f(M_{\mathrm{S}}) = \sum_{\mathrm{i}\in \mathrm{SV}} y_{\mathrm{i}}\alpha_{\mathrm{i}} K(M_{\mathrm{S}}, M_{\mathrm{i}}) + b$$

where SV is the set of all support vectors, y_i is the label of the ith support vector and M_S is the PSFM of S. For the subcellular localization problem which is a multi-classification problem, the one-vs-rest strategy is applied in this work.

2.2.6 Module 6

This module applies the residue-couple model [28] to extract the distribution of amino acid pairs from the sequence. A term "Rank" is defined to describe the distance between two amino acids in a pair. Rank = 1 represents the pair is composed of two consecutive amino acids; Rank = 2 represents the two amino acids in a pair are separated by a residue between them; Rank = 3 represents there are two residues inserted in the pair; and so on. For each rank, a 400-D vector is calculated to record the distribution of the amino acid pairs in the sequence. The final feature has $400 \times m$ dimensions, where m is the total number of ranks.

2.2.7 SVM fusion

Among the six modules, module 6 extracts the residue-couple features from the amino acid sequences; modules 1, 2, and 4 extract features from the PSSM while modules 3 and 5 extract features from the PSFM. Modules 1–4 are trained with the PNN classifiers and other two modules are trained with the SVM classifiers. The prediction results from the six modules are fused by another SVM classifier. Specifically, the output value from these modules are encoded to six 5-D sparse binary vectors and are concatenated to a 30-D vector which is input to an SVM classifier for the final decision. The prediction process of GNBSL is illustrated in Fig. 1.

2.3 Assessment of prediction results

The leave-one-out crossvalidation (jackknife) test, which is regarded as more rigorous than the widely used k-fold crossvalidation [11–13], is employed to evaluate the performance of GNBSL. During the process of a leave-one-out crossvalidation test, each protein is singled out in turn for testing and the remaining proteins are merged for training.

The overall accuracy, the accuracy in each location and Matthew's Correlation Coefficient (MCC) are used to assess the prediction result. Please refer to the Supplementary Materials for more details.

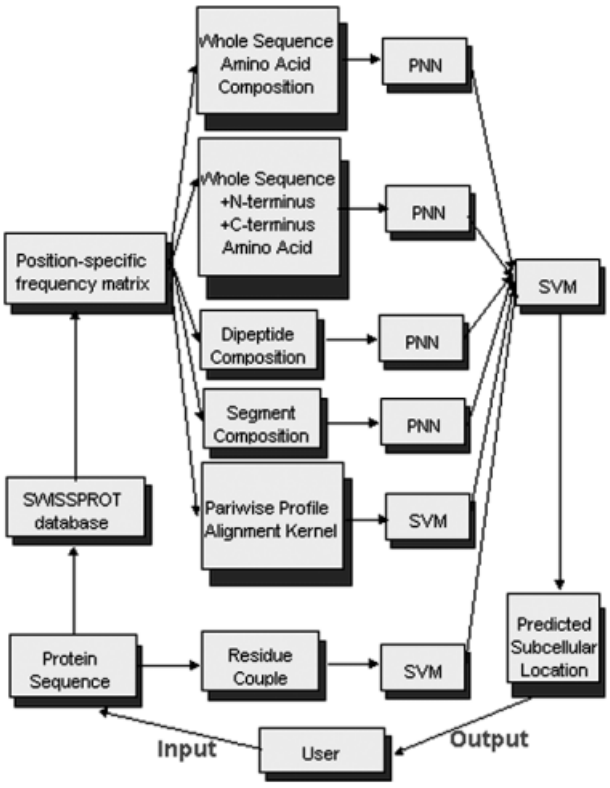

Figure 1. The prediction process of GNBSL.

3 Results

3.1 Prediction results and comparison

The optimized parameters of each module are listed in Table 1 and the results from the leave-one-out crossvalidation test are listed in Table 2. Module 1 extracts amino acid composition from PSSM and reaches an overall accuracy of 88.5%. By combining with the amino acid compositions from the N- and C-terminus of the PSSM, module 2 reaches a better prediction accuracy of 90.7%. Module 3 extracts the dipeptide compositions from the PSFM but the overall accuracy is almost the same as module 1, which only considers the amino acid composition from PSSM. Module 4 extracts amino acid composition from PSSM segments with equal length (number of columns of PSSM) and reaches an overall accuracy of 89.2%, slightly better than modules 1 and 3. Instead of extracting the feature directly from the profiles, module 5 generates the kernel matrix for SVM by calculating the similarity score using the local profile alignment algorithm. The prediction results show that this method performs slightly worse than the other five modules. The last module extracts residue-couple features from the amino acid sequence and performs comparable to modules 1 and 3. Table 2 shows that these modules have different advantages on different subcellular locations. For example, module 2 performs best on cytoplasm and module 3 performs best on the outer-membrane. To utilize the complementary information extracted by these modules, an additional SVM is used to fuse the output from the six modules. The overall accuracy of the SVM fusion reaches 93.4%, which is significantly better than that of each individual module.

Table 1. Optimized parameters for each module and for SVM fusion

	Profile type	Classifier type	Parameters
Module 1	PSSM	PNN	$\sigma = 0.035$
Module 2	PSSM	PNN	$\sigma = 0.057$ $L_N = 50$ $L_C = 50$
Module 3	PSFM	PNN	$\sigma = 0.08$
Module 4	PSSM	PNN	$\sigma = 0.055$ $N_{split} = 2$
Module 5	PSFM	SVM	$\gamma = 17$ $C = 0.0006$
Module 6	–	SVM	$\gamma = 7$ $C = 100$ $N_{rank} = 10$
Fusion	–	SVM	$\gamma = 0.08$ $C = 50$

σ is the parameter of PNN, γ is the parameter of RBF kernel for SVM, C is the regularization parameter of soft-margin SVM (formula 5), L_N and L_C are the number of N- and C-terminal residues extracted by module 2, respectively, N_{split} is the number of fragments in module 4, and N_{rank} is the rank number in module 6. Please refer to the Supplementary Materials for more details.

The prediction result of GNBSL (SVM fusion) is compared with that of PSORT-B [44], CELLO [44], and PSLpred [47] on the same dataset. The characteristic of GNBSL is that its modules (modules 1–5) extract information from profiles rather than from an amino acid sequence. The comparison results of the four methods are listed in Table 3. The overall accuracy of GNBSL reaches 93.4%, which is 2.2, 4.5, and 18.6% higher than PSLpred, CELLO, and PSORT-B, respectively. In addition, GNBSL performs better that PSLPred on the cytoplasm and inner-membrane and performs comparably on the periplasm, outer-membrane and extracellular.

3.2 Negative control test

We did a negative control test for each module. First, the sequence of each test protein is shuffled and the PSSM and the PSFM of the shuffled sequence are regenerated by PSI-BLAST. The extracted feature vector (modules 1–4 and 6) or kernel value (module 5) is classified by the pretrained classifiers. The leave-one-out crossvalidation is also used to do the negative control test by replacing the samples in the old test set by the new shuffled samples. The test results for the six modules are listed in Table 4. The prediction results of each modulesignificantly fall when test proteins are shuffled.

4 Description of webserver

All the modules introduced in this paper have been implemented in a webserver named "GNBSL" and can be accessed from http://166.111.24.5/webtools/GNBSL/index.htm. All

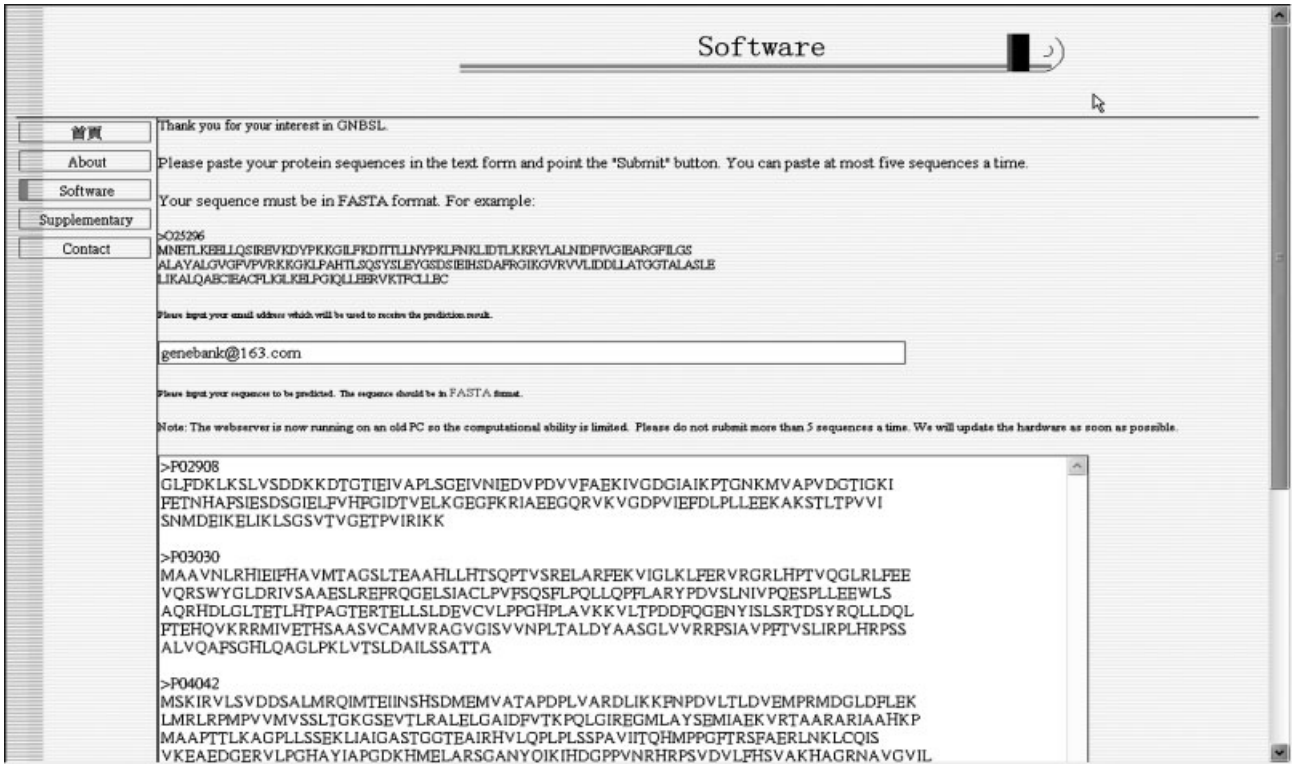

Figure 2. A snapshot of the webpage of GNBSL for sequence submission.

Table 2. The performance of six modules and SVM fusion of GNBSL on the Gram-negtative bacteria protein dataset

	Module 1		Module 2		Module 3		Module 4		Module 5		Module 6		SVMFusion	
	AC	MC	AC	MC	AC	MC	AC	MC	AC	MC	AC	MC	AC	MC
Cyt	83.5	0.82	94.8	0.95	84.7	0.81	85.9	0.84	75.8	0.78	91.0	0.84	95.2	0.90
Inn	89.2	0.89	88.4	0.88	88.1	0.88	89.6	0.91	90.3	0.83	88.8	0.91	92.3	0.94
Per	85.3	0.78	88.9	0.89	82.4	0.77	89.8	0.81	77.1	0.81	85.3	0.78	91.0	0.87
Out	93.5	0.93	95.5	0.96	96.6	0.93	92.6	0.92	97.7	0.89	93.5	0.91	97.2	0.95
Ext	88.4	0.84	83.2	0.83	86.8	0.87	85.8	0.84	88.4	0.86	80.0	0.84	87.9	0.90
OA	88.5	–	90.7	–	88.6	–	89.2	–	86.8	–	88.5	–	93.4	–

Cyt: cytoplasm; Inn: inner-membrane; Per: periplasmic; Out: outer-membrane; Ext: extracellular; OA: overall accuracy; AC: accuracy in a certain location; MC: MCC in a certain location. The value of AC is represented in percentage.

Table 3. Compare the performance of GNBSL with other three subcellular localization methods on the Gram-negative bacteria protein dataset

	PSORT-B	CELLO		PSLpred		GNBSL	
	AC	AC	MCC	AC	MCC	AC	MCC
Cyto	69.4	90.7	0.85	90.7	0.86	95.2	0.90
Inn	70.0	78.9	0.82	86.8	0.88	92.3	0.94
Per	78.7	88.4	0.92	90.3	0.90	91.0	0.87
Outer	90.3	94.6	0.90	95.2	0.95	97.2	0.95
Exta	57.6	86.9	0.80	90.6	0.84	87.9	0.90
OA	74.8	88.9	–	91.2	0.89	93.4	–

Notations: See the legend to Table 2.

the CGI scripts are written in matlab and run on a matlab webserver program. The PSI-BLAST program is downloaded from http://www.ncbi.nlm.hig.gov/blast and is called by a matlab script. The SVM is implemented by a matlab machine learning toolbox SPIDER downloaded from http://www.kyb.tuebingen.mpg.de/bs/people/spider/. The user can submit their sequence in FASTA format and the prediction results can be retrieved by two ways: accepting an e-mail automatically sent by the server or downloading the results from our FTP sites: ftp://166.111.24.5/ (the username, password, and port will be provided on the webpage after you submit your sequences).

Table 4. The prediction results from the negative control test for the six modules

	Module 1		Module 2		Module 3		Module 4		Module 5		Module 6	
	AC	MC	AC	MC	AC	MC	AC	MC	AC	MC	AC	MC
Cyt	51.2	0.53	68.6	0.39	69.8	0.61	51.2	0.48	38.3	0.45	85.5	0.75
Inn	76.9	0.72	65.3	0.66	76.9	0.72	78.0	0.78	84.0	0.51	85.8	0.84
Per	74.6	0.47	37.3	0.27	66.0	0.58	79.1	0.45	19.3	0.36	71.7	0.62
Out	29.0	0.39	29.3	0.32	77.6	0.72	15.6	0.25	66.5	0.52	73.9	0.69
Ext	65.8	0.44	40.5	0.28	65.8	0.62	64.2	0.44	53.2	0.41	56.8	0.56
OA	57.0	–	47.3	–	72.0	–	54.2	–	53.9	–	75.7	–

Notations: See the legend to Table 2.

5 Conclusions

This paper introduces an integrative method for protein subcellular localizations for Gram-negative bacteria. Five modules are constructed by extracting information from the profiles and an additional module is used to extract residue-couple distributions from the amino acid sequence. The outputs of the six modules are fused by an SVM classifier for the final decision. On a benchmark dataset, the performance of GNBSL has been demonstrated to be comparable or better than the existing methods. A web service is also available from http://166.111.24.5/webtools/GNBSL/index.htm.

We thank Dr. A. Reinhardt and Dr. Y. Huang for sharing their eukaryotic protein dataset. We also thank Mr. Ying Liu who helped in setting up the webserver. This work was supported by Human Liver Proteome Project (2004BA711A21) and The National Nature Science Foundation of China (10371063).

6 References

[1] Nakai, K., *Adv. Protein Chem.* 2000, *54*, 277–344.

[2] Nielsen, H., Engelbrecht, J., Brunak, S., von Heijne, G., *Int. J. Neural Sys.* 1997, *8*, 581–599.

[3] Nielsen, H., Brunak, S., von Heijne, G., *Protein Eng.* 1999, *12*, 3–9.

[4] Emanuelsson, O., Nielsen, H., Brunak, S., von Heijne, G., *J. Mol. Boil.* 2000, *300*, 1005–1016.

[5] Nakashima, H., Nishikawa, K., *J. Mol. Boil.* 1994, *238*, 54–61.

[6] Cedano, J., Aloy, P., Perez-Pons, J. A., Querol, E., *J. Mol. Boil.* 1997, *266*, 594–600.

[7] Reinhardt, A., Hubbard, T., *Nucleic Acids Res.* 1998, *26*, 2230–2236.

[8] Chou, K. C., Elord, D., *Protein Eng.* 1999, *12*, 107–118.

[9] Hua, S. J., Sun, Z. R., *Bioinformatics* 2001, *17*, 721–728.

[10] Guo, J., Lin, Y. L., Sun, Z. R., *Proc. APBC* 2004, 21–27.

[11] Chou, K. C., Zhang, C. T., *Crit. Rev. Biochem. Mol. Biol.* 1995, *30*, 275–349.

[12] Zhou, G. P., *J. Protein Chem.* 1998, *17*, 729–738.

[13] Zhou, G. P., Assa-Munt, N., *Proteins: Struct., Funct., Genet.* 2001, *44*, 57–59.

[14] Chou, K. C., *Proteins: Struct., Funct., Genet.* 2001, *43*, 246–255. (Erratum, ibid. 2001, Vol. 44, 60).

[15] Feng, Z. P., *Biopolymers* 2001, *58*, 491–499.

[16] Feng, Z. P., *In Silico Biol.* 2002, *2*, 291–303.

[17] Zhou, G. P., Doctor, K., *Proteins: Struct., Funct., Genet.* 2003, *50*, 44–48.

[18] Shen, H. B., Chou, K. C., *Biochem. Biophys. Res. Commun.* 2005, *337*, 752–756.

[19] Xiao, X., Shao, S. H., Ding, Y. S., Huang, Z. D, Chou, K. C., *Amino Acids* 2006, *30*, 49–54.

[20] Gao, Y., Shao, S. H., Xiao, X., Ding, Y. S. *et al.*, *Amino Acids* 2005, *28*, 373–376.

[21] Chou, K. C., Cai, Y. D., *J. Cell. Biochem.* 2004, *91*, 1197–1203.

[22] Chou, K. C., Cai, Y. D., *Bioinformatics* 2005, *21*, 944–950.

[23] Cai, Y. D., Chou, K. C., *Bioinformatics* 2004, *20*, 1151–1156.

[24] Wang, M., Yang, J., Xu, Z. J., Chou, K. C., *J. Theor. Biol.* 2005, *232*, 7–15.

[25] Shen, H. B., Chou, K. C., *Biochem. Biophys. Res. Commun.* 2005, *334*, 288–292.

[26] Yuan, Z., *FEBS Lett.* 1999, *451*, 23–26.

[27] Huang, Y., Li, Y. D., *Bioinformatics* 2004, *20*, 21–28.

[28] Guo, J., Lin, Y. L., Sun, Z. R., *Proc. APBC* 2005, 117–129.

[29] Park, K. J., Kanehisa, M., *Bioinformatics* 2004, *19*, 1656–1663.

[30] Liu, H., Wang, M., Chou, K. C., *Biochem. Biophys. Res. Commun.* 2005, *336*, 737–739.

[31] Chou, K. C., *Biochem. Biophys. Res. Commun.* 2001, *278*, 477–483.

[32] Feng, Z., Zhang, C. T., *Int. J. Biol. Macromol.* 2001, *28*, 225–261.

[33] Chou, K. C., Cai, Y. D., *J. Biol. Chem.* 2002, *277*, 45765–45769.

[34] Chou, K. C., Cai, Y. D., *Biochem. Biophys. Res. Commun.* 2003, *311*, 743–747.

[35] Chou, K. C., Cai, Y. D., *J. Cell. Biochem.* 2003, *90*, 1250–1260. (Addendum, ibid. 2004, 91, No.5, P.1085).

[36] Chou, K. C., Cai, Y. D., *Biochem. Biophys. Res. Commun.* 2004, *320*, 1236–1239.

[37] Chou, K. C, Cai, Y. D., *Biochem. Biophys. Res. Commun.* 2004, *321*, 1007–1009. (Corrigendum, ibid., 2005, Vol. 329, 1362).

[38] Chou, K. C, Cai, Y. D., *Biochem. Biophys. Res. Commun.* 2006, *339*, 1015–1020.

[39] Chou, K. C., Cai, Y. D., *Protein Sci.* 2004, *13*, 2857–2863.

[40] Chou, K. C., Cai, Y. D., *Biochem. Biophys. Res. Commun.* 2005, *327*, 845–847.

[41] Xiao, X., Shao, S., Ding, Y., Huang, Z. *et al.*, *Amino Acids* 2005, *28*, 57–61.

[42] Cai, Y. D., Zhou, G. P., Chou, K. C., *Biophys. J.* 2003, *84*, 3257–3263.

[43] Yu, C. S, Lin, C. J., Hwang, J. K., *Protein Sci.* 2004, *13*, 1402–1406.

[44] Gardy, J. L., Spencer, C., Wang, K., Ester, M. *et al.*, *Nucleic Acids Res.* 2003, *31*, 3613–3617.

[45] Bhasin, M., Raghava, G. P. S., *Nucleic Acids Res.* 2004, *32*, W414–W419.

[46] Garg, A., Bhasin, M., Raghava, G. P. S., *J. Biol. Chem.* 2005, *280*, 14427–14432.

[47] Bhasin, M., Garg, A., Raghava, G. P. S., *Bioinformatics* 2005, *21*, 2522–2524.

[48] Altschul, S. F., Madden, T. L., Schaffer, A. A., Zhang, J. *et al.*, *Nucleic Acids Res.* 1997, *25*, 3389–3402.

[49] Mittelman, D., Sadreyve, R., Grishin, N., *Bioinformatics* 2003, *19*, 1531–1539.

