國立交通大學

生物資訊及系統生物研究所

博士論文

蛋白質細胞定位及核糖核酸結合點之預測

Prediction of Subcellular Localization and RNA-binding Sites in Proteins

研究生	：	蘇家玉
Student	：	Chia-Yu Su
指導教授	：	許聞廉 教授
		黃鎮剛 教授
Advisors	：	Prof. Wen-Lian Hsu
		Prof. Jenn-Kang Hwang

中華民國九十八年二月

蛋白質細胞定位及核糖核酸結合點之預測

Prediction of Subcellular Localization and RNA-binding Sites in Proteins

研究生：蘇家玉 Student： Chia-Yu Su

指導教授：許聞廉 Advisors：Wen-Lian Hsu

黃鎮剛 Jenn-Kang Hwang

國立交通大學

生物資訊及系統生物研究所

博士論文

A Dissertation

Submitted to Institute of Bioinformatics and Systems Biology

College of Biological Science and Technology

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Ph.D.

in

Bioinformatics

February 2009

Hsinchu, Taiwan, Republic of China

中華民國九十八年二月

中文摘要

近年來隨著後基因體時代的來臨，生物資料庫中逐漸累積了許多待以分析的蛋白質序列。於是，如何自動地來分析和註解蛋白質的功能，已經在生物研究上扮演一個不可或缺的角色。在這當中，蛋白質細胞定位及核糖核酸結合點之相關研究，對於功能分析、基因體標註和藥物標靶發現是非常重要的。然而，利用傳統實驗方法來決定蛋白質細胞定位或結構非常昂貴又耗時，因此利用計算方法來分析和預測蛋白質功能，已經在蛋白質研究上成為一個非常重要的課題。

在蛋白質細胞定位預測中，我們發展出兩套不同的方法，PSL101 和 PSLDoc。PSL101 是根據細菌轉位路徑來擷取與細胞區室相關的生物特徵，再整合結構同源方法和支持向量機模型，來預測蛋白質在細胞中座落的位置。PSLDoc 則是先將蛋白質序列以間隙二肽的方法表示，結合位置加權矩陣的演化資訊後，利用機率式潛在語意分析模型來找出序列特徵，最後以支持向量機預測序列在細胞中的位置。我們提出的兩個方法中，在革蘭氏陰性菌的蛋白質細胞定位預測皆達到93%的整體準確率；對於低同源性的資料集，準確率更是比目前最好的結果提升7.4%。實驗結果證實，無論從轉位路徑擷取來的生物特徵，或是藉由文件分類技巧發展出的特徵精簡，皆能顯著地提高預測準確率。此外，我們所提出的生物特徵和間隙二肽標誌特徵，皆屬於可解釋的生物特徵，這些特徵可提供生物學家在進一步的研究和實驗設計上做為參考。

在核糖核酸結合點預測方面，我們提出 RNAProB 這方法來預測蛋白質序列上的核糖核酸結合點。我們針對傳統的位置加權矩陣提出一個新的平滑編碼設計，並利用支持向量機來預測蛋白質序列上的核糖核酸結合點。我們提出的位置加權矩陣之平滑編碼設計中，最大的特點在於考慮了蛋白質序列裡，每個氨基酸鄰近殘基的交互作用和關聯性。實驗結果顯示平滑編碼設計能夠顯著地提高預測準確率，尤其在敏感性的提升更為顯著。在目前較佳的預測方法中，我們所提出的方法較其他方法在整體準確率、敏感性、特異性和馬修斯相關係數上，分別提高了4.90%~6.83%，7.05%~26.90%，0.88%~5.33%和 0.10~0.23。實驗結果支持了我們所提出的這個假設：平滑編碼設計考慮了鄰近氨基酸之間的關聯性，因此能更準確地分辨出和核糖核酸有交互作用的殘基和沒有交互作用的殘基之間的歧異性。

基於我們所提出方法所具有的普遍性，將可以廣泛地延伸應用在其他生物資訊的研究上。此外，我們方法所預測的蛋白質細胞定位和核糖核酸結合點資訊，能夠幫助生物學家推論蛋白質功能和發現合適的藥物標靶；因此，我們深信文中所提出的高通量蛋白質體分析研究，將對科學發現有所貢獻。

Abstract

Automated function annotation is a major goal of post-genomic era with tremendous amount of protein sequences in the databases. Prediction of subcellular localization or binding sites in proteins is crucial for function analysis, genome annotation, and drug discovery. Determination of localization or structure using experimental approaches is time-consuming; thus, computational approaches become highly desirable.

We proposed two protein subcellular localization prediction methods, PSL101 and PSLDoc. PSL101 combines a structural homology approach and a support vector machine model, in which compartment-specific biological features derived from bacterial translocation pathways are incorporated. PSLDoc uses a probabilistic latent semantic analysis on gapped-dipeptides of various distances, where evolutionary information from position specific scoring matrix (PSSM) is utilized. Our methods achieve 93% in overall accuracy for Gram-negative bacteria, and compared favorably to the state-of-the-art results by 7.4% on a benchmark dataset having low homology to the training set. Experiment results demonstrate that both biological features derived from translocation pathways and feature reduction by document classification techniques can lead to a significant improvement in the prediction performance. Moreover, the proposed biological features and gapped-dipeptide signatures are interpretable and can be applied in advanced studies and experiment designs.

For RNA-binding site prediction, we propose another method, RNAProB, which incorporates a new smoothed PSSM encoding scheme in a support vector machine model. The proposed smoothed PSSM encoding considers correlation and dependency from neighboring residues for each amino acid in a protein sequence. Experiment results show that smoothed PSSM encoding significantly enhances the prediction performance, especially for sensitivity. Our method performs better than the state-of-the-art systems by 4.90%~6.83%, 7.05%~26.90%, 0.88%~5.33%, and 0.10~0.23 in terms of overall accuracy, sensitivity, specificity, and Matthew’s correlation coefficient, respectively. This also supports our assumption that smoothed PSSM encoding can better resolve the ambiguity in discriminating between interacting and non-interacting residues by modeling the dependency from surrounding residues.

Because of the generality of the proposed methods, they can be extended to other research topics in the future. Moreover, the information from predicted localization and structure of proteins can be used collectively to assist biologists in both inferring protein function and finding suitable drug targets. Therefore, we believe that our work can contribute to scientific discoveries on a high-throughput basis.

Acknowledgement

憶起過去的博士班生涯，對於我的指導教授：許聞廉老師、宋定懿老師和黃鎮剛老師，我的心中充滿無限感激。從碩士班的資訊科學跨足到博士班的跨領域生物資訊研究，在老師們耐心指導和諄諄教誨下，我逐漸成長並順利完成學業。特別要感謝許聞廉老師在博士班生涯給予的教導和協助，不論是對於學術研究的熱忱，或是做人處事的態度，老師都是我學習的榜樣；此外，老師也總是盡其所能提供學生良好的學習環境，並鼓勵我們參加國際會議拓展視野。更感謝口試委員許聞廉老師、宋定懿老師、黃鎮剛老師、何信瑩老師和楊進木老師，在口試時提供許多寶貴的建議。此外，還要感謝交大何信瑩老師、楊進木老師、盧錦隆老師、黃憲達老師和彭慧玲老師，在修課和研究上，不吝給予幫忙和協助。中研院資訊所張復老師和日本東京大學的 Paul Horton 老師，也提供機會讓我從不同角度去思考問題。

感謝台大和師大的師長們，在研究所和大學時期幫我奠定將來作學術研究的基礎，其中台大陳信希老師和趙坤茂老師的鼓勵，讓我有勇氣跨入生物資訊學的殿堂，學習這些有趣新穎的題目；師大陳世旺老師、柯佳伶老師和蔡榮宗老師的鼓勵，讓我在研究路途遇到挫折沮喪時，給我信心繼續努力。

實驗室成員給予我的幫助也是促使我順利完成學業的一大助力，尤其感謝家銘、樺聲、成偉和光倫，這些優秀的合作夥伴，藉由彼此間的討論激發出許多有趣的想法，每當研究遇到瓶頸和困難時往往也能迎刃而解；而坤品學長、宗翰學長和敏育學長所提供的許多寶貴經驗，使我更有信心面對各個難題。實驗室信男、鯨太、辭修、一原、翊文、俊男、義樺、文奇、殷豪、煌耀、照玲等人，大家這幾年的大力幫忙和熱心協助，我都會銘記在心。還要感謝建興、青嵚、涵堃和景盛，跟你們討論研究問題，總是讓我獲益良多。

最後要感謝依兒和家人的支持和鼓勵。在爸爸和媽媽無條件的支持和信任下，讓我在研究所和大學畢業時，能夠繼續就讀博士班從事喜愛的學術研究；也要感謝妹妹莉容，總是第一時間跟我分享開心或是難過。特別感謝依兒，在研究或生活上遇到挫折時，給我鼓勵和支持。因為你們，我才能心無旁騖專注在研究上，感謝你們！

Contents

CHAPTER 4

Prediction of RNA-binding Sites in Proteins

List of Figures

List of Tables

CHAPTER 1

Computational Approaches for Protein Function Analysis

1.1 Protein subcellular localization prediction

1.1.1 Introduction

The prediction of protein subcellular localization (PSL) focuses on determining localization sites of unknown proteins in a cell. The study of PSL is important for elucidating protein functions involved in various cellular processes. Despite recent technical advances, experimental determination of PSL remains time-consuming and labor-intensive. In addition, researches in the post-genomic era have yielded a tremendous amount of sequence data. Given the size and complexity of the data, many researchers would prefer to use prediction systems to identify and screen possible candidates for further analyses. Hence, computational approaches have become increasingly important.

1.1.2 Previous work

Extensive studies of PSL prediction have led to the development of several methods, which can be classified as follows.

1. *Amino acid composition-based methods* These methods utilize machine learning techniques, including neural networks (Emanuelsson, et al., 2000) and support vector machines (SVM) (Hoglund, et al., 2006; Hua and Sun, 2001; Park and Kanehisa, 2003; Pierleoni, et al., 2006; Wang, et al., 2005; Yu, et al., 2006;

Yu, et al., 2004). This category includes methods like P-CLASSIFIER (Wang, et al., 2005) and CELLO (Yu, et al., 2006; Yu, et al., 2004), which utilize *n*-peptide composition-based SVM approaches.

2. *Methods that integrate various protein characteristics* Several methods including expert systems (Bannai, et al., 2002; Nakai and Kanehisa, 1991), *k*-nearest neighbor (Chou and Cai, 2005; Horton, et al., 2006), SVM (Bhasin, et al., 2005; Nair and Rost, 2005; Su, et al., 2006), support vector data description (Lee, et al., 2006), and Bayesian networks (Gardy, et al., 2005; Gardy, et al., 2003; Lu, et al., 2004; Scott, et al., 2005), integrate various biological features that influence localization. The features that characterize a protein can be extracted from biological literature, public databases, and related prediction systems. Both PSORTb (Gardy, et al., 2005; Gardy, et al., 2003) and PSLpred (Bhasin, et al., 2005) integrate different analytical modules and demonstrate that the hybrid approaches perform better than each individual module.

3. *Sequence homology-based methods* It has been suggested that PSL is an evolutionary conserved trait (Gardy and Brinkman, 2006; Nair and Rost, 2002). Efforts to address the relationship between evolutionary information and localization identity have relied heavily on exploiting sequence similarity to infer PSL. Such methods include phylogenetic profiling (Marcotte, et al., 2000), domain projection (Mott, et al., 2002), and a sequence homology-based method (Yu, et al., 2006). Several other methods, such as PSORTb and PSLpred, also incorporate such sequence homology-based components in their analyses.

1.1.3 Challenges

The prediction of PSL presents several challenges. First, the performance of amino acid composition-based and sequence homology-based methods might be significantly degraded if homologous sequences are not detected. Second, the results of these two methods are generally difficult to interpret; therefore, it is difficult to determine which biological features should be used to identify specific PSL and why they work well for prediction. If the features were biologically interpretable, the resultant knowledge could help in designing artificial proteins with the desired properties. Meanwhile, methods that integrate various features could suffer from the problem of low coverage in high-throughput proteomic analyses due to the lack of information to characterize unknown proteins. Finally, many PSL methods are implemented on redundant training sets, which might lead to overestimation of the predictive performance. Thus, the performance would be significantly lower if redundant sequences were meticulously removed.

1.1.4 Our methods – PSL101 and PSLDoc

In Chapter 2, we propose a hybrid method that combines a one-versus-one (1-v-1) SVM model referred to as PSL101 (Protein Subcellular Localization prediction by 1-On-1 classifiers) and a structural homology approach called PSLsse (Protein Subcellular Localization prediction by secondary structure element alignment) to predict the PSL for Gram-negative bacteria. PSL101 comprises a number of binary classifiers, where compartment-specific biological features derived from Gram-negative bacteria translocation pathways are incorporated. In PSLsse, we employ secondary structure alignment for structural similarity comparison and assign the known localization of the top-ranked protein as the predicted localization of a query protein. Experiment results show that PSL101 achieves high prediction accuracy, which demonstrates that biological features derived from Gram-negative bacteria translocation pathways significantly enhance the performance. Moreover, since the selected features are biologically interpretable, they can be easily applied to advanced analyses and experimental designs. Most notably, the overall accuracy of combining PSL101 and PSLsse is further improved to 93.7%, which is a 2.5% improvement over the second best method. Our analysis suggests that, in addition to sequence homology, structural homology can also be an effective indicator for inferring PSL. Lastly, since sequence redundancy in the training data often leads to overestimation of prediction accuracy, we present an evaluation using non-redundant data sets. It is also known that cross-validation may overestimate the predictive performance when parameters are optimized repeatedly on the same test data. Therefore, we adopt a three-way data split procedure for evaluating the non-redundant data sets. The results suggest that these techniques can prevent overestimation of the performance such that the general performance of PSL prediction should be approximately 85%. In the assessment of the evaluation data sets, our hybrid method also provides accurate prediction, especially for those sequences of low homology to the training set.

In addition, PSL prediction can be formulated as a document classification problem. The document classification problem is to assign an electronic document to one or more categories, based on its contents. A protein sequence can be considered as the content of a document, and localization sites are considered as categories. To predict the localization site(s) of a protein is equivalent to predicting the topic (e.g., sport, politics) of a document (e.g., a piece of news). Given a large number of documents, document classification is usually tackled by the following three steps. First, documents have to be transformed into feature vectors in which each distinct term corresponds to a feature. The value of a feature in a vector represents the weight of a term in a document. Another set of documents with known categories are used as training set. Second, because of high-dimensional feature spaces, feature reduction is necessary before applying machine learning methods, to improve generalization accuracy (Valdes-Perez, et al., 2000) and to avoid overfitting (Namburu, et al., 2005; Valdes-Perez, et al., 2000). Finally, these reduced feature vectors are used to perform the category assignment automatically. The first two steps could be considered as feature representation. In Chapter 3, we present another prediction method, PSLDoc (Protein Subcellular Localization prediction based on Document classification), which incorporates a probabilistic latent semantic analysis (PLSA) with a one-versus-rest (1-v-r) SVM model based on document classification techniques. The feature representation of PSLDoc includes the following tasks: (1) define the terms of a protein, (2) design a term weighting scheme, and (3) apply a feature reduction and extraction method. For a benchmark data set of Gram-negative bacteria, PSLDoc also performs better than the other approaches. Our method demonstrates that the specific feature representation for proteins can be successfully applied to PSL prediction.

1.2 Prediction of RNA-binding sites in proteins

1.2.1 Introduction

RNA-protein interaction plays an important role in various biological processes, such as protein synthesis, gene expression, post-transcriptional regulation, and antiviral drug discovery. The prediction results of RNA-binding sites in proteins can provide biological insights for investigating RNA-protein interaction. For instance, ribosome is a protein synthesis complex consisting of ribosomal RNAs (rRNAs) and proteins. Sunita *et al.* (Sunita, et al., 2007) applied predicted RNA-binding sites to study the relationship between RNA methyltransferases RsmC and 16S rRNA. In addition, Bechara *et al.* (Bechara, et al., 2007) incorporated predicted results from a RNA-binding site predictor to inspect the connection between fragile X mental retardation protein and G-quartet RNA structure. Moreover, some RNA viruses, such as human immunodeficiency virus (HIV) and hepatitis C virus, have a RNA genome and replicate themselves by interacting with host proteins (McKnight and Heinz, 2003). Therefore, identification of the RNA interacting residues in proteins provides valuable information for understanding the mechanisms of protein synthesis, gene regulation, and pathogen-host interaction.

In recent years, rapid advances in genomic and proteomic studies have yielded a tremendous amount of DNA and protein sequences. We used the keyword “RNA-binding” to search against the National Center for Biotechnology Information (NCBI) protein sequence database on June 9, 2008, and obtained 196,686 protein sequences. However, when searching against Protein Data Bank (PDB) (Berman, et al., 2002) for molecular / chain type containing protein and RNA, we only retrieved 684 structures. In addition, experimental determination of RNA-protein interaction remains time-consuming and labor-intensive. Therefore, computational approaches for predicting RNA-binding sites in proteins have become increasingly important to understand the mechanisms of RNA-protein interaction.

1.2.2 Previous work

Extensive studies of RNA-protein binding site prediction have lead to the development of several methods, which can be classified as follows.

1. Amino acid composition-based methods

Jeong *et al.* (Jeong, et al., 2004) used an artificial neural network (ANN) to predict RNA-protein interacting residues based on amino acid compositions and predicted secondary structure elements. It achieved Matthew's correlation coefficient (*MCC*) of 0.29 and overall accuracy of 77.50% along with specificity of 87.29% and sensitivity of 40.30%. Terribilini *et al.* (Terribilini, et al., 2006) presented RNABindR using a Naïve Bayes classifier on amino acid sequences to predict RNA binding sites in proteins. RNABindR attained *MCC*, overall accuracy, specificity, and sensitivity of 0.35, 84.80%, 93%, and 38%, respectively.

2. Evolutionary information-based methods

Jeong and Miyano (Jeong and Miyano, 2006) applied an ANN to predict the RNA interacting residues based on evolutionary information from the position-specific scoring matrix (PSSM), and achieved *MCC*, overall accuracy, specificity, and sensitivity of 0.39, 80.20%, 91.04%, and 43.40%, respectively. The *MCC* is further improved to 0.41 by the incorporation of weighted profiles. Kumar *et al.* proposed a predictor, PPRint (Kumar, et al., 2008), using PSSM profiles in a support vector machine (SVM) model, and it achieved *MCC*, overall accuracy, specificity, and sensitivity of 0.45, 81.16%, 89.55%, and 53.05%, respectively.

3. Hybrid methods

Wang and Brown (Wang and Brown, 2006) developed an SVM-based classifier, BindN, using features including relative solvent accessible surface area, hydrophobicity index, side chain pKa value, molecular mass, and BLAST results. The overall accuracy, specificity, and sensitivity of BindN are 74.25%, 75.70%, and 65.78%, respectively.

1.2.3 Challenges

Although many methods have been proposed for RNA-binding site prediction, several challenges still remain. First, many of previous methods yield low sensitivities in tradeoff for high specificities since some biological applications, such as identification of critical residues for site-specific mutagenesis, emphasize more on specificities rather than sensitivities (Kumar, et al., 2008; Terribilini, et al., 2006). These methods could suffer from low coverage of RNA-binding sites in high-throughput proteomic analyses. Second, the *MCC* values of existing methods remain in the range of 0.27~0.45, which presents a great room for improvement in the complementary measure of prediction performance. Finally, the parameters, such as the size of the sliding window, in most methods are selected from testing results evaluated by *n*-fold cross-validation, which may lead to overestimation of the prediction performance. Thus, the performance would be much lower if a more rigorous procedure is applied for parameter selection and performance evaluation.

1.2.4 Our method - RNAProB

In Chapter 4, we propose a method, RNAProB (RNA-Protein Binding site prediction), for prediction of RNA-binding residues in proteins using SVM classifiers and a new smoothed PSSM encoding scheme. Besides incorporation of upstream and downstream residues in a standard PSSM generated by PSI-BLAST, smoothed PSSM encoding also considers, for each amino acid in a sequence, the dependency effect from its neighboring amino acids. Similar to the spatial domain method used in the research field of image processing (Gonzalez and Woods, 2002), smoothed PSSM encoding calculates the evolutionary information of a central position based on the sum of those from surrounding residues. Experiment results show that the prediction performance of smoothed PSSM encoding performs better than the state-of-the-art approaches on the benchmark data sets. Evaluated by five-fold cross-validation, RNAProB outperforms the other approaches by 0.10~0.23 in *MCC*, 4.90%~6.83% in overall accuracy, and 0.88%~5.33% in specificity. Most notably, our method significantly improves sensitivity by 26.90%, 26.62%, and 7.05% for the RBP86, RBP109, and RBP107 data sets, respectively. To avoid data overfitting, we also incorporate a three-way data split procedure to evaluate the prediction performance of RNAProB. Our results show that our method not only achieves significant improvement on the performance, but also attains a high prediction accuracy evaluated by a three-way data split procedure. Moreover, our analysis indicates that smoothed PSSM could serve as a more discriminative feature for distinguishing between interacting and non-interacting residues. We believe that the proposed encoding scheme could be applicable to other research fields, such as DNA-binding sites, protein-protein interaction, and prediction of post-translational modification sites.

CHAPTER 2

Protein Subcellular Localization Prediction Based on Compartment-Specific Features Derived from Translocation Pathways

2.1 Methods

2.1.1 Gram-negative bacteria translocation pathways

Proteins synthesized in the cytosol must be targeted and transported to their designated compartments in Gram-negative bacteria through one of the translocation pathways (Wickner and Schekman, 2005). Gram-negative bacteria have five major PSL sites: the cytoplasm (CP), inner membrane (IM), periplasm (PP), outer membrane (OM), and extracellular space (EC). Figure 2.1 shows some of the translocation pathways in Gram-negative bacteria. Translocations through the IM are targeted, both co-translationally and post-translationally, to the SecYEG translocase via the signal recognition particle (SRP)-dependent pathways and the SecB-dependent pathways, respectively. Alternatively, proteins localized to the PP can cross the IM by the twin arginine translocation pathway. PP proteins can be inserted or translocated across the OM through five secretory pathways, including Type I (Holland, et al., 2005) and Type II (Pugsley, 1993) export systems. Regardless of the mode of translocation, the process is largely substrate specific, and therefore requires one or more signals in order to cross a membrane. For example, non-cytoplasmic proteins contain signal sequences that direct them to translocate through the IM. Furthermore, many proteins localized to a compartment have characteristic structures and amino acid compositions. Integral IM proteins contain mainly transmembrane α-helices, in which their cores are populated by hydrophobic residues. Therefore, we model the prediction system according to the translocation pathways by identifying signals that influence the targeting and compartment-specific features that correlate with various localization sites.

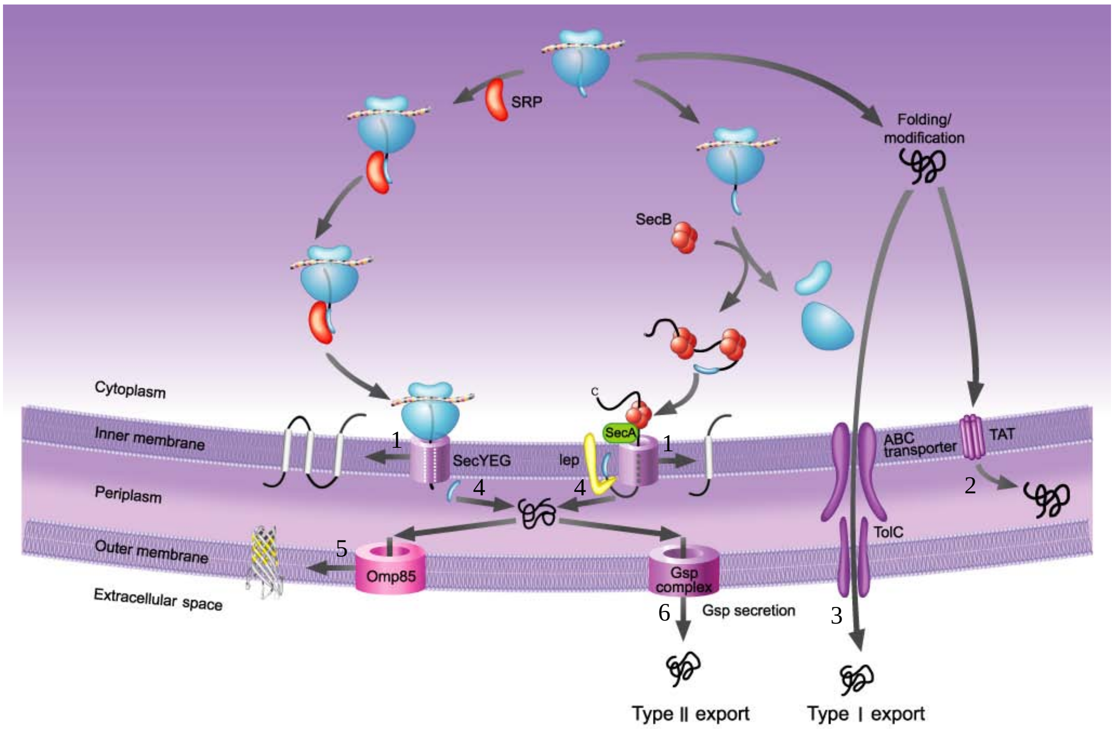


Figure 2.1: Diversity of Gram-negative bacteria translocation pathways. 1, 2, 3, 4, 5, and 6 represent translocation pathways from CP to IM, CP to PP, CP to EC, IM to PP, PP to OM, and PP to EC, respectively. SRP, signal recognition particle, SecB, export-specific cytoplasmic chaperone, SecA, preprotein translocase SecA subunit, SecYEG, preprotein translocase complex, lep, leader peptidase, TAT, twin argine translocase, Gsp complex, general secretion pathway complex, Omp85, outer membrane protein assembly factor, ABC transporter, ATP-binding cassette transporter, TolC, Type I secretion outer membrane protein. [Modified from Wickner and Schekman (2005) with their permission]

2.1.2 System architecture of PSL101

The system architecture of PSL101, shown in Figure 2.2, comprises ten binary 1-v-1 SVM (Vapnik, 1995) classifiers for the prediction of five localization sites of Gram-negative bacteria. Each translocation step across compartments *i* and *j* is represented by a binary classifier $C_{i,j}$ in which different biological features intrinsic to the proteins in compartments *i* and *j* are incorporated. All translocations in Figure 2.1, i.e., translocation pathways 1 to 6, can be modeled in this way by using six binary classifiers. The remaining four classifiers, although not biologically occurring, are still constructed with compartment-specific features and combined with the above classifiers for an integrated prediction. For each query protein, a predicted class and its corresponding probability are returned by each classifier. To determine the predicted localization site of the protein, we combine the results of the ten binary classifiers based on majority vote. In the case of a tie, the localization site with the highest average probability is assigned as the final prediction result.

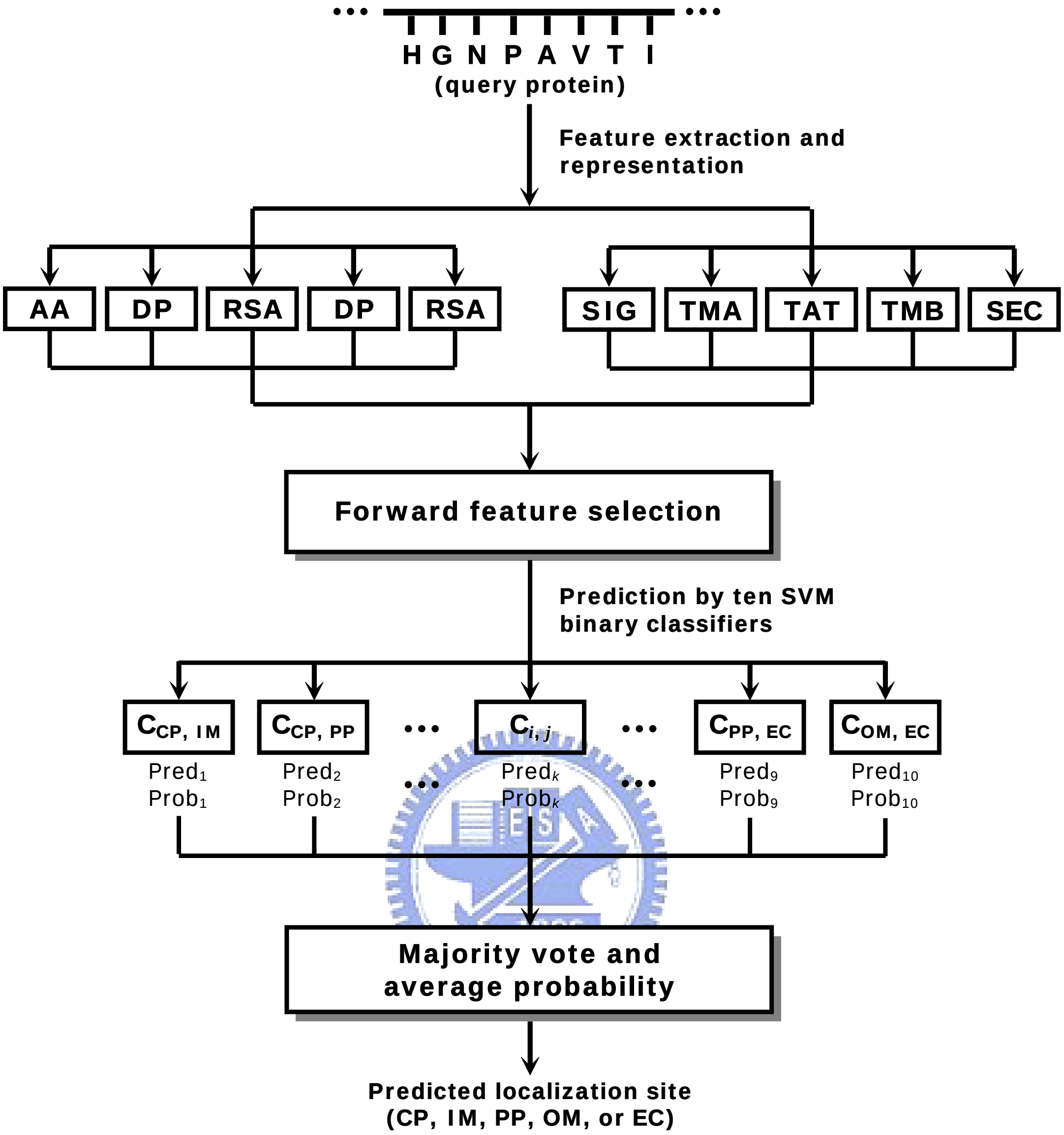


Figure 2.2: System architecture of PSL101.

2.1.3 Feature extraction and representation of PSL101

We consider the following biological features to distinguish between proteins translocated to different compartments, and construct our classification framework to mimic the translocation process of Gram-negative bacterial secretory pathways. Since some of these features may not be readily available, we utilize several web services to predict them.

General biological features

1. Amino acid (AA) composition: Protein descriptors based on *n*-peptide compositions or their variations have proved effective in PSL prediction (Yu, et al., 2004). If *n* = 1, then the *n*-peptide composition reduces to amino acid composition, which generates a 21 dimensional feature vector (i.e., 20 amino acid types plus a symbol 'X', for others) that represents the occurrence frequency of amino acids in a protein sequence.

2. Di-peptide (DP) composition: Similar to (1), if *n* = 2, the di-peptide composition gives a fixed length of 21×21 di-peptides, which represent the occurrence frequency of amino acid pairs in a protein sequence.

3. Relative solvent accessibility (RSA): Proteins in different compartments have various buried and exposed residue compositions (Nair and Rost, 2003; Nair and Rost, 2003). For example, CP proteins have a balance of acidic and basic surface residues, while EC proteins have a slight excess of acidic surface residues (Andrade, et al., 1998). We use amino acid compositions of both buried and exposed residues, with a cutoff of 25% (Cheng, et al., 2005), to represent the results derived by SABLE II (Adamczak, et al., 2005), a relative solvent accessibility prediction method.

4. Secondary structure elements encoding scheme 1 (SSE1): Transmembrane α-helices are frequently observed in IM proteins, while transmembrane β-barrels are primarily found in OM proteins (Pautsch and Schulz, 1998). Secondary structure elements (SSE) are crucial for detecting proteins localized in the IM and OM. We compute the amino acid compositions of three SSEs (Nair and Rost,

2003; Nair and Rost, 2005), α-helix, β-strand, and loop, based on the predictions of HYPROSP II (Lin, et al., 2005), a knowledge-based SSE prediction approach.

5. Secondary structure elements encoding scheme 2 (SSE2): SSE1 alone cannot discriminate proteins that share similar SSE compositions and localize in different compartments. For example, the SSE compositions of OM proteins might be similar to proteins localized in other compartments, but OM proteins are characterized by β-strand repeats throughout the transmembrane domains. To further depict such properties in a protein, three descriptors, composition, transition, and distribution, are used to encode predictions of HYPROSP II. Composition describes the global composition of a given SSE type in a protein. Transition characterizes the percentage frequency that amino acids of a particular SSE type are followed by a different type. Distribution measures the chain length within which the first, 25, 50, 75 and 100% of the amino acids of a particular SSE type are located (Dubchak, et al., 1995). An example is shown in Figure 1.1S in Appendix 1.1.

Compartment-specific biological features

1. Signal peptides (SIG): Signal peptides are N-terminal peptides, typically between 15 and 40 amino acids long, which target proteins for translocation through the general secretory pathway (Emanuelsson, et al., 2000). The presence of a signal peptide suggests that the protein does not reside in the CP. We employ SignalP 3.0 (Bendtsen, et al., 2004), a neural network- and hidden Markov model-based method, to predict the presence and location of signal peptide cleavage sites.

2. Transmembrane α-helices (TMA): Integral IM proteins are characterized by α-helices, typically 20-25 amino acids in length, which traverse the IM. The

presence of one or more transmembrane α-helices implies that the protein is located in the IM. We apply TMHMM 2.0 (Krogh, et al., 2001), a hidden Markov model-based method, to identify potential transmembrane α-helices.

3. Twin-arginine translocase (TAT) motifs: The twin-arginine translocase system exports proteins from the CP to the PP. The proteins translocated by twin-arginine translocase bear a unique twin-arginine motif (Berks, 1996), the presence of which is a useful feature for distinguishing between PP and non-PP proteins. We use TatP 1.0 (Bendtsen, et al., 2005), a neural network-based method, to predict the presence of twin-arginine translocase motifs.

4. Transmembrane β-barrels (TMB): A large number of proteins residing in the OM are characterized by β-barrel structures; thus, they could be candidate features for detecting OM proteins. We adopt TMB-Hunt (Garrow, et al., 2005), a method that uses a *k*-nearest neighbor algorithm, to distinguish between transmembrane β-barrels and non- transmembrane β-barrels.

5. Non-classical protein secretion (SEC): For a long time, it was believed that an N-terminal signal peptide was absolutely necessary to export a protein to the extracellular space. However, recent studies have shown that several EC proteins can be secreted without a classical N-terminal signal peptide (Nickel, 2003). Identification of non-classical protein secretion could be a potential discriminator for CP and EC proteins. Predictions from SecretomeP 2.0 (Bendtsen, et al., 2005), a non-classical protein secretion prediction method, are incorporated in our method.

2.1.4 Sequence and structure conservation

Because PSL tends to be evolutionary conserved, the known localization sites of homologous sequences could be useful indicators of the actual localization of an unknown protein. We apply both sequence and structural homology approaches to infer localization. For the sequence homology approach, we develop a prediction method, called PSLseq, which is based on pairwise sequence alignment of ClustalW (Thompson, et al., 1994). In the structural homology approach, we employ secondary structural similarity comparison, referred to as PSLsse. Based on secondary structure elements predicted by HYPROSP II, we use SSEA (Fontana, et al., 2005) to perform pairwise secondary structure alignment. In the sequence and structural homology approaches, the known localization of the top-rank aligned protein is assigned to the query protein as its predicted localization.

2.1.5 Data sets

To assess our method, we utilize several data sets of Gram-negative bacteria proteins that have been used in previous works (Bhasin, et al., 2005; Gardy, et al., 2005; Gardy, et al., 2003; Wang, et al., 2005; Yu, et al., 2006; Yu, et al., 2004). Table 2.1 lists the number of proteins in different localization sites in the data sets, which are detailed in Table 1.1S of Appendix 1.2.

1. Benchmark data sets: Derived from the first release of ePSORTdb (Rey, et al., 2005), the first data set, referred to as PS1302, consists of proteins with experimentally determined localizations. The second data set, PS1444, is an expanded version of PS1302.

2. Non-redundant data sets: To assess the predictive performance of non-homologous proteins, we utilize CD-HIT (Li and Godzik, 2006), a redundancy filtering program, to eliminate sequences that share greater or equal to 30% sequence identity in the PS1302 and PS1444 data sets, which yields the NR755 and NR828 data sets, respectively.

Table 2.1: Number of proteins distributed in different localization sites in the data sets.

Localization	Benchmark		Non-redundant		Evaluation		
	PS1302	PS1444	NR755	NR828	EV90_high	EV153_low	EV243_all
Cytoplasm (CP)	248	278	206	229	28	96	124
Inner membrane (IM)	268	309	182	205	26	26	52
Periplasm (PP)	244	276	147	161	13	11	24
Outer membrane (OM)	352	391	134	148	19	9	28
Extracellular space (EC)	190	190	86	85	4	11	15
Total	1302	1444	755	828	90	153	243

3. Evaluation data sets: Recently, a new data set (Gardy and Brinkman, 2006) comprised of 299 proteins was created for comparison of different methods. We first apply ClustalW to divide the new set into two subsets according to the sequence identity of each protein pair between the 299 proteins and proteins in the known training sets (i.e., PS1302 and PS1444) with a cutoff of 30%. Then, redundant sequences are removed from each subset by CD-HIT with a 30% threshold; the resultant non-redundant data sets are called EV90_high (greater or equal to 30%) and EV153_low (less than 30%). The combination of both sets is referred to as the EV243_all data set.

2.1.6 Performance assessment

For comparison with other approaches, we follow the measures used in previous works (Bhasin, et al., 2005; Wang, et al., 2005; Yu, et al., 2006; Yu, et al., 2004). To assess the performance in each localization class, the accuracy and Matthew’s correlation coefficient (*MCC*) (Matthews, 1975) are calculated by Equations (1.1) and (1.2), respectively. The overall accuracy is defined in Equation (1.3).

$$Acc_i = \left(TP_i / N_i \right) \times 100\% \quad (1.1)$$

$$MCC_i = \frac{(TP_i)(TN_i) - (FP_i)(FN_i)}{\sqrt{(TP_i + FN_i)(TP_i + FP_i)(TN_i + FP_i)(TN_i + FN_i)}} \quad (1.2)$$

$$Acc = \left(\sum_{i=1}^{l} TP_i \Big/ \sum_{i=1}^{l} N_i \right) \times 100\% \qquad (1.3)$$

where l = 5 is the total number of localization sites for Gram-negative bacteria, and TP_i, TN_i, FP_i, FN_i, and N_i are, respectively, the number of true positives, true negatives, false positives, false negatives, and proteins in localization site *i*. *MCC*, which considers both under- and over-predictions, provides a complementary measure of the predictive performance, where *MCC* = 1 indicates a perfect prediction, *MCC* = 0 indicates a completely random assignment, and *MCC* = -1 indicates a perfectly reverse correlation.

2.1.7 Training and testing

We apply the LIBSVM (Chang and Lin, 2001) software in our experiments. For all classifiers, we use the Radial Basis Function kernel, and tune the cost (*c*) and gamma (*γ*) parameters. The probability estimates in LIBSVM are used to determine the confidence levels of the classifications (Wu, et al., 2004). The performance of PSL101 is assessed as follows.

1. *n*-fold cross-validation: The data set is randomly partitioned into ten distinct non-overlapping sets of proteins (i.e., *n* = 10), nine of which are used to train the predictor. Then, the accuracy of the predictor is evaluated on the remaining set.

2. Three-way data split: To prevent data overfitting, a three-way data split procedure (Ritchie, et al., 2003) is used to assess the performance of PSL101. The data set is randomly divided into three disjoint sets, i.e., a training set for classifier learning, a validation set for feature selection and parameter tuning, and a test set for performance evaluation. Here, we divide the data set into ten distinct sets: eight for training, one for validation, and one for testing.

2.2 Results and discussion

2.2.1 Effect of biological features derived from Gram-negative bacteria translocation pathways

Since it is impractical to try all possible feature combinations in different classifiers, heuristics guided by biological insights are used to determine a small subset of feature sets specific to each classifier. Starting with an empty subset, a sequential forward search algorithm (Tsai, et al., 2006) keeps adding the best feature sets that improve the accuracy. The process terminates when adding a feature set no longer makes any improvement. The performance of PSL101 evaluated by ten-fold cross-validation for the benchmark data sets is shown in the leftmost column of Table 2.2. PSL101 attains overall accuracy of 92.7% and 91.6% for the PS1302 and PS1444 data sets, respectively. Most notably, CP and IM proteins attain accurate prediction performance in terms of both accuracy and *MCC*, which can be explained by the fact that proteins localized in CP and IM are characterized by several well-known biological features in our method.

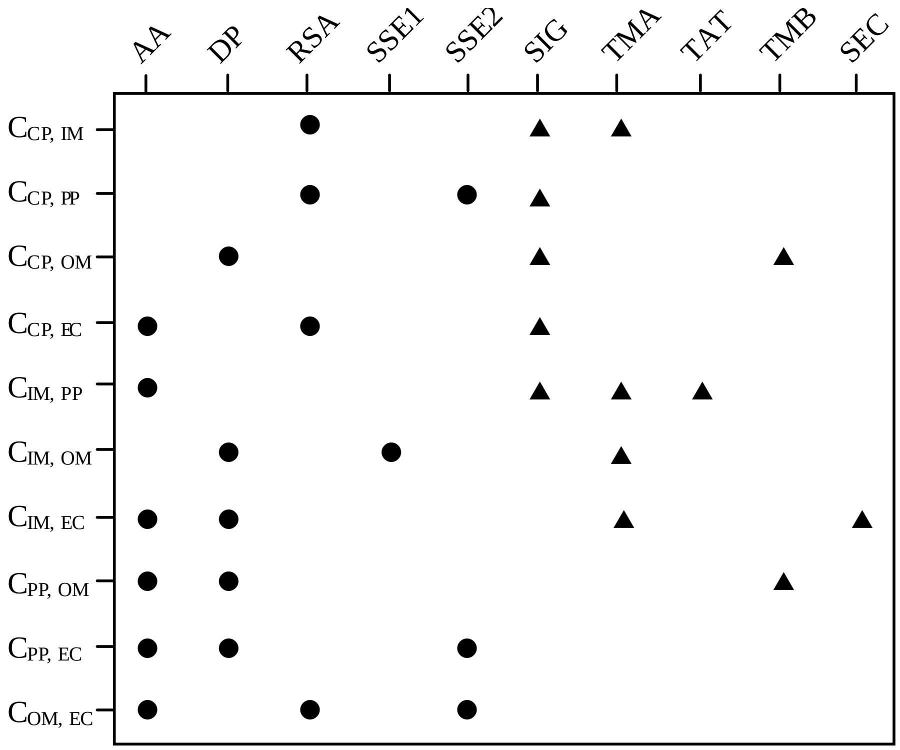


Figure 2.3: Feature combinations derived from the PS1302 data set using cross-validation. Selected general and compartment-specific features are represented by filled circles and triangles, respectively.

The features selected from PSL101 for the PS1302 data set using cross-validation are shown in Figure 2.3; the same set of features is used in the corresponding training and testing scheme for the PS1444 data set. The experiment results demonstrate that our feature selection not only yields a significant improvement in the performance, but also correlates well with biological insights. For example, in Figure 2.3, PSL101 selects signal peptides, transmembrane α-helices, and relevant solvent accessibility (i.e. SIG, TMA, and RSA) as the optimal features to distinguish CP and IM proteins. In addition, di-peptide composition, signal peptides, and transmembrane β-barrels (i.e. DP, SIG, and TMB) are used in the discrimination of CP and OM proteins. The combination of general and compartment-specific features works well in differentiating between any two compartments in each classifier; accordingly, the overall accuracy of the combined predictions of each classifier is improved. The re

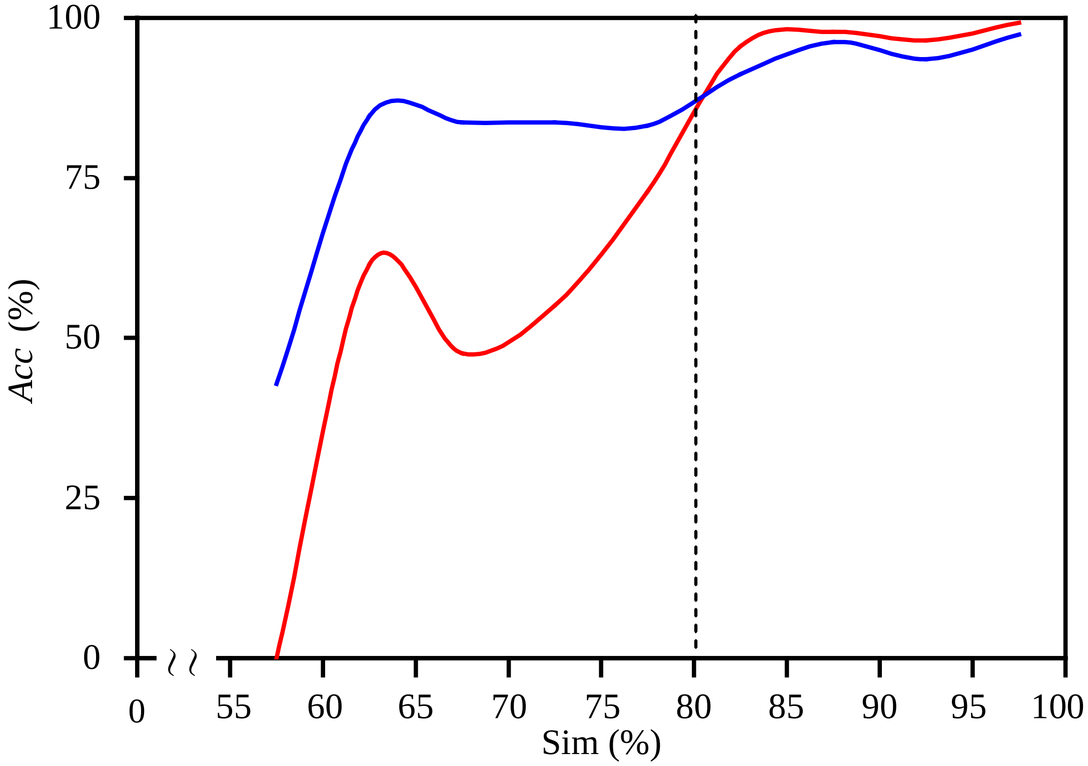


Figure 2.4: The distribution of the prediction accuracy as a function of secondary structure similarity. The blue line and the red line indicate the distribution of the prediction accuracy as a function of secondary structure similarity for PSL101 and PSLsse using cross-validation for the PS1444 data set, respectively.

sults support our assumption that compartment-specific biological features derived from Gram-negative bacteria translocation pathways can significantly enhance the performance of PSL prediction. Moreover, the selected features are biologically interpretable and can be easily applied in further analyses.

2.2.2 Effect of sequence and structure conservation

We now explore the relationship between sequence and structural similarity and localization identity. Both sequence and structural homology approaches, referred to as PSLseq and PSLsse, are developed to infer localization based on sequence alignment using ClustalW and secondary structure alignment using SSEA, respectively. Figure 2.4 shows that when the structural similarity is greater or equal to 80%, PSLsse performs slightly better than PSL101; otherwise, PSL101 is significantly better. Thus, we propose a hybrid approach that combines PSLsse and PSL101, called PSLsse+PSL101. For each query protein, if the top-rank aligned protein shares an 80% or greater structural similarity with any of the proteins in the training set, the localization is predicted by PSLsse; otherwise, it is predicted by PSL101. In addition, we implement another hybrid approach, called PSLseq+PSL101, which uses a cutoff of 30% sequence identity [7] to combine PSLseq and PSL101.

Table 2.2 compares the performance of different hybrid approaches using ten-fold cross-validation for the benchmark data sets. Compared with PSL101, the performance of the two hybrid approaches, PSLseq+PSL101 and PSLsse+PSL101, is significantly enhanced in terms of the overall accuracy, as well as the accuracy and *MCC* of most localization sites. Most notably, the accuracy of EC proteins in both data sets is improved by 1.6%~5.3%, which suggests that homology-based approaches can compensate for the performance of PSL101 and thereby enhance the prediction of EC proteins. Moreover, PSLsse+PSL101 achieves an overall accuracy of 93.7% and 93.2% in the PS1302 and PS1444 data sets, respectively, which are 0.6%~0.8% improvements over PSLseq+PSL101. We show that homology approaches based on sequence and structure conservation work well in PSL prediction; in fact, structural homology could be effective for prediction in addition to sequence homology. Thus, it could also be a useful indicator for inferring PSL.

Table 2.2: Comparison of different hybrid approaches using cross-validation for the benchmark data sets.

	PS1302					
Localization	PSL101		PSLseq+PSL101		PSLsse+PSL101	
	Acc (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*
CP	97.2 (94.8)	0.91 (0.89)	96.4 (94.4)	0.90 (0.89)	95.6 (94.4)	0.90 (0.90)
IM	94.4 (92.9)	0.95 (0.94)	93.3 (91.8)	0.95 (0.93)	93.3 (91.8)	0.94 (0.93)
PP	87.7 (88.1)	0.86 (0.84)	88.9 (88.9)	0.86 (0.85)	91.4 (91.0)	0.88 (0.88)
OM	94.3 (93.8)	0.94 (0.91)	95.5 (95.7)	0.96 (0.93)	96.3 (96.9)	0.96 (0.95)
EC	87.9 (83.2)	0.87 (0.84)	89.5 (85.8)	0.89 (0.87)	90.0 (87.9)	0.89 (0.89)
Overall	92.7 (91.2)	-	93.1 (91.9)	-	93.7 (92.9)	-
	PS1444					
Localization	PSL101		PSLseq+PSL101		PSLsse+PSL101	
	Acc (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*
CP	96.0 (94.2)	0.91 (0.90)	94.6 (92.8)	0.89 (0.88)	95.0 (93.5)	0.91 (0.90)
IM	94.5 (92.6)	0.95 (0.94)	93.5 (91.6)	0.94 (0.93)	93.5 (91.6)	0.94 (0.93)
PP	85.1 (88.0)	0.82 (0.83)	87.0 (88.4)	0.84 (0.83)	90.2 (91.7)	0.86 (0.87)
OM	94.9 (93.9)	0.93 (0.91)	95.9 (95.7)	0.95 (0.93)	96.7 (96.4)	0.96 (0.95)
EC	82.6 (83.2)	0.83 (0.85)	87.9 (86.3)	0.87 (0.88)	87.4 (87.9)	0.87 (0.89)
Overall	91.6 (91.1)	-	92.4 (91.6)	-	93.2 (92.8)	-

§ The performance of incorporating a three-way data split procedure is indicated in the parentheses.

Table 2.3: Performance comparison of different approaches using cross-validation for the benchmark data sets.

Localization	PS1302									
	HYBRID		CELLO		PSORTb v.1.1		PSLpred		P-CLASSIFIER	
	Acc (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*
CP	95.6	0.90	90.7	0.85	69.4	0.79	90.7	0.86	94.6	0.85
IM	93.3	0.94	88.4	0.92	78.7	0.85	86.8	0.88	87.1	0.92
PP	91.4	0.88	86.9	0.80	57.6	0.69	90.3	0.90	85.9	0.81
OM	96.3	0.96	94.6	0.90	90.3	0.93	95.2	0.95	93.6	0.90
EC	90.0	0.89	78.9	0.82	70.0	0.79	90.6	0.84	86.0	0.89
Overall	93.7	-	88.9	-	74.8	-	91.2	-	89.8	-

Localization	PS1444									
	HYBRID		CELLO II		PSORTb v.2.0		PSLpred		P-CLASSIFIER	
	Acc (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*
CP	95.0	0.91	95.3	0.89	70.1	0.77	-	-	-	-
IM	93.5	0.94	90.0	0.91	92.6	0.92	-	-	-	-
PP	90.2	0.86	87.7	0.82	69.2	0.78	-	-	-	-
OM	96.7	0.96	92.8	0.90	94.9	0.95	-	-	-	-
EC	87.4	0.87	79.5	0.82	78.9	0.86	-	-	-	-
Overall	93.2	-	90.0	-	82.6	-	-	-	-	-

§ The best performance of overall and individual localization sites is underlined.

2.2.3 Performance comparison of *n*-fold cross validation and three-way data split

The performance of the three-way data split experiments is shown in parentheses in Table 2.2. The features selected from PSL101 for the PS1302 data set using three-way data split are shown in Figure 1.2S in Appendix 1.3; the same set of features is used in the corresponding training and testing scheme for the PS1444 data set. The overall accuracy of PSL101, PSLseq+PSL101, and PSLsse+PSL101 drop 0.4%~1.5% for both the PS1302 and PS1444 data sets. Specifically, the accuracy and *MCC* of the same localization sites are consistent across the two different data sets. Moreover, the performance of the two data sets evaluated using a three-way data split is more consistent than that assessed by ten-fold cross-validation. This suggests that a three-way data split procedure could avoid overestimation of the predictive performance; therefore, it should be considered in PSL prediction.

2.2.4 Comparison with other approaches using the benchmark data sets

Table 2.3 compares the performance of PSLsse+PSL101, referred to as HYBRID, with other prediction methods using cross-validation on the benchmark data sets. HYBRID attains the best overall accuracy of 93.7% and 93.2% for the PS1302 and PS1444 data sets, respectively. In both sets, HYBRID achieves improvements of 2.5%~3.2% in overall accuracy compared to the second best approaches in each data set. With respect to accuracy and *MCC*, HYBRID performs better than the other approaches in most localization sites. HYBRID ranks the best in terms of accuracy for CP, IM, and OM proteins, in which more biological features are incorporated than the other localization sites. The high predictive performance for CP, IM, and OM proteins demonstrates that biological features derived from Gram-negative bacteria translocation pathways are effective for PSL prediction. Most notably, it outperforms the second best approaches for IM proteins by 4.9~14.6% and 0.9~3.5% in terms of accuracy for the PS1302 and PS1444 data sets, respectively. This is a particular strength of HYBRID because IM proteins constitute the key components of various cellular processes and serve as important targets for drug discovery [30]. In addition, it is interesting to note that the accuracy of IM proteins is significantly improved from 78.7% in PSORTb v.1.1 to 92.6% in PSORTb v.2.0, in which an expanded homology module is incorporated. This also lends support on our assumption that sequence and structural homology approaches could be effective indicators for inferring PSL.

2.2.5 Comparison with other approaches using the evaluation data sets

The evaluation data sets were submitted to the web servers of each prediction method. The predictive performance is shown in Table 2.4. CELLO II and P-CLASSIFIER achieve consistent overall accuracy in the range of 71.9%~77.8% for the EV90_high and EV153_low data sets. PSLpred attains overall accuracy of 72.5% and 88.9% for the EV153_low and EV90_high sets, respectively. PSORTb v.2.0 performs very well for the EV90_high set, but poorly for the EV153_low set. HYBRID yields the best predictions for lowly homologous sequences and ranks second best for highly homologous sequences. This demonstrates that when no homologous sequences are detected, biological features derived from Gram-negative bacteria translocation pathways yields accurate prediction; on the other hand, the incorporation of structural homology approach further improves the predictive performance for highly homologous sequences. When both data sets are evaluated on the EV243_all set, HYBRID achieves an overall accuracy of 84.0%, which is a 5.4% improvement over the second best method. This suggests that HYBRID could enhance the robustness of PSL prediction, especially when highly homologous sequences are not detected.

2.2.6 Performance of non-redundant data sets

In both benchmark data sets, proteins sharing up to 30% sequence identity comprise approximately 42% of the sets. One drawback of a high level of redundancy in data sets is that it could lead to poor generalization for a predictor, since the predictor might fail to assign a correct PSL, especially when low homology is detected. For this reason, the construction of non-redundant data sets is necessary when evaluating the performance of PSL prediction.

Here, we present performance assessments using non-redundant sequences from Gram-negative bacteria data sets. Using the same features derived from the PS1302 set by cross-validation, we use HYBRID to train and evaluate the two non-redundant sets via ten-fold cross-validation. The performance is shown in Table 2.5. The overall accuracy declines markedly by approximately 8% using the non-redundant sets compared with those using the redundant sets. The *MCC* for individual localization sites also drops by 0.04~0.26. These results indicate that the general performance of PSL prediction for Gram-negative bacteria is approximately 85% for non-redundant data sets. Methods that are less dependent on homology detection should be developed if highly homologous sequences are removed completely.

Table 2.4: Predictive performance of different prediction methods for the evaluation data sets.

	EV153_low									
Localization	HYBRID		CELLO II		PSORTb v.2.0		PSLpred		P-CLASSIFIER	
	Acc (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*
CP	91.7	0.67	91.7	0.70	63.5	0.61	89.6	0.59	91.7	0.66
IM	65.4	0.73	46.2	0.64	46.2	0.58	38.5	0.41	30.8	0.48
PP	45.5	0.25	81.8	0.49	0.0	-0.03	54.5	0.34	81.8	0.49
OM	44.4	0.58	33.3	0.34	22.2	0.46	44.4	0.58	22.2	0.17
EC	27.3	0.43	45.5	0.50	9.1	0.29	45.5	0.54	27.3	0.33
Overall	76.5	-	76.5	-	49.7	-	72.5	-	71.9	-

	EV90_high									
Localization	HYBRID		CELLO II		PSORTb v.2.0		PSLpred		P-CLASSIFIER	
	Acc (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*
CP	100.0	0.95	92.9	0.83	100.0	1.00	96.4	0.88	92.9	0.78
IM	96.2	0.97	73.1	0.75	100.0	1.00	92.3	0.92	80.8	0.84
PP	100.0	0.96	61.5	0.58	100.0	1.00	92.3	0.83	46.2	0.46
OM	94.7	0.97	73.7	0.67	94.7	0.97	68.4	0.79	73.7	0.69
EC	75.0	0.86	75.0	0.54	100.0	1.00	100.0	0.81	75.0	0.54
Overall	96.7	-	77.8	-	98.9	-	88.9	-	77.8	-

	EV243_all									
Localization	HYBRID		CELLO II		PSORTb v.2.0		PSLpred		P-CLASSIFIER	
	Acc (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*	*Acc* (%)	*MCC*
CP	93.5	0.80	91.9	0.77	71.8	0.73	91.1	0.72	91.9	0.73
IM	80.8	0.85	59.6	0.70	73.1	0.80	65.4	0.68	55.8	0.67
PP	75.0	0.56	70.8	0.51	54.2	0.66	75.0	0.57	62.5	0.45
OM	78.6	0.85	60.7	0.58	71.4	0.83	60.7	0.73	57.1	0.53
EC	40.0	0.57	53.3	0.50	33.3	0.57	60.0	0.62	40.0	0.39
Overall	84.0	-	77.0	-	67.9	-	78.6	-	74.1	-

§ The best performance of overall and individual localization sites is underlined. HYBRID is trained on the PS1444 data set.

Table 2.5: Performance of non-redundant data sets.

Localization	NR755		NR828	
	Acc (%)	*MCC*	*Acc* (%)	*MCC*
CP	95.6	0.86	97.8	0.87
IM	88.5	0.88	88.8	0.90
PP	81.0	0.76	80.7	0.76
OM	85.1	0.84	83.8	0.82
EC	64.0	0.65	57.6	0.61
Overall	85.6	-	85.6	-

2.3 Conclusion

In this chapter, we have proposed a hybrid method for predicting PSL for Gram-negative bacteria based on a combination of a 1-v-1 SVM model using compartment-specific biological features and a structural homology approach using secondary structure alignment. Experiment results show that the SVM model achieves high prediction accuracy for both benchmark data sets, thus supporting the assumption that biological features derived from Gram-negative bacteria translocation pathways could significantly improve the performance. The overall accuracy of combining the SVM model and the structural homology approach is further improved, which indicates that structural homology, like sequence homology, could also be a useful indicator for inferring PSL. A three-way data split procedure is incorporated to prevent overfitting of the parameters and features. In addition, non-redundant data sets have been used for the evaluation of Gram-negative bacteria. The results suggest that the performance could be overestimated if redundant sequences are considered. In the assessment of the evaluation data sets, our hybrid method provides accurate predictions, especially when sequences of low sequence similarity to the training data are detected. The proposed method can be used in large-scale analyses of proteomes and is freely available for public use at [32].

There are still some challenges to be addressed in PSL prediction. In our work, we only consider proteins with single localization sites. However, proteins with multiple localization sites are not a rarity, especially in higher order species [33,34]. In our future development, we will consider those proteins localized to multiple compartments. In addition, better accuracy and coverage are needed, particularly for several poorly predicted localization sites. We will also extend our method to combine more biological features, analyze multiple compartment proteins, and incorporate proteins of more species, including those of humans.

CHAPTER 3

Protein Subcellular Localization Prediction Based on Gapped-dipeptide Signatures and Probabilistic Latent Semantic Analysis

3.1 Methods

3.1.1 A baseline system using TFIDF

Before describing our PSL prediction method based on document classification, we introduce a baseline system for performance comparison that uses a traditional document classification method. Salton's vector space model (VSM) is one of the most widely used methods for ad-hoc retrieval (Manning and Schütze, 1999) in document classification. Each document is represented by a feature vector (vector, in short) composed of all terms in a collection of documents, where each entry (or feature) of the vector corresponds to a term and its value is given by the weight of the term in the document(Salton, et al., 1975). The similarity between two documents *d* and *q*, denoted by *sim*(*d*,*q*), can be defined as the cosine of the angle between their vectors, called *cosine similarity*, as shown below:

$$sim(d,q)=\cos\left(\angle(\vec{d},\vec{q})\right)=\frac{\left\langle \vec{d},\vec{q}\right\rangle}{\left\|\vec{d}\right\|\left\|\vec{q}\right\|} \tag{3.1}$$

where $\vec{d}$ denotes the vector for a document *d*. Given a collection of documents with known categories, we classify a document with unknown category (called *query document*) into the same category as the document whose cosine similarity with the query document is the largest. We refer to this prediction method as the *1-nearest neighboring* (*1*-NN) method based on cosine similarity. The advantage of the *1*-NN method is that there is no training required as in normal machine learning approach.

Weighting scheme, i.e., determining the weight of each entry in a vector, is crucial in document classification. In this baseline system, we use *term frequency–inverse document frequency* (TFIDF) as the weighting scheme. For a term t_i in a document d, a simple *term frequency* (TF) is the number of occurrences of t_i in the document, denoted as n_i. However, to prevent a bias towards longer documents, term frequency $tf(t_\mathrm{i},d)$ is usually normalized as follows:

$$tf(t_i, d) = \frac{n_i}{\sum_k n_k}, \tag{3.2}$$

where the denominator is the number of occurrences of all terms. The term frequency $tf(t_i,d)$ gives a measure of the importance of the term t_i in the document d. The higher the term frequency, the more likely the term is a good description of the content of the document. In contrast, *inverse document frequency* (IDF) of t_i is a measure of the general importance of the term. A semantically important term will often occur several times in a document if it occurs at all. However, semantically unimportant terms are spread out homogeneously over all documents. A frequently used IDF for t_i, $idf(t_i)$ (Salton and Buckley, 1988), is defined as follows:

$$idf(t_i) = \log \frac{|D|}{|(d_i \supset t_i)|}, \tag{3.3}$$

where $|D|$ is the number of documents in the collection, and $|(d_i \supset t_i)|$ denotes the number of documents in which t_i appears. In the TFIDF scheme, the weight of the term t_i in a document d, $W(t_i, d)$[1], equals to $tf(t_i, d)$ multiplied by $idf(t_i)$ (Salton and Buckley, 1988). The values in a vector are normalized to (0~1] by dividing the maximum value in the vector.

3.1.2 Gapped-dipeptides as the terms of proteins

PSLDoc uses gapped-dipeptide (Liang, et al., 2005) as the terms of a protein and calculates their weights according to position specific score matrix (PSSM) instead of the TFIDF used in the baseline system. PLSA is used for feature reduction to improve learning efficiency and accuracy. The reduced feature vectors are input to five 1-v-r SVM classifiers corresponding to five localization sites. The probability estimated by a classifier can be considered as the confidence level of a target protein belonging to the corresponding localization site. The final prediction is determined to be the localization site whose corresponding classifier outputs the largest confidence score.

When considering proteins as documents, many different types of terms have been proposed, including single amino acid (AA) (Cedano, et al., 1997; Chou and Elrod, 1999; Garg, et al., 2005; Hua and Sun, 2001; Nair and Rost, 2003; Park and Kanehisa, 2003; Reinhardt and Hubbard, 1998) as a uni-gram descriptor, and the general *n*-peptide (Yu, et al., 2004), i.e., peptides of length *n* without gaps. In particular, for $n = 2$, dipeptide (Dip) is a neighboring bi-gram descriptor. However, AA and Dip, are not able to represent information between two gapped peptides. The use of *n*-peptide to capture long distance amino acid information will result in a high-dimensional vector space. For example, the feature number of a vector is 3,200,000 (= 20^5), when *n* equals 5. Liang *et al.* (Liang, et al., 2005) proposed a method based on amino acid-coupling patterns to extract the information from a protein sequence, which works well on distinguishing thermophilic proteins. An amino

[1] In this paper, we consider the weights of the terms in a document and the vector to be the same. Which is denoted by $W(t_i, d)$ or $W(t_i, \vec{d}\,)$ depending on the context.

acid-coupling pattern *XdZ* denotes the peptides of length $d + 2$ such that amino acids *X* and *Z* are separated by *d* amino acids, where *d* can be negative depending on whether the position of *X* closer to N-terminus or C-terminus.

We adopt the same encoding scheme as in Liang *et al.* with non-negative *d* as the term of a protein sequence regardless of whether the pattern appear near the N-terminus or C-terminus. We call such amino acid-coupling pattern as *gapped dipeptide*. For example, the gapped dipeptides for *d*=0 are dipeptide without gaps (Dip's). Given a positive integer *l* as *the upper bound of gapped distance*, each protein sequence is represented by a vector in the space of gapped dipeptides with each feature given by *XdZ* for $0 \leqq d \leqq l$. The length of vectors is the number of all possible combinations of gapped dipeptide, i.e., (*l*+1)x20x20. For example, given *l* = 10, a protein is represented as a feature vector of 4,400 (= 11x20x20) features.

3.1.3 Term weighting - position specific score matrix information

Motivation

On the basis of the finding in a previous work that sequence identity and subcellular localizations of proteins have a strong correlation (Yu, et al., 2006), Yu *et al.* (Yu, et al., 2006) proposed a homology search method for PSL prediction, which predicted the localization of a query protein by the most similar protein among the aligned protein sequences with known localizations generated by the global alignment program ALIGN (Myers and Miller, 1988). The authors observed that, when the query protein and its most similar protein with known localization have sequence identity over 30%, the homology search method performed very well with 97.7% accuracy. But the prediction performance dropped significantly when the sequence identity is under 20%. In this case, it would be difficult to predict the localization of a query protein based on the sequence identity or sequence information. To overcome this difficulty, we borrow the idea from protein secondary structure prediction, in which homologous sequences are usually removed from the testing and training data set (Cuff and Barton, 1999; Hua and Sun, 2001; Jones, 1999; Lin, et al., 2005; Rost and Sander, 1993; Wu, et al., 2004).

Most of the prediction methods address the problem of weak homology by utilizing sequence evolutionary information. One widely used representation of evolutionary information if the PSSM generated by PSI-BLAST (Altschul, et al., 1997),which has been used in PSIPRED (McGuffin, et al., 2000), a very popular secondary structure prediction method. PSI-BLAST finds remote homologues to a query protein from a chosen sequence database (e.g. NCBI nr database(Wheeler, et al., 2007) nr(Wheeler, et al., 2007)). Instead of TFIDF based on the sequence information, our weighting scheme is based on PSSM.

Position specific score matrix

The PSSM of a sequence S of length n is represented by an $n \times 20$ matrix, in which the n rows correspond to the amino acid sequence of S and the columns correspond to the 20 distinct amino acids. Each row of a PSSM represents the log-likelihood of the residue substitutions at the corresponding positions in S (Altschul, et al., 1997).The PSSM elements are scaled to the required 0~1 range using the following logistic function (Jones, 1999):

$$f(x) = \frac{1}{1+e^{-x}}, \tag{3.4}$$

where x is the original PSSM value. The higher the scaled value of the residue is, the higher the propensity of the residue is in this position. In PSLDoc, the PSI-BLAST's parameters were set to j = 5 (five iterations), e = 10^{-2} (E-value <0.01) and the sequence database is NCBI nr (which contains 3,747,820 sequences).

TFPSSM weighting scheme

We design a term weighting scheme based on PSSM, denoted by TFPSSM as follows. Given a protein sequence S of length n, any gapped dipeptide XdZ of S has PSSM entries corresponding to gapped dipeptide $S(i)dS(i+d+1)$ for $1 \leqq i \leqq n-(d+1)$, where $S(i)$ denotes the ith amino acid of S. Take the sequence MPLDLYNTLT as an example. Its PSSM (with original value without normalization) is shown in Figure 3.1. From the sequence information, M2D only occurs once. However, in view of PSSM, M2D may occur in the corresponding gapped dipeptides obtained from the sequence, i.e., M2D, P2L, L2Y, D2N, L2T, Y2L, N2T. We define the weight of XdZ in S as

$$W(XdZ,S) = \sum_{1 \le i \le n-(d+1)} f(i,X) \times f(i+d+1,Z) \qquad (3.5)$$

where $f(i,Y)$ denotes the normalized value of the PSSM entry at the ith row and the column corresponding to amino acid Y. In the above example, the weight of M2D based on PSSM is given by $f(1,\mathrm{M}) \times f(4,\mathrm{D}) + f(2,\mathrm{M}) \times f(5,\mathrm{D}) + \ldots + f(7,\mathrm{M}) \times f(10,\mathrm{D})$ $= 0.99995 \times 0.04743 + 0.11920 \times 0.00247 + \ldots + 0.00669 \times 0.26894$. It is unnecessary to incorporate IDF based on PSSM because the term occurs in all documents based on PSSM.

As mentioned before, each protein is represented by a vector, and each entry of the vector is given by TFPSSM of the corresponding gapped dipeptide. Note that the values in each feature vector are normalized between (0~1] by dividing the maximum value in the vector.

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
1 M	-3	-3	-4	-5	-3	-3	-4	-5	-4	0	1	-3	10	-2	-5	-4	-3	-4	-3	-1
2 P	2	-3	-3	-1	-3	-1	-1	-1	-4	-2	-4	-2	-2	-5	4	2	4	-5	-4	-3
3 L	-4	-5	-6	-6	-4	-3	-5	-6	-5	3	5	-5	4	0	-5	-5	-3	-4	-3	2
4 D	-2	5	-1	-3	-4	2	-1	-4	2	-5	-3	5	-2	-2	-4	-2	0	-1	0	-3
5 L	-4	-5	-6	-6	-4	-5	-6	-6	-4	4	4	-5	0	1	-5	-5	-3	-4	-3	3
6 Y	-4	-3	-3	-5	-5	-3	-4	-5	4	-4	-3	-3	-2	4	-5	-3	-2	2	8	-4
7 N	-4	-3	8	4	-6	-3	-2	-3	-2	-6	-6	-3	-5	-6	-4	-1	-3	-7	-5	-6
8 T	-2	-3	-1	-3	-1	-3	-3	-4	-3	-4	-4	-1	-4	-4	-4	4	6	-5	-4	-2
9 L	0	-1	-5	-5	-4	-3	-4	-4	-3	-1	5	-3	3	0	-4	-3	-3	-3	-2	-1
10 T	-1	-3	-1	-1	-4	-2	-3	-2	-1	-4	-3	-1	-3	-4	-4	3	6	-5	-4	-3

Figure 3.1: PSSM of the sequence MPLDLYNTL, where each entry is the original value without normalization.

3.1.4 Feature reduction - probabilistic latent semantic analysis

Motivation

There are some limitations on the VSM for document classification. First, the vector space is high-dimensional (Namburu, et al., 2005). Training and testing have to deal with the curse of dimensionality. Second, document vectors are typically very sparse, i.e., most features of a vector are zero, which are susceptible to noise (Kumar, et al., 2006), and cosine similarity could be inaccurate. Finally, the inner product defining document similarity can only match occurrences of the same terms. As a result, the vector representation does not capture semantic relations between terms. Furthermore, this representation, which considers a document as a bag of words, is unable to capture phrases and semantic/syntactic regularities.

Hence, dimension reduction (feature reduction) is proposed for dealing with above limitations. The goal of dimension reduction is to map similar terms to similar location in a low dimensional space called *latent semantic space*, which reflects semantic associations. The usual dimension reduction is Latent Semantic Analysis (LSA) (or called Latent Semantic Indexing in some papers) which uses singular value decomposition (SVD) to do data mapping (Deerwester, et al., 1990).The document similarity based on the inner product is computed in the latent semantic space. Experimentally, there are advantages of SVD over naive VSM. However, SVD still has the following disadvantages (Nair and Rost, 2003). First, the resulting dimensions might be difficult to interpret. For instance, the size of a vector is reduced from three to two by LSA as shown below:

{(A0A), (A1A), (G0G)} --> {(1.3* A0A + 0.2 * G0G), (A1A)}

The value of the first reduced feature equals 1.3 multiplied by the value of the origanl first feature plus 0.2 multiplied by the value of the original third feature. This leads to results which might be justifiable on the mathematical level, but have no interpretable meaning in the original application. Second, the probabilistic model of LSA does not match observed data (Hofmann, 2001). Third, the reconstruction may contain negative entries, which are inappropriate as a distance function for count vectors.

Probabilistic latent semantic analysis

Hofmann proposed probabilistic latent semantic analysis (PLSA) based on an aspect model for dealing with those disadvantages (Hofmann, 2001).The aspect model is a latent variable model for co-occurrence data (i.e., documents and terms) which associates an unobserved class variable $z \in Z=\{z_1, \ldots, z_K\}$with each observation. The weight of the term w in a document d, $W(w, d)$, is considered as a joint probability

P(*w*, *d*) between *w* and *d*, which is modeled through *z*, a latent variable which can be loosely thought of as a topic or a reduced feature. Thus, the joint probability *P*(*w*, *d*) based on PLSA model is

$$P(w,d) = P(d)P(w\,|\,d), P(w\,|\,d) = \sum_{z \in Z} P(w\,|\,z)P(z\,|\,d). \tag{3.6}$$ [2]

where *P*(*w*|*z*) denotes the topic-conditional probability of a term conditioned on the unobserved topic, and *P*(*z*|*d*) denotes a document-specific probability distribution over the latent variable space; that is, considering a vector $\vec{d}$ in latent variable space, *P*(*z*|*d*) denotes the weight of the latent variable *z*. Hence, a vector is mapped from term space to latent space and its size is reduced from |W| to |Z|.

PLSA model fitting (training)

A PLSA model is parameterized by *P*(*w*|*z*) and *P*(*z*|*d*) which are estimated by fitting *P*(*w*, *d*) to a training corpus *D*, i.e., *W*(*w*, *d*). The fitting process is obtained by maximizing the log-likelihood function *L* given below (Hofmann, 2001):

$$L = \sum_{w \in d} \sum_{d \in D} W(w,d) \log P(w,d) \tag{3.7}$$

The parameters of a PLSA model, *P*(*w*|*z*) and *P*(*z*|*d*), are estimated using the iterative Expectation-Maximization (EM) algorithm by maximizing the log-likelihood function *L*. *P*(*w*|*z*) and *P*(*z*|*d*) are initialized by random values in (0,1)-range. Then, the EM procedure iterates between the E-step and the M-step. In the E-step, the probability that a term *w* in a particular document *d* which is explained by the class corresponding to *z*, is estimated as

[2] It is assumed that the distribution of terms given a class is conditionally independent of the document, i.e., *P*(*w*|*z*,*d*) = *P*(*w*|*z*).

$$P(z \mid w,d) = \frac{P(z,w,d)}{P(w,d)} \tag{3.8}$$

$$P(z,w,d) = P(d)P(z \mid d)P(w \mid z)^{3}$$
(3.9)

Using Equations (3.6), (3.8) and (3.9), we can get

$$P(z \mid w,d) = \frac{P(d)P(z \mid d)P(w \mid z)}{P(d)\sum_{z'} P(w \mid z')P(z' \mid d)} = \frac{P(w \mid z)P(z \mid d)}{\sum_{z'} P(w \mid z')P(z' \mid d)} \tag{3.10}$$

In the M-step, we calculate

$$\begin{aligned} P(w \mid z) &= \frac{\sum_{d} W(w,d)P(z \mid w,d)}{\sum_{w'}\sum_{d} W(w',d)P(z \mid w',d)} \\ P(z \mid d) &= \frac{\sum_{w} W(w,d)P(z \mid w,d)}{\sum_{z'}\sum_{w} W(w,d)P(z' \mid w,d)} \end{aligned} \tag{3.11}$$

where parameters $P(w|z)$ and $P(z|d)$ are re-estimated to maximize L.

PLSA model testing

After training, the estimated $P(w|z)$ parameters are used to estimate $P(z|q)$ for new (test) documents q through a *folding-in* process (Hofmann, 2001). In the folding-in process, EM procedure runs in a similar manner to the training step. The E-step is identical but the M-step keeps all the $P(w|z)$ constant and only re-calculates $P(z|q)$. Usually, a very small number of iterations of the EM algorithm are sufficient for folding-in process.

Feature reduction by PLSA

[3] This equation is derived from according to the Figure 1(a) of Hofmann[39].

We apply PLSA not only for feature reduction but also for gapped-dipeptide semantic relation extraction. Vectors are mapped from the gapped-dipeptide space to the latent semantic space. This will lead to improvement in learning performance and efficiency. Though it is not easy to determine an appropriate reduced feature size of PLSA, it can be approximated by the reduced feature size of LSA. To determine the reduced feature size of LSA, we calculate singular values of LSA and sort them in decreasing order. Then, the reduced feature size of LSA equals to *n* if the *n*-th largest singular value is close to zero.

3.1.5 System architecture of PSLDoc

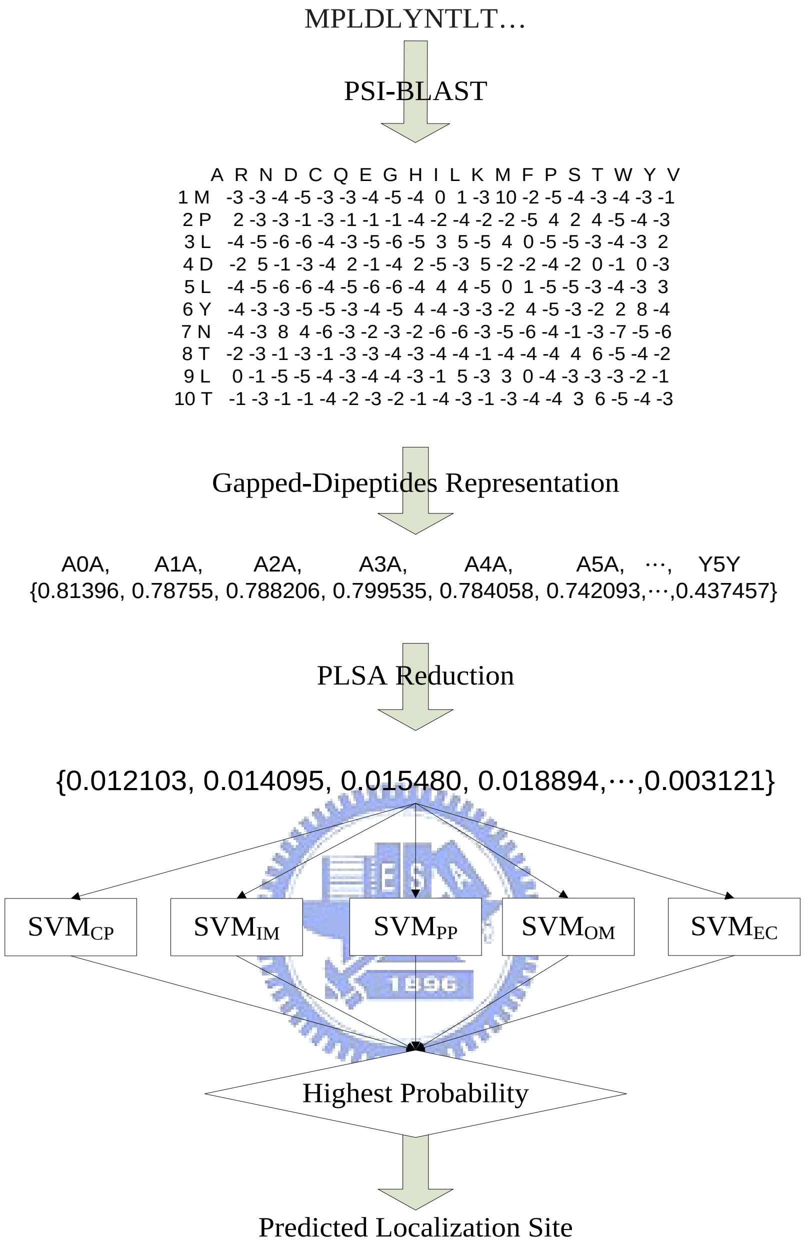
Prediction of PSL can be treated as a multiclass classification problem. For multiclass classification, the 1-v-r SVM model has demonstrated a good classification performance (Garg, et al., 2005). For each class *i*, we construct a 1-v-r (C_i versus non-C_i) binary classifier. PSLDoc consists of five 1-v-r SVM classifiers corresponding to five localization sites in Gram-negative bacteria. Input features for all binary classifiers are the same. The SVM program LIBSVM (Chang and Lin, 2001) is used in PSLDoc, and it can generate probability estimates that are used for determining the confidence levels of classifications (Wu, et al., 2004). For all classifiers, we use the Radial Basis Function kernel, and tune the cost (c) and gamma (γ) parameters optimized by 10-fold cross-validation on the training data set.

Given a protein, PSLDoc performs the following steps:

1. Use PSI-BLAST to generate PSSM of the protein.
2. Generate the feature vector of the protein, where each feature is defined as TFPSSM corresponding to a gapped dipeptide.
3. Perform PLSA to generate a reduced feature vector, which will be the input to each 1-v-r classifier.

4. Run five 1-v-r SVM classifiers.

In the training stage of PSLDoc, to train PLSA model with some topic size and the SVM classifiers, proteins with known localizations are used to estimate $P(w|z)$ and $P(z|d)$, and reduced vectors are used to determine the c and γ parameters of the RBF kernel of each classifier. In the testing stage of PSLDoc, Step 3 of PSLDoc performs PLSA folding-in process on trained $P(w|z)$. Step 4 of PSLDoc is performed on the trained SVM classifiers. The localization site of the protein is predicted as the class with the highest probability ($prob_i$: the confidence of the query protein predicted as class i; $0 \leqq prob_i \leqq 1$) generated from the five 1-v-r classifiers. The system architecture of PSLDoc is shown in Figure 3.2.



	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
1 M	-3	-3	-4	-5	-3	-3	-4	-5	-4	0	1	-3	10	-2	-5	-4	-3	-4	-3	-1
2 P	2	-3	-3	-1	-3	-1	-1	-1	-4	-2	-4	-2	-2	-5	4	2	4	-5	-4	-3
3 L	-4	-5	-6	-6	-4	-3	-5	-6	-5	3	5	-5	4	0	-5	-5	-3	-4	-3	2
4 D	-2	5	-1	-3	-4	2	-1	-4	2	-5	-3	5	-2	-2	-4	-2	0	-1	0	-3
5 L	-4	-5	-6	-6	-4	-5	-6	-6	-4	4	4	-5	0	1	-5	-5	-3	-4	-3	3
6 Y	-4	-3	-3	-5	-5	-3	-4	-5	4	-4	-3	-3	-2	4	-5	-3	-2	2	8	-4
7 N	-4	-3	8	4	-6	-3	-2	-3	-2	-6	-6	-3	-5	-6	-4	-1	-3	-7	-5	-6
8 T	-2	-3	-1	-3	-1	-3	-3	-4	-3	-4	-4	-1	-4	-4	-4	4	6	-5	-4	-2
9 L	0	-1	-5	-5	-4	-3	-4	-4	-3	-1	5	-3	3	0	-4	-3	-3	-3	-2	-1
10 T	-1	-3	-1	-1	-4	-2	-3	-2	-1	-4	-3	-1	-3	-4	-4	3	6	-5	-4	-3

Figure 3.2: System architecture of PSLDoc based on 1-v-r SVM models using reduced/transformed feature vectors.

Table 3.1: Number of proteins in different localization sites.

Localization sites	No.
Cytoplasmic (CP)	278
Inner membrane (IM)	309
Periplasmic (PP)	276
Outer membrane (OM)	391
Extracellular (EC)	190
All sites	1,444

3.1.6 Data sets

To evaluate the performance of PSLDoc, we utilize a benchmark data set of proteins from Gram-negative bacteria with single localization that have been used in previous works (Gardy, et al., 2005; Yu, et al., 2004). It consists of 1444 proteins with experimentally determined localizations, referred to as PS1444 (Rey, et al., 2005). Table 3.1 lists the distribution of localization sites of the data set.

To analyze the performance of PSLDoc under the effect of sequence homology information, we further classify each protein in PS1444 into two data sets, the high- or low-homology data sets based on whether or not the protein's highest sequence identity of all-against-all alignment by ClustalW is greater than an identity threshold of 30%. The high-homology data set, referred to as PSHigh783, consists of 783 proteins and the low-homology set, referred to as PSLow661, consists of 661 proteins. The three data sets are available at http://bio-cluster.iis.sinica.edu.tw/~bioapp/PSLDoc/DataSet.htm.

3.1.7 Evaluation measures

To evaluate the performance of our method, we follow the same measures used in previous works (Gardy, et al., 2003; Nakai and Kanehisa, 1991; Wang, et al., 2005; Yu, et al., 2004) for comparison with other approaches. These measures include accuracy (*Acc*), precision, recall, Matthew's correlation coefficient (*MCC*) (Matthews, 1975) for five localization sites, and the overall accuracy defined in Eq. (3.12), (3.13), (3.14), (3.15) and (3.16) below:

$$Acc_i = TP_i / N_i \tag{3.12}$$

$$Precision_i = TP_i /(TP_i + FP_i) \tag{3.13}$$

$$Recall_i = TP_i /(TP_i + FN_i) \tag{3.14}$$

$$MCC_i = \frac{(TP_i)(TN_i) - (FP_i)(FN_i)}{\sqrt{(TP_i + FN_i)(TP_i + FP_i)(TN_i + FP_i)(TN_i + FN_i)}} \tag{3.15}$$

$$Acc = \sum_{i=1}^{l} TP_i / \sum_{i=1}^{l} N_i \ , \tag{3.16}$$

where $l = 5$ is the number of localization sites, and TP_i, TN_i, FP_i, FN_i ,and N_i are the number of true positives, true negatives, false positives, false negatives, and proteins in localization site *i*, respectively. *MCC* considers both under- and over-predictions, and takes range from –1 to 1, where *MCC* = 1 indicates a perfect prediction; *MCC* = 0 indicates a completely random assignment; and *MCC* = -1 indicates a perfectly reverse correlation. The Acc_i is the same as Recall_i because N_i equals to the sum of TP_i and FN_i. We will use Acc_i or Recall_i interchangeably in the experiments depending on which method is compared.

3.1.8 Five simple PSL prediction methods

To evaluate the benefit of each step in our document classification method, we propose two simple prediction methods: 1NN_TFIDF and 1NN_TFPSSM, which consist of different parts of PSLDoc. To further analyze the effect of the PSSM information generated from databases of different sizes, we propose two methods based on PSI-BLAST: 1NN_PSI-BLAST$_{\text{ps}}$ and 1NN_PSI-BLAST$_{\text{nr}}$. In addition, we also construct a homology search method, 1NN_ClustalW, which is similar to Yu *et. al.*'s (Yu, et al., 2004) for comparison with PSLDoc.

1NN_TFIDF

1NN_TFIDF solely incorporates protein encoding scheme, the gapped-dipeptides of PSLDoc. The remaining steps are the same as the baseline system. That is, terms are weighted according to the TFIDF weighting scheme, and a query protein is predicted by 1-NN method based on cosine similarity.

1NN_TFPSSM

1NN_TFPSSM incorporates two parts of PSLDoc, the gapped-dipeptide encoding scheme and the TFPSSM weighting scheme. It predicts a query protein using 1-NN method based on cosine similarity.

1NN_PSI-BLAST$_{ps}$

1NN_PSI-BLAST$_{ps}$ performs two PSI-BLAST searches, one of which for generating a PSSM and the other for searching the most similar protein using the PSSM generated in the previous step. First, for each query protein, PSI-BLAST search is performed against the training data and its parameters are the same as those in PSLDoc. Then, 1NN_PSI-BLAST$_{ps}$ performs a one-run PSI-BLAST search (i.e., j = 1)[4] against the training data using the obtained PSSM[5]. Finally, the localization site of the protein with the highest e-value is assigned as the predicted localization for the query protein. In a five-fold cross-validation, the PSSM information used in

[4] The parameters of e-vlaue are ignored because we want to find the most similar protein instead of constructing a PSSM.

[5] Please refer to the last example on blastpgp's document for how to save a PSSM and perform PSI-BLAST search from the PSSM (http://biowulf.nih.gov/apps/blast/doc/blastpgp.html).

1NN_PSI-BLAST$_{ps}$ is generated from a small database which consists of approximately 1,155 (=1,444×4/5) sequences from PS1444.

1NN_PSI-BLAST$_{nr}$

Although 1NN_PSI-BLAST$_{ps}$ utilizes the PSSM information, the source database used is not as large as that of 1NN_TFIDF and PSLDoc. For fair comparison with 1NN_TFIDF and PSLDoc, we construct 1NN_PSI-BLAST$_{nr}$ which uses PSSM generated from the NCBI nr database. The only difference between 1NN_PSI-BLAST$_{nr}$ and 1NN_PSI-BLAST$_{ps}$ is the size of the databases searched in the first step, and the remaining steps are all the same, including the generation of PSSM, followed by performing a second PSI-BLAST search, and lastly, the prediction of the localization site of the query protein.

1NN_ClustalW

1NN_ClustalW differs from Yu *et al.*'s method (Yu, et al., 2004) only in the pairwise sequence alignment algorithm used, i.e., ClustalW in the former and ALIGN in the latter. For a query protein, we calculate its pairwise sequence identities with the remaining proteins by performing 1-against-others pairwise sequence alignment. Then, the localization site of the query protein is predicted by the 1-NN method based on pairwise sequence identity, that is, its localization site is assigned as that of the protein whose pairwise sequence identity is highest.

3.1.9 Experiment design

We conduct the following experiments to evaluate the benefit of each step in our document classification model where the gapped distance upper bound, *l*, ranges from 3 to 15. We follow the same validation procedures for the performance measurement as those of the other approaches (Gardy, et al., 2005; Yu, et al., 2006). All experiments are carried out in five-fold cross-validation, that is, the data is equally divided into five parts. In each run, four folds are used for training and the remaining fold is used for testing. All reported results are average over the five folds. We have conducted the following six experiments:

Experiment 1: Comparison between 1NN_TFIDF and 1NN_TFPSSM on the PS1444, PSHigh783, and PSLow661 data sets

The purpose of this experiment is to evaluate the benefit of using the TFPSSM weighting scheme because the simple 1NN prediction method can reflect the relation between performance and weighting schemes avoiding the effect of the prediction algorithm. The distribution of benefit among 1444 protein sequences is further analyzed by comparing their performance on PSHigh783 and PSLow661.

Experiment 2: Comparison among 1NN_TFPSSM, 1NN_ClustalW, 1NN_PSI-BLAST$_{ps}$, and 1NN_PSI-BLAST$_{nr}$ on the PSHigh783 and PSLow661 data sets

To compare the effect of utilizing PSSM, we compare the performance of 1NN_TFPSSM, 1NN_ClustalW, 1NN_PSI-BLAST$_{ps}$, and 1NN_PSI-BLAST$_{nr}$. 1NN_ClustalW is based on a pairwise sequence alignment in which no PSSM information is incorporated. We further analyze the relationship between the effect of PSSM and the size of databases used in the construction of PSSM. Compared with 1NN_PSI-BLAST$_{ps}$, both 1NN_TFPSSM and 1NN_PSI-BLAST$_{nr}$ incorporate a larger database for PSSM construction. Finally, the comparison between 1NN_TFPSSM and 1NN_PSI-BLAST$_{nr}$ serves to highlight the benefit of gapped-dipeptide encoding scheme.

Experiment 3: Comparison between PSLDoc and PSLDoc$_{-PLSA}$ on the PS1444 data set

PSLDoc$_{\text{-PLSA}}$ represents PSLDoc without PLSA, which simply applies SVM on the original feature vectors. The overall accuracies of PSLDoc and PSLDoc$_{\text{-PLSA}}$ are compared in order to evaluate the benefit of PLSA feature reduction for SVM learning.

Experiment 4: Comparison among PSLDoc, 1NN_TFPSSM, and 1NN_ClustalW on the PSHigh783 and PSLow661 data sets

Using the PSHigh783 data set, we can verify whether PSLDoc can replace 1NN_ClustalW. Using PSLow661, we can investigate whether PSLDoc can improve 1NN_TFPSSM by applying PLSA and SVM classification. Hence, we could determine whether PSLDoc is suitable for both high- and low-homology data sets.

Experiment 5: Comparison among PSLDoc, HYBRID, and PSORTb v.2.0 on the PS1444 data set

We compare the performance of PSLDoc, HYBRID, and PSORTb v.2.0. Besides, we also assess the performance of PSLDoc using a three-way data split procedure (Ritchie, et al., 2003) which is usually used in machine learning to prevent overestimation of the performance. The data set is randomly divided into three disjoint sets, that is, a training set for classifier learning, a validation set for feature selection and parameter tuning, and a test set for performance evaluation. Hence, for each run in the original five-fold cross-validation, we divide the training data set into four distinct sets: three for training, one for validation. Then, we select the gapped distance upper bound and PLSA reduced feature size based on validation set instead of test set. Then PSLDoc performance is evaluated under the selected parameters in the original five-fold cross-validation.

Experiment 6: PSLDoc under Prediction Threshold versus PSORTb v.2.0 on the PS1444 data set

The precision and recall of PSLDoc is evaluated under different prediction thresholds to compare with PSORTb v.2.0.

3.2 Results

3.2.1 Experimental results

Experiment 1: The benefit of using the TFPSSM weighting scheme

The overall accuracy of 1NN_TFIDF and 1NN_TFPSSM for each gapped distance are shown in Figure 3.3. The highest overall accuracy of 1NN_TFPSSM is 89.47% when *l* equals 4, 5, and 13 and it is considerably higher than the best 1NN_TFIDF score 74.38% when *l* equals to 4. Therefore, adopting the TFPSSM weighting scheme significantly improves the performance of 1NN_TFIDF.

The performance of 1NN_TFIDF and 1NN_TFPSSM in the high- and low-homology data sets is shown in Table 3.2. 1NN_TFPSSM dramatically improves the performance of 1NN_TFIDF by about 26% in overall accuracy on PSLow661. Hence, incorporation of PSSM in the weighting scheme is useful for improving performance due to insufficient sequence information in the low-homology data set.

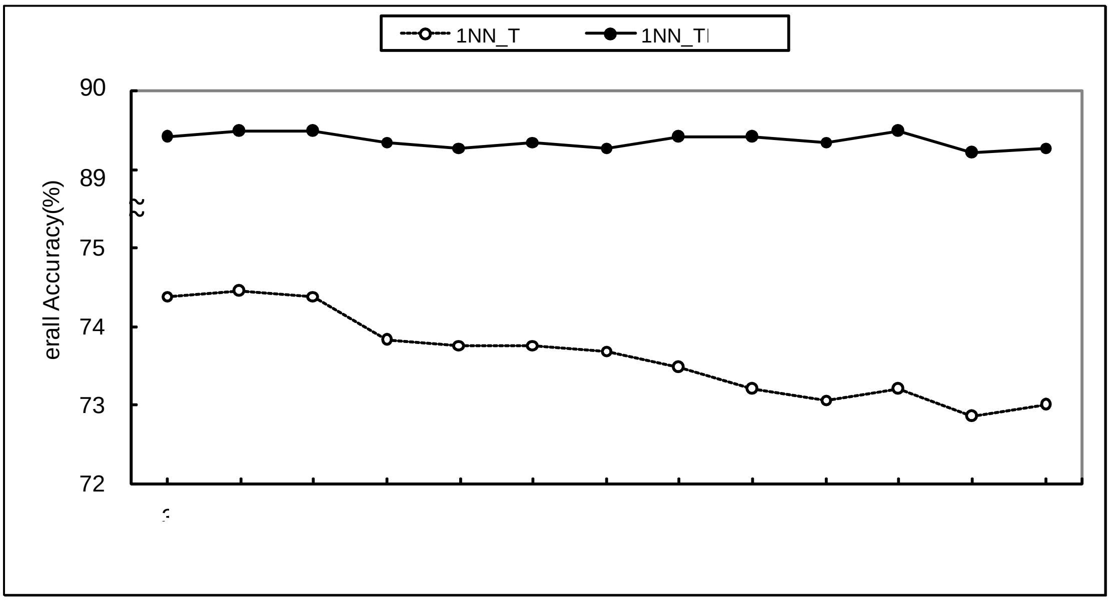


Figure 3.3: Overall accuracy of 1NN_TFIDF and 1NN_TFPSSM with respect to gapped distances on the PS1444 data set.

Table 3.2: The comparison of 1NN_TFIDF and 1NN_TFPSSM on the PSHigh783 and PSLow661 data sets.

	PSHigh783				PSLow661			
	1NN_TFPSSM		1NN_TFIDF		1NN_TFPSSM		1NN_TFIDF	
Loc. Sites	*Acc.*(%)	*MCC*	*Acc.* (%)	*MCC*	*Acc.* (%)	*MCC*	*Acc*(%).	*MCC*
CP	94.20	0.96	71.01	0.74	83.25	0.77	41.15	0.36
IM	99.31	0.99	98.62	0.89	82.93	0.82	84.15	0.48
PP	95.86	0.94	86.21	0.89	74.05	0.63	38.17	0.46
OM	99.66	0.99	95.88	0.95	85	0.82	66.00	0.48
EC	96.99	0.96	92.48	0.91	57.89	0.51	28.07	0.26
Overall	97.96	-	91.83	-	79.43	-	53.86	-

Experiment 2: The effect of incorporating PSSM information and gapped-dipeptide encoding scheme

Table 3.3 shows the performance of 1NN_TFPSSM, 1NN_ClustalW, 1NN_PSI-BLAST$_{ps}$, and 1NN_PSI-BLAST$_{nr}$ on the PSHigh783 and PSLow661 data sets. The overall accuracy on the PSHigh783 data set is very similar for all methods. However, for the PSLow661 data set, 1NN_ClustalW, 1NN_PSI-BLAST$_{ps}$, and 1NN_PSI-BLAST$_{nr}$ attain 42.97%, 57.94% and 66.57%, respectively, in overall accuracy. This result reveals that better performance can be achieved when a larger database is used in constructing PSSM. This also lends support to our assumption that incorporating more information into PSSM is more effective for the prediction of proteins with low sequence identity to the training set. Most notably, 1NN_TFPSSM outperforms 1NN_PSI-BLAST$_{nr}$ by 12.86% in overall accuracy. This suggests that the incorporation of PSSM based on gapped-dipeptide encoding scheme significantly improves the predictive performance, especially for proteins of low sequence identity.

Table 3.3: Comparison of 1NN_TFPSSM, 1NN_ClustalW, 1NN_PSI-BLAST$_{ps}$ and 1NN_PSI-BLAST$_{nr}$ for the PSHigh783 and PSLow661 data sets

PSHigh783

Loc.	1NN_TFPSSM		1NN_ClustalW		1NN_PSI-BLAST$_{ps}$		1NN_PSI-BLAST$_{nr}$	
Sites	*Acc.*(%)	*MCC*	*Acc.*(%)	*MCC*	*Acc.*(%)	*MCC*	*Acc.*(%)	*MCC*
CP	94.20	0.96	89.86	0.90	88.41	0.92	86.96	0.90
IM	99.31	0.99	98.62	0.97	99.31	0.98	99.31	0.98
PP	95.86	0.94	93.79	0.93	93.79	0.93	92.41	0.91
OM	99.66	0.99	99.66	0.99	99.66	0.99	99.66	0.99
EC	96.99	0.96	98.50	0.98	98.50	0.98	98.50	0.98
Overall	97.96	-	97.32	-	97.32	-	96.93	-

PSLow661

Loc.	1NN_TFPSSM		1NN_ClustalW		1NN_PSI-BLAST$_{ps}$		1NN_PSI-BLAST$_{nr}$	
Sites	*Acc.*(%)	*MCC*	*Acc.*(%)	*MCC*	*Acc.*(%)	*MCC*	*Acc.*(%)	*MCC*
CP	83.25	0.77	39.23	0.23	36.84	0.40	55.50	0.53
IM	82.93	0.82	46.95	0.33	68.29	0.57	75.00	0.66
PP	74.05	0.63	41.98	0.44	59.54	0.51	64.12	0.54
OM	85.00	0.82	45.00	0.47	87.00	0.57	87.00	0.66
EC	57.89	0.51	43.86	0.10	50.88	0.37	52.63	0.45
Overall	79.43	-	42.97	-	57.94	-	66.57	-

Experiment 2: The benefit of PLSA feature reduction

Determine the reduced size of PLSA

The size of PLSA is determined by LSA singular values. Figure 3.4 show the singular values in decreasing order on different gapped distances upper bound data sets.

The 40-th largest singular value is close to zero in Figure 3.4, but in the inset the 160-th largest singular value is close to zero. Hence, the reduced feature size of PLSA is set to 40, 80 and 160. However, we do not test larger PLSA reduced size or one-by-one PLSA reduced size in consideration of the training efficiency and avoidance of data overfitting.

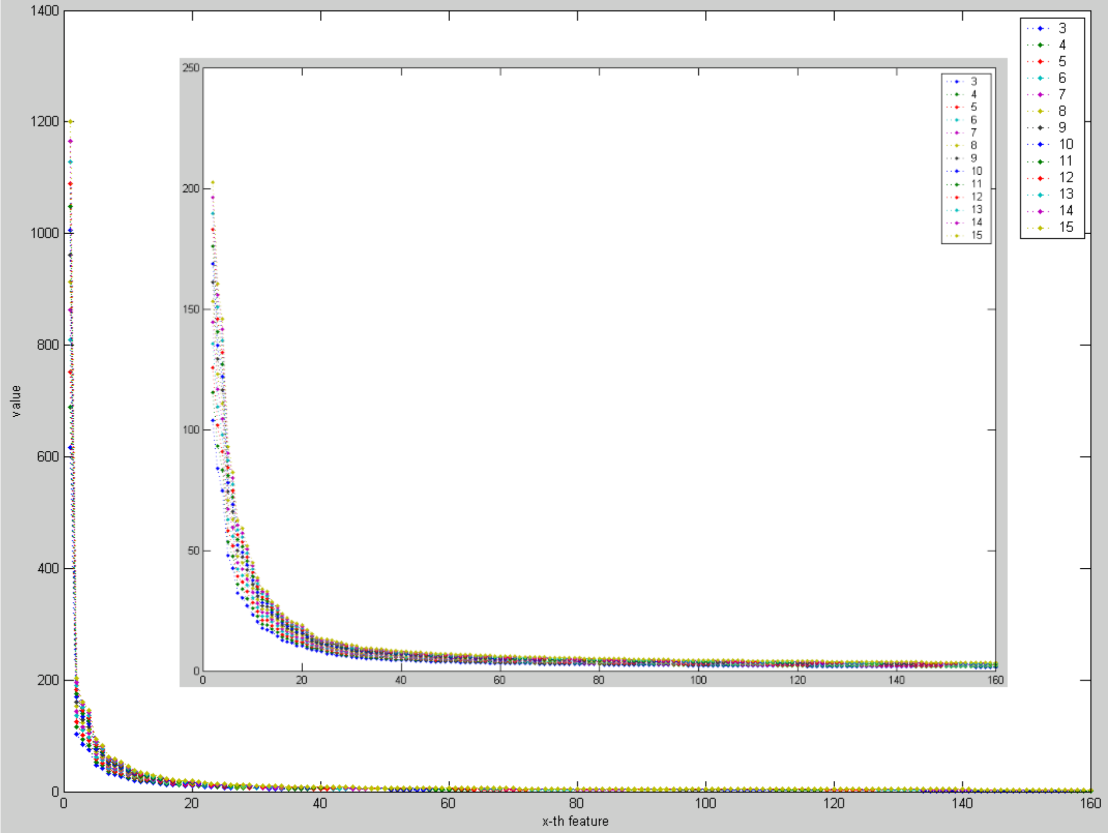


Figure 3.4: Singular values in decreasing order of each gapped distance. The inset shows singular values without 1-th largest one for detailed representation.

For one PLSA reduced size, the training and testing procedures of PSLDoc take 1.5 hours and about 2~3 minutes for all gapped distances, respectively. However, PSLDoc$_{\text{-PLSA}}$ takes about 180 and 1.4 hours in training and testing, respectively. Figure 3.5 shows the performance of PSLDoc$_{\text{-PLSA}}$ and PSLDoc, where PSLDoc_F*x* denotes PSLDoc with PLSA reduced size *x*.

The highest overall accuracy among all gapped distances of PSLDoc_F40, PSLDoc_F80, and PSLDoc_F160 is 92.31%, 93.01%, and 92.52%, respectively, which is 0.83%, 1.52%, and 1.04% better than that of PSLDoc$_{\text{-PLSA}}$. Using PLSA not only improves learning efficiency but also performance. In the following experiments, PSLDoc takes the gapped distance 13 and PLSA at reduced size 80.

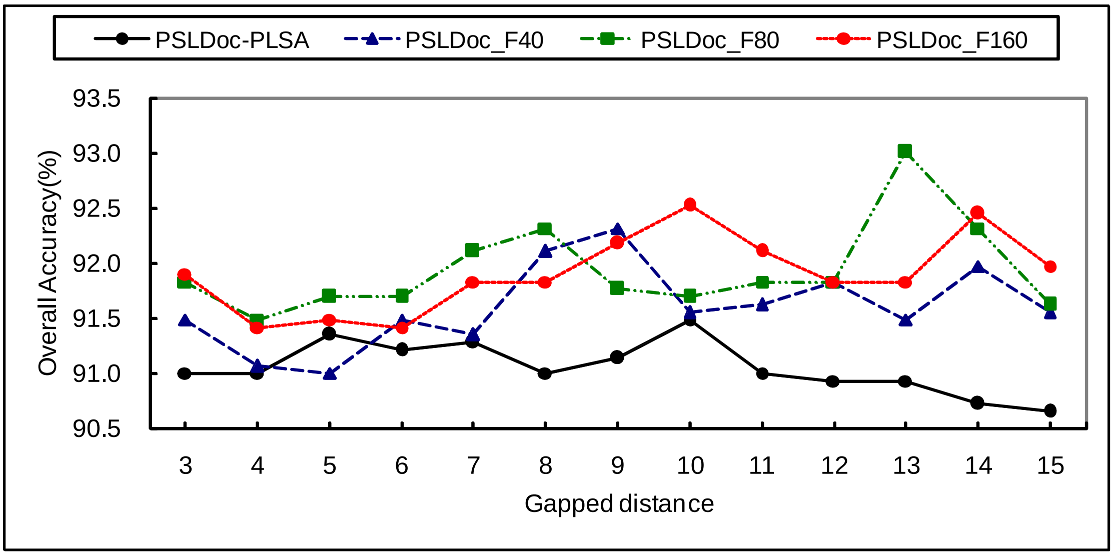


Figure 3.5: Overall accuracy of PSLDoc_F40, PSLDoc_F80, PSLDoc_F160 and PSLDoc-PLSA with respect to gapped distance on the PS1444 dataset.

Experiment 4: The benefit of SVM and PLSA feature reduction

Table 3.4 shows the performance of PSLDoc, 1NN_TFPSSM and 1NN_ClustalW on PSHigh783 and PSLow661. The overall accuracy of 1NN_ClustalW on PSHigh783 (97.32%) is very similar to that of Yu *et. al.*'s (97.7%). 1NN_TFPSSM and PSLDoc perform better than 1NN_ClustalW on PSHigh783. On the other hand, PSLDoc improves 1NN_TFPSSM on PSLow661 by 7.41% due to the non-linear SVM classification and PLSA feature reduction and extraction. This shows that PSLDoc is suitable for both the high- and low-homology data sets.

Table 3.4: Comparison of PSLDoc, 1NN_TFPSSM, and 1NN_ClustalW for the PSHigh783 and PSLow661 data sets.

	PSHigh783						PSLow661					
	PSLDoc		1NN_TFPSSM		1NN_ClustalW		PSLDoc		1NN_TFPSSM		1NN_ClustalW	
Loc.	*Acc.*(%)	*MCC*	*Acc.*(%)	*MCC*	*Acc.*(%)	*MCC*	*Acc.*(%)	*MCC*	*Acc.*(%)	*MCC*	*Acc.*(%)	*MCC*
CP	95.65	0.96	94.2	0.96	91.3	0.89	94.74	0.88	83.25	0.77	39.23	0.23
IM	99.31	0.99	99.31	0.99	97.93	0.97	87.80	0.88	82.93	0.82	46.95	0.33
PP	95.17	0.94	95.86	0.94	93.1	0.93	82.44	0.78	74.05	0.63	41.98	0.44
OM	99.66	0.99	99.66	0.99	99.66	0.99	84.00	0.84	85.00	0.82	45.00	0.47
EC	98.5	0.98	96.99	0.96	99.25	0.99	70.18	0.65	57.89	0.51	43.86	0.10
Overall	98.21	-	97.96	-	97.32	-	86.84	-	79.43	-	42.97	-

Table 3.5: Comparison of PSLDoc, HYBRID and PSORTb v.2.0 on the PS1444 data sets. The PSLDoc performance of incorporating a three-way data split procedure is indicated in the parentheses.

	PSLDoc		HYBRID		PSORTb v.2.0	
Loc. Sites	*Acc.*	*MCC*	*Acc.*	*MCC*	*Acc.*	*MCC*
CP	94.96(94.24)	0.91(0.91)	95.00	0.89	70.10	0.77
IM	93.20(93.53)	0.94(0.94)	90.60	0.92	92.60	0.92
PP	89.13(89.13)	0.87(0.85)	88.80	0.84	69.20	0.78
OM	95.65(95.14)	0.95(0.94)	95.10	0.93	94.90	0.95
EC	90.00(87.37)	0.87(0.86)	85.30	0.87	78.90	0.86
Overall	93.01(92.45)	-	91.60	-	82.60	-

Experiment 5: Comparison of PSLDoc, HYBRID and PSORTb v.2.0

Table 3.5 shows the performance of PSLDoc, HYBRID, and PSORTb v2.0 on PS1444. PSLDoc achieves the best performance 93.01%, better than HYBIRD 91.6% and PSORTb 82.6%.

Experiment 6: PSLDoc under different prediction thresholds versus PSORTb v.2.0 on the PS1444 data set

Prediction confidence

The probability estimated by LIBSVM is used for determining the confidence levels of classifications. The class with the largest probability is chosen as the final predicted class. The confidence of the final predicted class, *prediction confidence* (Jones, 1999), could be regarded as the value of the largest probability minus the second largest probability. Figure 3.6 shows the relationship between accuracy and prediction confidence. For proteins with prediction confidence in the range [0.9-1], the prediction accuracy is near 100% (99.12%).

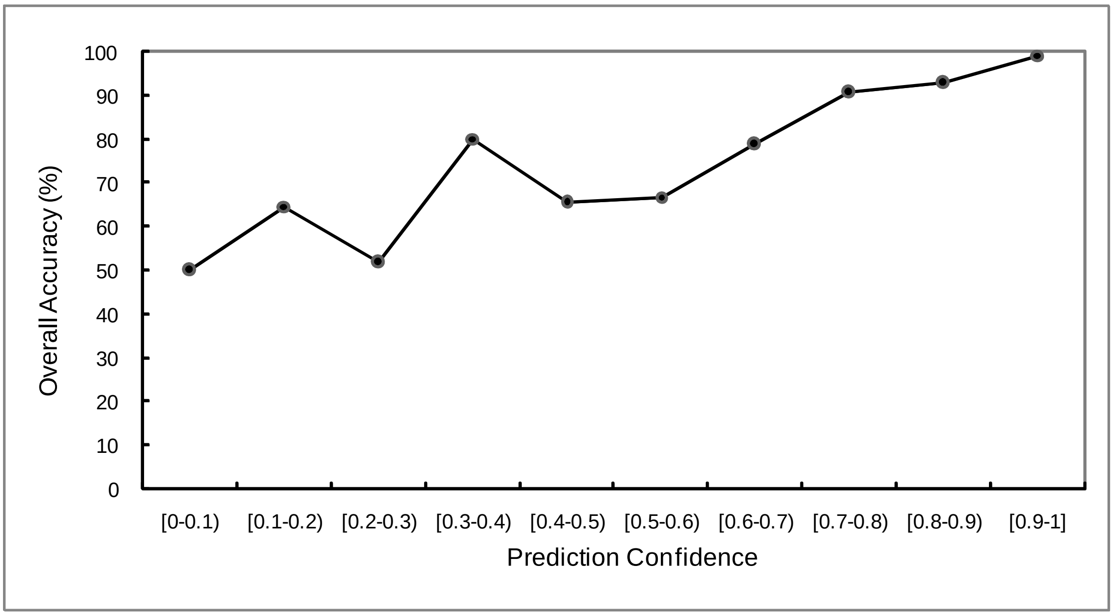


Figure 3.6: Overall accuracy of PSLDoc with respect to prediction confidence. [x,y) represents the prediction confidence is more than x but under y.

Prediction threshold

Gardy *et al.* suggested that when a prediction system is unable to generate a confident prediction, the program had better report a result of "*Unknown*" because biologists usually prefer correct prediction (high precision) to prediction coverage (recall) (Gardy, et al., 2005). To provide more precise prediction results, we determine a *prediction threshold* to filter out prediction results with low confidence. That is, the SVM classifier predicts results only when the prediction confidence is above the threshold, otherwise the SVM classifier will output "Unknown" (Gardy, et al., 2005; Gardy, et al., 2003). Recall and precision for each prediction threshold are shown in Figure 3.7.

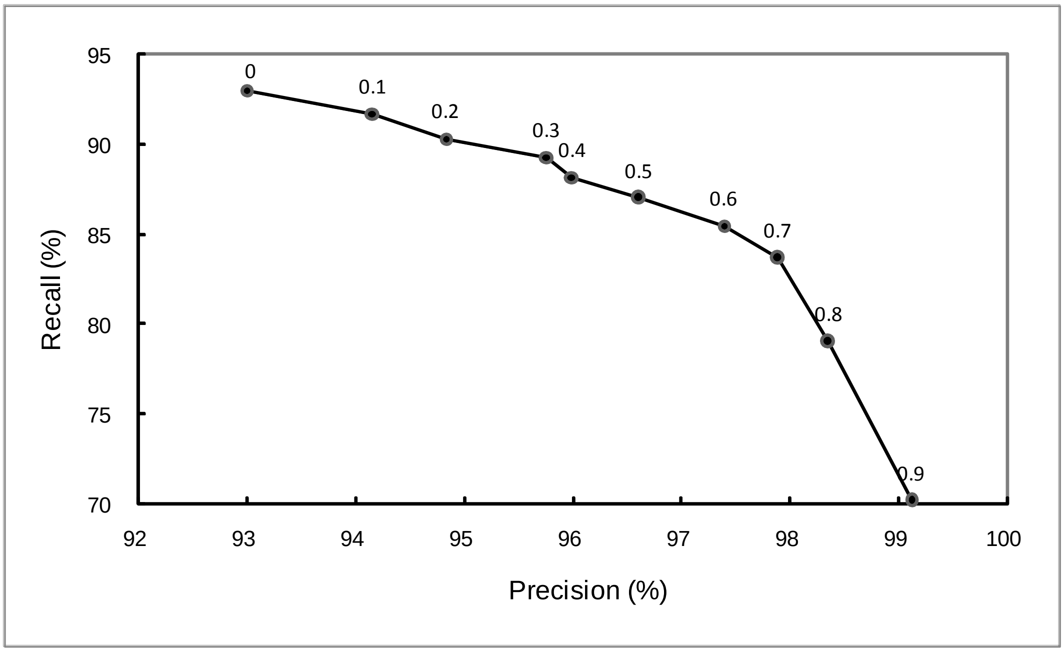


Figure 3.7: Overall accuracy of PSLDoc with respect to prediction confidence. The value above the point denotes the corresponding prediction threshold.

Table 3.6 shows the performance of PSLDoc under different prediction thresholds. Setting the prediction threshold to 0.7, PSLDoc achieves slightly better recall than PSORTb v.2.0 (83.66% versus 82.6%), whereas the precision of PSLDoc is better than PSORTb v.2.0 (97.89% versus 95.8%). Besides, when the prediction threshold is set to 0.3, PSLDoc achieves comparable precision to PSORTb v.2.0 (95.77% vs. 95.8%), and PSLDoc's recall is much better than that of PSORTb v.2.0 (89.27% vs. 82.6%).

Table 3.6. Comparison of PSLDoc under the prediction threshold 0.7, PSLDoc under the prediction threshold 0.3 and PSORTbv.2.0

Loc.	PSLDoc_PreThr=0.7					PSLDoc_PreThr=0.3					PSORTb v.2.0				
	TP	FP	FN	Pre.	Rec.	TP	FP	FN	Pre.	Rec.	TP	FP	FN	Pre.	Rec.
CP	216	6	62	97.30	77.70	243	13	35	94.92	87.41	195	15	83	92.86	70.14
IM	273	3	36	98.91	88.35	285	6	24	97.94	92.23	286	14	23	95.33	92.56
PP	202	8	74	96.19	73.19	226	17	50	93.00	81.88	191	9	85	95.50	69.20
OM	366	2	25	99.46	93.61	372	6	19	98.41	95.14	371	10	20	97.38	94.88
EC	151	7	39	95.57	79.47	163	15	27	91.57	85.79	150	4	40	97.40	78.95
Total	1208	26	236	97.89	83.66	1289	57	155	95.77	89.27	1193	52	251	95.82	82.62

3.3 Discussion

In PLSA, we associate proteins and gapped-dipeptides with topics. Through analyzing the trained PLSA model with $P(w|z)$ and $P(z|d)$ for gapped-dipeptide w, topic z and protein d, gapped-dipeptide signatures in proteins with different localization sites are discovered for the PS1444 data set. Some of these signatures have been reported in the literature as motifs critical for stability or localization. We also discuss the problem of polysemy and solve it through the PLSA model.

3.3.1 Gapped-dipeptide signatures for Gram-negative bacteria localization sites

In Figure 3.8, we show the distribution of topic versus protein as visualized by $P(z|d)$ for topic $z \in Z$ and protein $d \in D$. In the figure, the size of topic ($|Z|$) is set to 80 according to the conclusion from Experiment 2.

To find site-topic preference, we then cluster proteins according to their localization sites for examining preferred topics for each localization site. The *site-topic preference* of the topic z for a localization site l is calculated by averaging $P(z|d)$, where d (a protein) belongs to l class (i.e., d has localization site l.) The site-topic preference over topics per localization site is shown in Figure 3.9. We can observe from the figure that topics can be divided into five groups such that each group “prefers” a specific localization site.

We say a topic z *prefer*s a localization size l, if the corresponding site-topic preference is the largest of all localization sites. For some topics preferring PP and EC classes, the difference of the site-topic preference between their own preferring site and other sites are not obvious in Figure 3.9. This also reflects the relative poor performance of PSLDoc in PP and EC classes.

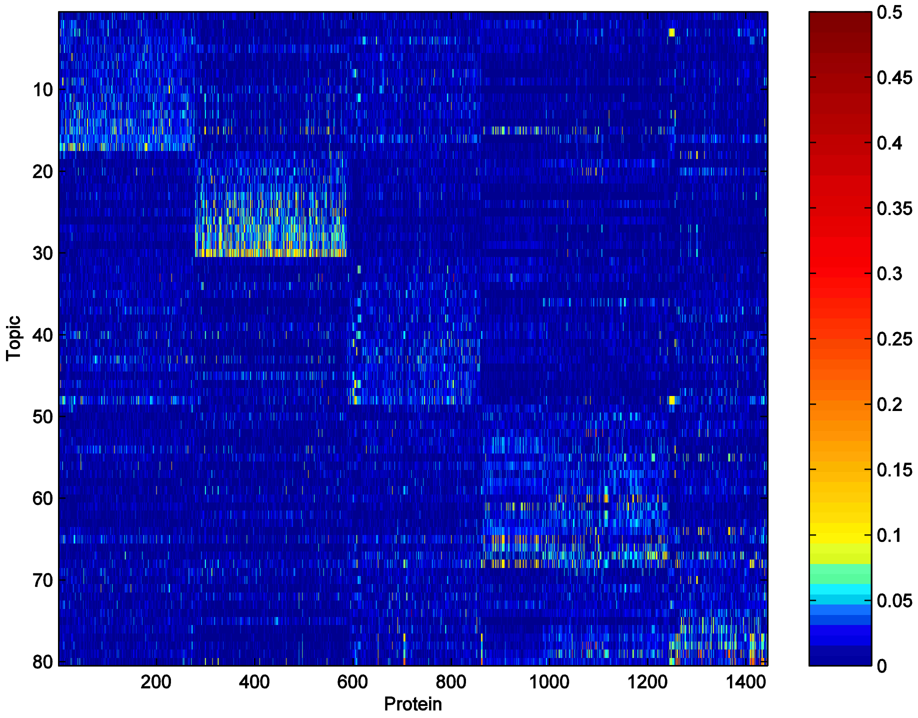


Figure 3.8: Distribution of topic versus protein plotted as an image with its colormapd, where the topics are sorted such that topics "preferring" (to be explained in the third paragraph) the same localization site are grouped together.

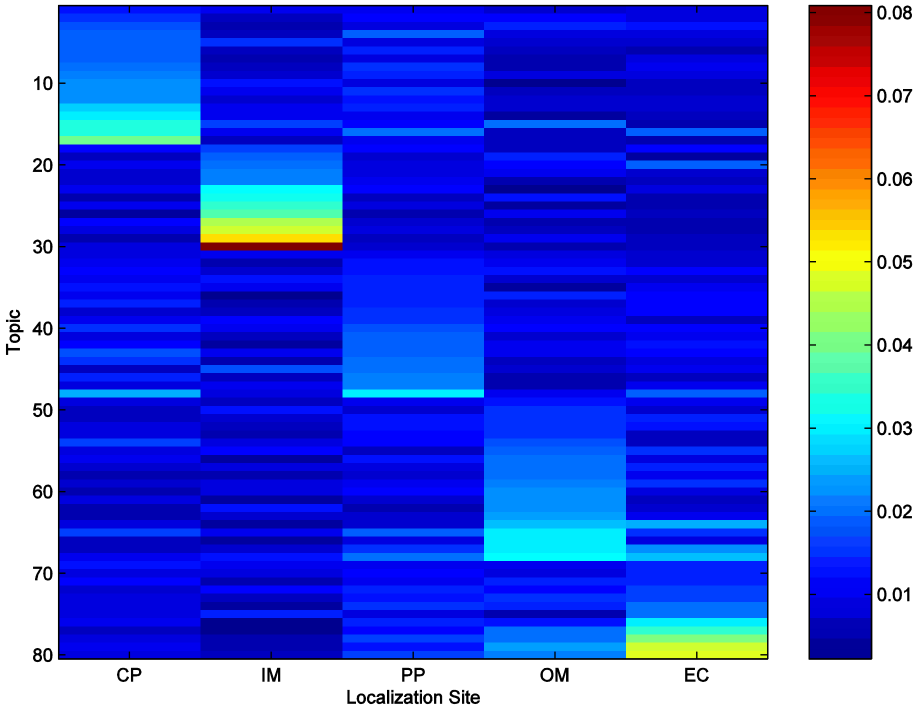


Figure 3.9: Distribution of site-topic preference versus localization site. The 80 topics are divided into five groups of 17, 13, 18, 20 and 12 topics that prefer CP, IM, PP, OM and EC, respectively.

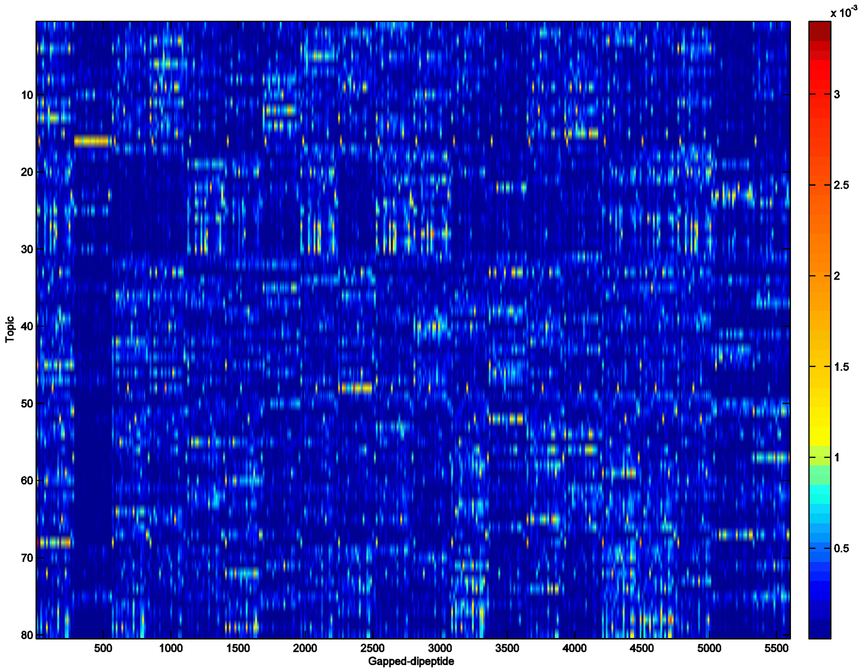


Figure 3.10: Distribution of topic versus gapped-dipeptide.

The distribution of topic versus gapped-dipeptide is visualized by $P(w|z)$ for gapped-dipeptide $w \in W$ and topic $z \in Z$ as shown in Figure 3.10. In the figure, the size of gapped-dipeptides ($|W|$) is set to 5,600 (=14×20×20) following the conclusion of Experiment 2.

To list gapped-dipeptides of interest, we select ten preferred topics for each localization site according to *site-preference confidence*, which is defined as the largest site-topic preference minus the second largest site-topic preference. For each topic, five most frequent gapped-dipeptides are selected. We list the gapped-dipeptides signatures of ten preferred topics corresponding to each of the localization sites in Table 3.7.

Table 3.7: Gapped-dipeptide signatures for each Gram-negative bacteria localization site.

Site	Gapped-dipeptide signatures		
CP	E0E, K1I, K5V, K1V, D0E;	L1H, L5H, L3H, H4L, H0L;	A12C, A9C, A13C, A5C, A7C;
	R3R, R6R, R2R, R0R, R9R;	A6A, A13A, A7A, A10A, A11A;	I0E, R6I, I3R, I3K, R6V;
	H3H, H1H, H7H, H13H, H10H;	H1M, H2M, H11M, M0H, H0M;	A4E, E1E, A2E, V4E, A9E;
	E4E, K6E, E6E, E3E, E0E		
IM	I2I, I3I, I0I, L0I, I0F;	L7L, L4L, L10L, L3L, L6L;	M3M, M2M, M0M, M8M, M6M;
	V2I, V2V, V3I, V3V, I0V;	T2F, T6F, F3F, T4F, T8F;	A1A, A7L, A4A, A1C, A11L;
	W3W, W0W, W2W, W6W, W4W;	Y12L, Y1L, Y11L, L0Y, L1L;	M2T, M3T, M10T, M4T, M0L;
	F10P, F8P, F12P, F3P, F13P		
PP	A1A, A2A, A0A, A3A, M4A;	M0H, W1Q, W1H, W1K, W5Q;	P1E, P0E, E0P, P0K, E1P;
	D0D, Q0D, D3D, D3Q, D11D;	W0E, E4W, W11E, E0W, W13E;	K3K, K0K, K2K, K1K, K7K;
	A3A, A7A, A1P, A6R, A10R;	P3N, N4P, N3P, N5P, N0P;	H6G, G3M, H7D, G11H, H11G;
	A10A, A11A, A6A, A12A, A3A		
OM	T1R, R3T, R1T, T5R, P0P;	R0F, R4F, Y13R, R6F, R2F;	N4N, N0N, N10N, N7N, F1N;
	Q6Q, Q1Q, Q3Q, Q13Q, Q4Q;	S0F, A3F, F0S, R9F, F7F;	G0G, A0G, A1G, G1A, G3A;
	N1Q, N1N, Q1Q, N12N, Q11V;	W2N, N2W, N0W, D2W, N13W;	Q5R, R1Q, Q1R, Q3R, R2Q;
	Y1Y, Y0Y, Y5Y, Y4Y, Y12Y		
EC	S6S, S2S, T11T, S13S, T6S;	G8G, G0G, G7G, G9G, G6G;	T1T, T3T, T5T, T9T, T10T;
	N10N, N9N, N13N, N11N, N12N;	N1N, N3N, N4N, N11N, N1T;	I5Y, Y12S, Y3S, Y9S, Y6I;
	Q2N, N1Q, Q1Q, N3Q, Q7Q;	K1S, S6S, S5S, S11M, S0S;	S3G, G3G, G4S, G3S, G2G;
	N0N, N12V, N4V, V12N, N9V		

3.3.2 Gapped-dipeptide signatures for Gram-negative bacteria localization sites

Interestingly, some of the signatures in Table 3.7 found by PSLDoc have been reported in the literature as motifs critical for stability or localization. One example is observed in the integral membrane (IM) proteins, in which helix-helix interactions are stabilized by aromatic residues(Sal-Man, et al., 2007). Specifically, the aromatic motif (WXXW or W2W) is involved in the dimerization of transmembrane (TM) domains by π-π interactions(Sal-Man, et al., 2007). Remarkably, one preferred topic predicted for the IM class includes this motif (W2W) among other signatures of aromatic residues. Another example is found in the outer membrane (OM) class, where the C-terminal signature sequence is recognized by the assembly factor, OMP85, regulating the insertion and integration of OM proteins in the outer membrane of gram-negative bacteria(Robert, et al., 2006). The C-terminal signature sequence contains a Phe (F) at the C-terminal position, preceded by a strong preference for a basic amino acid (K, R)(Robert, et al., 2006). One of the preferred topics indeed contains this motif (R0F.)

The above findings demonstrate the sensitivity of PSLDoc for capturing gapped-dipeptide signatures relevant to localization sites. Thus, the predicted signatures can provide important clues for further studies of uncharacterized sequence motifs related to protein localization.

3.3.3 Comparison of gapped-dipeptide signature encoding and amino acid composition

Figure 3.11 shows the amino acid compositions of single residues and gapped-dipeptide signatures for each localization site, respectively. It is observed that the distributions of 20 amino acids calculated from single residues and gapped-dipeptide signatures are quite different. The distribution from single residues [Fig. 11(A)] has no clear separation for some amino acids but the distribution from gapped-dipeptide signatures [Fig. 11(B)] has a clear separation among five classes.

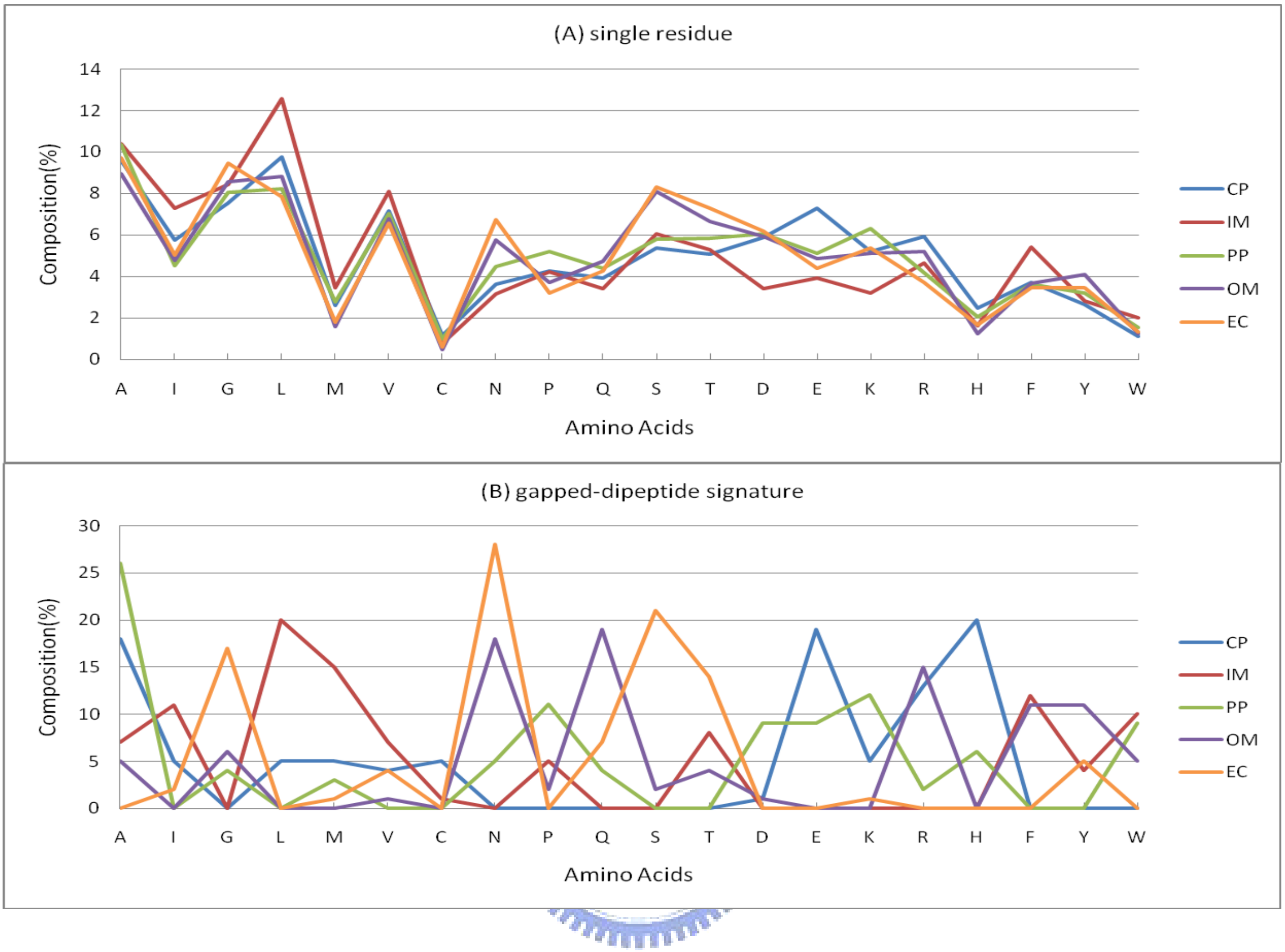


Figure 3.11: The amino acid compositions of single residues (A) and selected gapped-dipeptide signatures (B) in different localization sites.

From Fig. 11(A) and (B), it is observed that for some amino acids, general amino acid composition bias have an effect on the gapped-dipeptide signatures (e.g., CP: E; IM: I, L; PP: P, K; OM: Y; EC: G, N). That is, amino acids having high composition in a localization site tend to also have high composition in gapped dipeptide signatures of the localization site. For example, there are relatively high proportions for Ile and Leu in both single residue and gapped-dipeptide signature compositions in IM proteins. However, many amino acids have high compositions in at least two localization sites. Therefore, it is difficult to predict localization site based on single residue compositions. From the amino acid composition of gapped-dipeptide signatures, we observe a clear separation among different localizations for several amino acids, which are indistinguishable at the single residue level (i.e., A, M, V, Q, S, H, W). Specifically, Met, Val, and Trp have similar proportions across all five localizations in single residue composition. The small differences in single amino acid composition for these residues are amplified by examining the gapped-dipeptide signature compositions and thus, they can be used for predicting localization site in a discriminative manner. We further analyze the correlation between single amino acid and gapped-dipeptide signature compositions by the Pearson correlation coefficient whose definition for a series of *n* measurements of variables *X* and *Y* is as follows:

$$r_{xy} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^{\,2} - \left(\sum x_i\right)^2}\sqrt{n\sum y_i^{\,2} - \left(\sum y_i\right)^2}} \tag{3.17}$$

The Pearson correlation coefficient (r) between the two compositions (single residues vs. gapped dipeptide signatures) for CP, IM, PP, OM, EC and all localization sites are 0.29, 0.50, 0.41, 0.07, 0.50 and 0.36, respectively. The correlation for all localization sites is medium (in range 0.30 to 0.49)(Cohen, 1988).

In summary, the gapped-dipeptide signatures predicted by PSLDoc can (1) successfully capture the compositional bias inherent at the single residue level; and (2) better resolve ambiguity in discriminating amino acid compositions for each localization site.

3.3.4 The physicochemical preference of gapped-dipeptide signatures

To further analyze the physicochemical preference of gapped-dipeptide signatures, each amino acid is classified into one of the four groups: non-polar (AIGLMV), polar (CNPQST), charged (DEHKR), and aromatic (FYW). Figure 3.12 shows the grouped amino acid compositions of single residues and gapped-dipeptide signatures for each localization class. The grouped amino acid composition of single residues for each localization site has very similar preferences, but different preferences are observed for gapped-dipeptide signature composition. For example, in Fig. 12(A), IM, PP, OM, and EC have similar distribution, but in Fig. 12(B), each localization has distinct distribution of grouped amino acid composition. This also lends support to the second point in the previous section, that gapped-dipeptide signature can better resolve ambiguity in discriminating amino acid compositions for each localization. Furthermore, our analysis shows that the amino acid compositions of predicted gapped-dipeptide signatures exhibit some over-represented patterns for a particular compartment.

Gapped-dipeptide signatures predicted for CP, IM, and EC classes have distinct preferences for different groups of amino acids, possibly reflecting the physico-chemical constraints imposed by the environment of a subcellular compartment. In particular, the signatures predicted for IM has a high percentage of non-polar amino acids (60%) and no charged (0%) amino acids. This can be explained in terms of the physico-chemical properties of the lipid bilayer, in which non-polar amino acids are favored in the transmembrane domains of IM proteins(Ulmschneider, et al., 2005). In contrast, charged amino acids are disfavored due to the penalty incurred in energy terms during the assembly of IM proteins(Hessa, et al., 2005). CP and EC classes are found to contain a high percentage

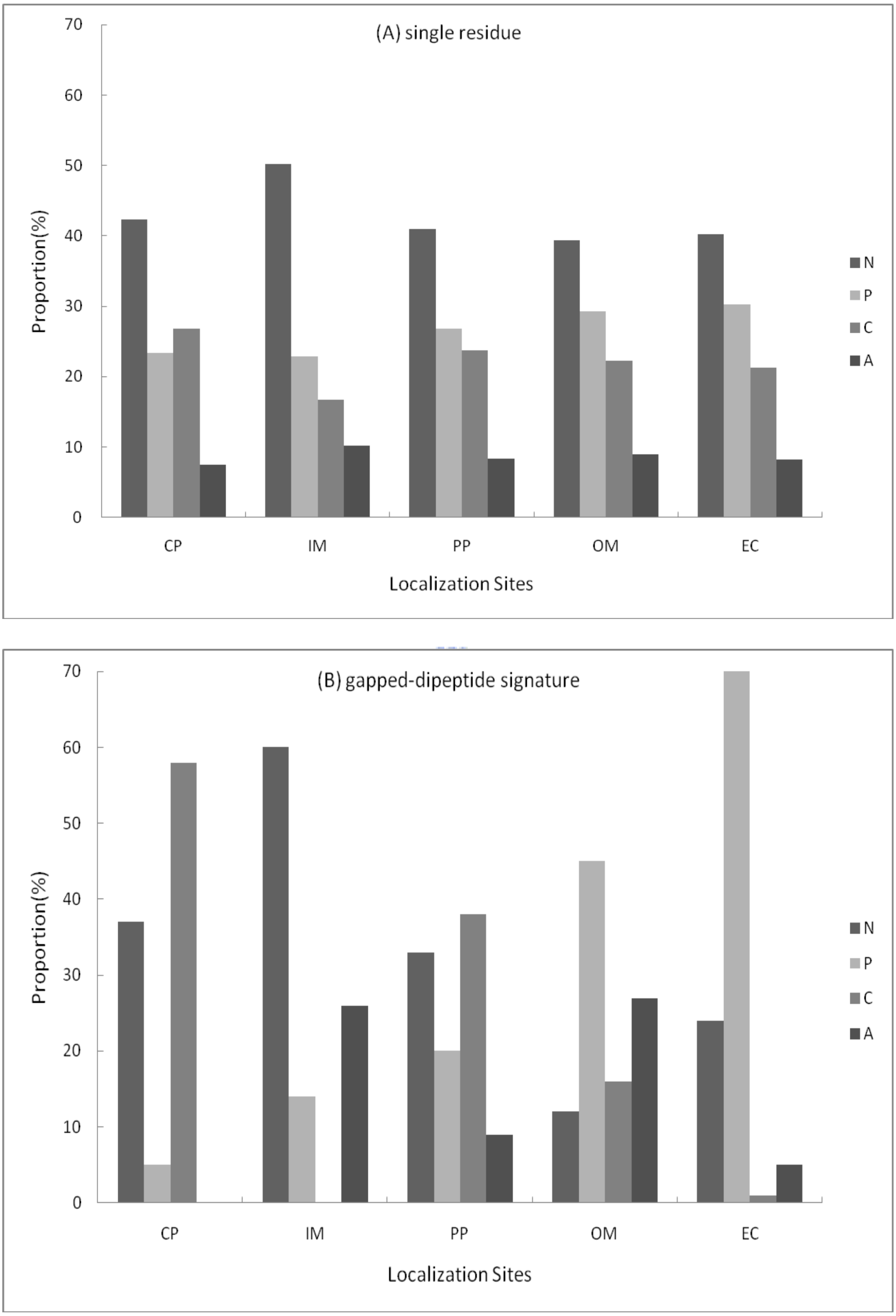


Figure 3.12: The amino acid compositions of single residues (A) and predicted gapped-dipeptide signatures (B) for each protein class distinguished by the localization site. Localization sites: CP, IM, PP, OM, and EC. Amino acid groups: N (non-polar: AIGLMV), P (polar: CNPQST), C (charged: DEHKR), and A (aromatic: FYW)

of charged and polar amino acids, respectively. The role of charged amino acids in the cytoplasm is probably related to pH homeostasis in which they act as buffers, whereas secreted proteins in the EC classes may require more polar amino acids for promoting interactions in the solvent environment(Booth, 1985).

Although gapped-dipeptide signatures are found, PSLDoc performs training and testing procedures solely based on the topics of the PLSA model. In addition, Hofmann(Hofmann, 2001) also noted that PLSA can capture the semantic meaning of words, in our case, the gapped-dipeptides. This part will be discussed in the following section.

3.3.5 Capability to solve the polysemy of Gapped-dipeptides

In document classification, a word with two different meanings is called polyseme (e.g., ‘bank’ means (i) an organization that provides various financial services or (ii) the side of a river). Hofmann mentioned PLSA could deal with polysemy and gave an example about the word “segment” ((i) an image region or (ii) a phonetic segment)(Hofmann, 1999; Hofmann, 2001). Such a word *w* would have a high probability in two different topics. The hidden topic variable, $P(w|z)$, associated with each word occurrence in a particular document is used to determine which particular topic *w* is assigned to, depending on the context of the document. Sivic *et al.*(Sivic, et al., 2005) applied PLSA to images and discussed the polysemy on images. We discuss the polysemy effect on gapped-dipeptides.

Table 3.8: “A6A” is among the top five frequent dipeptides of Topic 73 and Topic 6, where gapped-dipeptides are arranged to the decreasing order of $P(w|z)$.

Topic 73	Topic 6
A6A	A10A
A13A	A11A
A7A	A6A
A10A	A12A
A11A	A3A

A gapped-dipeptide may prefer two localization sites, e.g., “A6A” prefer CP and PP in Table 3.7. It is sometimes difficult to determine the localization site of a protein based on the weight of a polysemous gapped-dipeptide. PLSA can be used to remedy the polysemy effect of a gapped-dipeptide by associating the gapped-dipeptide with different topics. For example, “A6A” is among the top five frequent dipeptides of Topic 73 in CP and Topic 6 in PP that their probabilities $P(w|z)$ are sorted in a decreasing order as shown in Table 3.8.

For example, two proteins from PS1444 data set, chemotaxis protein cheZ and Endoglucanase B[6], contain subsequences of the polysemous gapped-dipeptide “A6A.” They are in different classes, CP class and PP class, respectively, and some of their relevant information is listed in Table 3.9. Using the original vector space, the two proteins have $P\{w$ = “A6A”, $d_{44}\}$ = 0.7001 and $P\{w$ = “A6A”, $d_{680}\}$ = 0.651, which differ slightly, and thus it is difficult to distinguish them. However, using the posterior probabilities of Topic 73 and Topic 6, given the different occurrences of “A6A” based on the PLSA reduced vector space can distinguish the two proteins and determine their classes. That is, since $(P\{z_{73}|\ w$ = “A6A”, $d_{44}\}$, $P\{z_6|\ w$ = “A6A”, $d_{44}\})$ = (0.0794, 0.0) and $(P\{z_{73}|\ w$ = “A6A”, $d_{680}\}$, $P\{z_6|\ w$ = “A6A”, $d_{680}\})$ = (0.0, 0.0596) and Topics 73 and 6 are associated with different classes, the proteins d_{44} and d_{680} can be distin-

[6] chemotaxis protein cheZ and Endoglucanase B are 44-th and 680-th proteins in PS1444, respectively. We use d_{44} and d_{680} to denote them for ease.

guished to be in CP and PP classes. This example demonstrates PLSA’s capability to remedy the polysemy effect of gapped-dipeptides.

Table 3.9: Examples of two proteins in PS1444 having the gapped-dipeptide “A6A”.

Protein Name	P{w_j= “A6A”, d_i }	(P{z_{73}|d_i, w_j = “A6A”}, P{z_6| d_i, w_j = “A6A”})	“A6A”	Localization Site

116294	0.7001	(0.0794, 0.0)	AIAEAAEA(40*)
			ASQPHQDA(75)
				CP

121816	0.651	(0.0, 0.0596)	APGDPGSA(362)
			AQWGVSNA(409)
			AQYGGFLA(420)
				PP

*The number in brackets denotes the starting position of the gapped-dipeptide “A6A.”

3.4 Conclusion

In this chapter, we present a new PSL prediction method, PSLDoc, based on gapped-dipeptides and PLSA and demonstrate that it is suitable for proteins of a wide range of sequence homologies. PSLDoc extracts features from gapped-dipeptides of various distances, where evolutionary information from the PSSM is utilized to determine the weighting of each gapped-dipeptides such that its performance is comparable to the homology search method in the high-homology data set. These features are further reduced by PLSA and incorporated as input vectors for SVM classifiers. PSLDoc performs very well in low-homology data set with overall accuracy of 86.84%. It can also achieve very high precision by using a flexible prediction threshold. Experiments show PSLDoc performs better than some of the current methods in overall accuracy by 1.51%. Because of the generality of this method, it can be extended to other species or multiple localization sites in the future. Through analyzing the amino acid composition of gapped-dipeptide signatures, there is a relationship between the amino acid group and localization sites. For future work, we will incorporate the amino acid groups with gapped-dipeptides to design a new representation of terms for predicting protein subcellular localization.

CHAPTER 4

Prediction of RNA-binding Sites in Proteins

4.1 Methods

In this chapter, we propose a method, RNAProB, which incorporates a new smoothed position-specific scoring matrix (PSSM) encoding scheme with a support vector machine model to predict RNA-binding sites in proteins. Besides the incorporation of evolutionary information from standard PSSM profiles, the proposed smoothed PSSM encoding scheme also considers the correlation and dependency from the neighboring residues for each amino acid in a protein.

4.1.1 Data sets

In this study, we apply three data sets used in previous studies to compare the performance of our method and other systems. Table 4.1 shows a summary of these data sets, which are detailed as follows and available in the supplementary material [see Appendix 2.1, 2.2, and 2.3].

1. RBP86

The RBP86 data set consists of 86 protein chains extracted from RNA-protein complexes with X-ray crystallography resolution better than 3Å in PDB. Sequence redundancy in the data set is removed so that no protein pair has a sequence identity greater than 70%. In the RNA-protein complexes, a residue is regarded as interacting with RNA if the distance between an RNA molecule and the residue in the protein is less than 6Å. The resultant data set contains 4,568 RNA interacting residues and 15,503 non-interacting residues. The RBP86 data set has been used in Terribilini *et al.* (Terribilini, et al., 2006) and Kumar *et al.* (Kumar, et al., 2008). In Kumar *et al.*, it is also referred to as the “main” data set.

2. RBP109

The RBP109 data set contains 109 protein sequences obtained from 56 RNA-protein complexes with X-ray crystallography resolution better than 3.5Å in PDB. For any two protein chains, the sequence identity is no more than 30%. The numbers of interacting and non-interacting residues are 3,581 and 21,526, respectively. The RBP109 data set is downloaded from RNABindR web server (http://bindr.gdcb.iastate.edu/RNABindR/) (Terribilini, et al., 2007). In Terribilini *et al.* (Terribilini, et al., 2006), it is named as the “RB109” data set.

3. RBP107

Derived from 61 RNA-protein complexes in PDB, the RBP107 data set is comprised of 107 protein chains with X-ray crystallography resolution better than 3.5Å and sequence identity no more than 25%. Based on a cutoff distance of 3.5Å, the RBP107 data set contains 2,555 interacting residues and 19,496 non-interacting ones. Wang and Brown (Wang and Brown, 2006) applied this data set to construct and evaluated their approach. In Kumar *et al.* (Kumar, et al., 2008), it is named as the “alternate” set.

Table 4.1: Summary of three benchmark data sets.

Data set	RBP86	RBP109	RBP107
Number of protein chains	86	109	107
X-ray crystallography resolution	>3Å	>3.5Å	>3.5Å
Sequence identity	≤70%	≤30%	≤25%
Number of interacting residues	4,568	3,581	2,555
Number of non-interacting residues	15,503	21,526	19,496
Non-interacting/interacting residues	3.39	6.01	7.63
Total number of residues	20,071	25,107	22,051

4.1.2 Support vector machines (SVM)

SVM is a machine learning approach proposed by Vapnik (Vapnik, 1995) based on structural risk minimization principle of statistics learning theory. It can be used to deal with classification or regression. Distinguishing RNA binding residues form non-binding residues in a protein could be regarded as a binary classification problem. For a set of given input data vectors x_i ($\mathrm{x_i} \in \mathbb{R}^d$, i = 1, 2, …, n) with labels y_i ($\mathrm{y_i} \in \{+1, -1\}$, i = 1, 2, …, n; where “+1” represents a positive instance and “-1” denotes a negative instance), the mission in the training procedure is to optimize the following equation that maps input vectors into a higher dimensional feature space (i.e., Hilbert space), and seeks a separation hyperplane with a maximum margin to divide positive instances from negative ones. The calculation of SVM is defined in Equation (4.1).

$$\mathrm{Min}_{\mathrm{w,b,}\xi_i}\left(\frac{1}{2}\mathrm{w^T w} + \mathrm{C}\sum_{i=1}^{n}\xi_i\right) \tag{4.1}$$

$$\text{subject to } \mathrm{y_i}(\mathrm{w^T}\Phi(\mathrm{x_i}) + \mathrm{b}) \geq 1 - \xi_i, \xi_i \geq 0, \mathrm{i} = 1,2, \dots, \mathrm{n},$$

where $\mathrm{w} \in \mathbb{R}^d$ is a weight vector, b is a bias (constant), and Φ is a mapping function. For more flexible classification, SVM allows instance i positions at the wrong side of hyperplane with slack variable ξ_i and cost parameter C. In SVM, a kernel function $\mathrm{K(x_i,x_j)}$, such as linear, polynomial, radial basis function (RBF), and sigmoid function, is used to present $\Phi(\mathrm{x_i}) \cdot \Phi(\mathrm{x_j})$ where x_i and x_j are two data vectors. In this study, we use RBF as the kernel function in the SVM. The formulation of RBF is defined in Equation (4.2), where γ is a training parameter.

$$\mathrm{K}(\mathrm{x_i}\ ,\mathrm{x_j}) = \exp\left(-\gamma\|\mathrm{x_i} - \mathrm{x_j}\|^2\right) \tag{4.2}$$

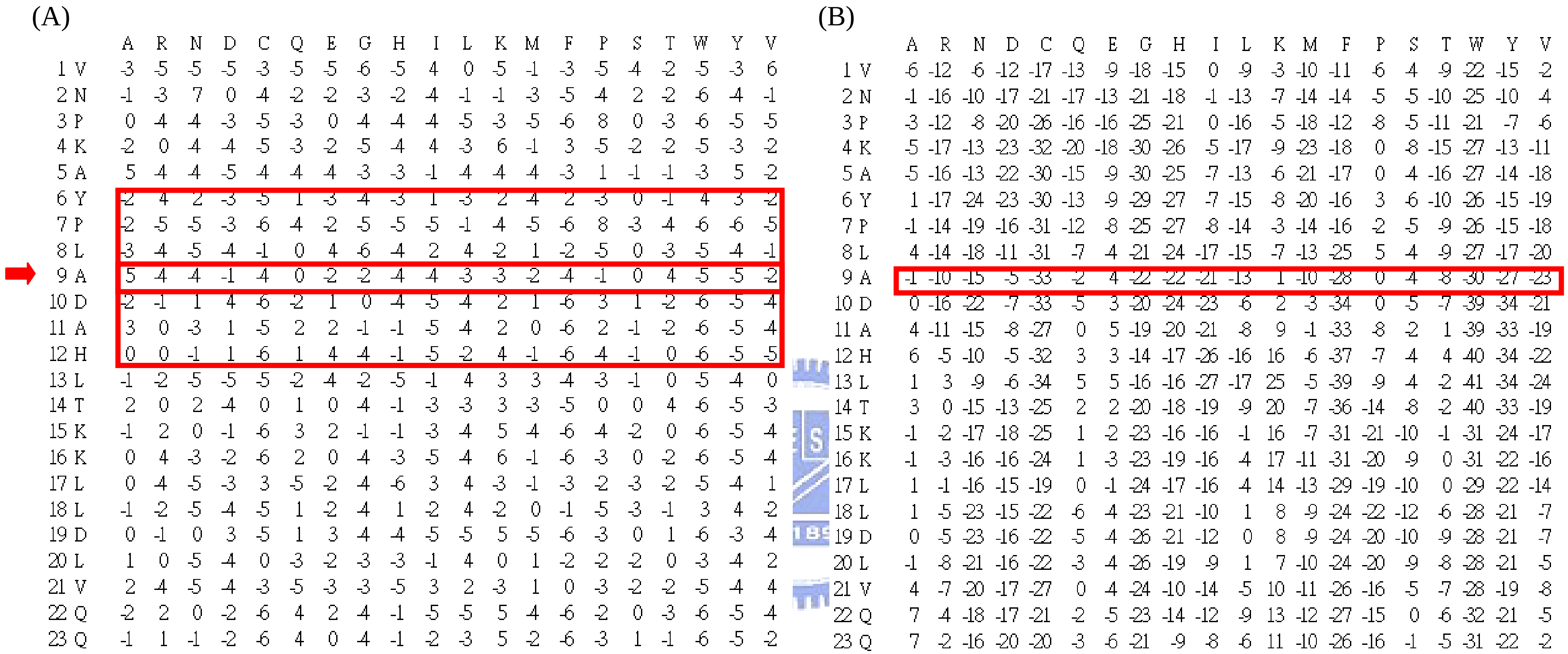
Developed by Lin *et al.* (Lin and Chang, 2001), LIBSVM is a powerful and well-known SVM package used by many researchers. We apply LIBSVM to implement our classifiers for prediction of RNA-binding sites in proteins.

4.1.3 Feature extraction and representation

Evolutionary information has been shown to be effective for RNA-binding site prediction (Kumar, et al., 2008). For this reason, we use PSI-BLAST (Altschul, et al., 1997) to search against NCBI non-redundant (nr) database and generate a PSSM based on BLOSUM62 substitution matrix (Henikoff and Henikoff, 1992) for each protein with e-value as 0.001 and iteration number as 3. A PSSM is comprised of L vectors (L denotes the length of the protein), in which contain the log-likelihoods for different amino acids in a position. Next, we illustrate two different encoding schemes to represent the PSSM.

1. Standard PSSM encoding scheme

Standard PSSM has been used for RNA-binding site prediction by Kumar *et al.* (Kumar, et al., 2008). For a PSSM profile, the feature representation of a residue α_i at position i in a protein sequence is presented by an evolutionary information vector V_i comprised of log-likelihoods for 20 different amino acids. Considering the surrounding residues of α_i, we apply a sliding window of size w to incorporate the evolutionary information from upstream and downstream neighbors. The feature vector of a residue α_i is represented by ($V_{i-(w-1)/2}$, …, V_i, …,$V_{i+(w-1)/2}$). For the N-terminal and C-terminal of a protein, (w-1)/2 ZERO vectors, consisting of 20 zero elements, are appended to the hand or tail of a PSSM profile. The feature values in each vector are normalized to a range between -1 and 1. In our study, we apply different sliding window sizes from 3 to 41 with a step as 2 (i.e., w = 3, 5, …, 41). Figure 4.1 (A) shows an example of



(A)

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
1 V	-3	-5	-5	-5	-3	-5	-5	-6	-5	4	0	-5	-1	-3	-5	-4	-2	-5	-3	6
2 N	-1	-3	7	0	-4	-2	-2	-3	-2	-4	-1	-1	-3	-5	-4	2	-2	-6	-4	-1
3 P	0	-4	-4	-3	-5	-3	0	-4	-4	-4	-5	-3	-5	-6	8	0	-3	-6	-5	-5
4 K	-2	0	-4	-4	-5	-3	-2	-5	-4	4	-3	6	-1	3	-5	-2	-2	-5	-3	-2
5 A	5	-4	-4	-5	-4	-4	-4	-3	-3	-1	-4	-4	-4	-3	1	-1	-1	-3	5	-2
6 Y	-2	4	2	-3	-5	1	-3	-4	-3	1	-3	2	-4	2	-3	0	-1	4	3	-2
7 P	-2	-5	-5	-3	-6	-4	-2	-5	-5	-5	-1	-4	-5	-6	8	-3	-4	-6	-6	-5
8 L	-3	-4	-5	-4	-1	0	4	-6	-4	2	4	-2	1	-2	-5	0	-3	-5	-4	-1
9 A	5	-4	-4	-1	-4	0	-2	-2	-4	-4	-3	-3	-2	-4	-1	0	4	-5	-5	-2
10 D	-2	-1	1	4	-6	-2	1	0	-4	-5	-4	2	1	-6	3	1	-2	-6	-5	-4
11 A	3	0	-3	1	-5	2	2	-1	-1	-5	-4	2	0	-6	2	-1	-2	-6	-5	-4
12 H	0	0	-1	1	-6	1	4	-4	-1	-5	-2	4	-1	-6	-4	-1	0	-6	-5	-5
13 L	-1	-2	-5	-5	-5	-2	-4	-2	-5	-1	4	3	3	-4	-3	-1	0	-5	-4	0
14 T	2	0	2	-4	0	1	0	-4	-1	-3	-3	3	-3	-5	0	0	4	-6	-5	-3
15 K	-1	2	0	-1	-6	3	2	-1	-1	-3	-4	5	-4	-6	-4	-2	0	-6	-5	-4
16 K	0	4	-3	-2	-6	2	0	-4	-3	-5	-4	6	-1	-6	-3	0	-2	-6	-5	-4
17 L	0	-4	-5	-3	3	-5	-2	-4	-6	3	4	-3	-1	-3	-2	-3	-2	-5	-4	1
18 L	-1	-2	-5	-4	-5	1	-2	-4	1	-2	4	-2	0	-1	-5	-3	-1	3	4	-2
19 D	0	-1	0	3	-5	1	3	-4	-4	-5	-5	5	-5	-6	-3	0	1	-6	-3	-4
20 L	1	0	-5	-4	0	-3	-2	-3	-3	-1	4	0	1	-2	-2	-2	0	-3	-4	2
21 V	2	-4	-5	-4	-3	-5	-3	-3	-5	3	2	-3	1	0	-3	-2	-2	-5	-4	4
22 Q	-2	2	0	-2	-6	4	2	-4	-1	-5	-5	5	-4	-6	-2	0	-3	-6	-5	-4
23 Q	-1	1	-1	-2	-6	4	0	-4	-1	-2	-3	5	-2	-6	-3	1	-1	-6	-5	-2

(B)

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
1 V	-6	-12	-6	-12	-17	-13	-9	-18	-15	0	-9	-3	-10	-11	-6	-4	-9	-22	-15	-2
2 N	-1	-16	-10	-17	-21	-17	-13	-21	-18	-1	-13	-7	-14	-14	-5	-5	-10	-25	-10	-4
3 P	-3	-12	-8	-20	-26	-16	-16	-25	-21	0	-16	-5	-18	-12	-8	-5	-11	-21	-7	-6
4 K	-5	-17	-13	-23	-32	-20	-18	-30	-26	-5	-17	-9	-23	-18	0	-8	-15	-27	-13	-11
5 A	-5	-16	-13	-22	-30	-15	-9	-30	-25	-7	-13	-6	-21	-17	0	-4	-16	-27	-14	-18
6 Y	1	-17	-24	-23	-30	-13	-9	-29	-27	-7	-15	-8	-20	-16	3	-6	-10	-26	-15	-19
7 P	-1	-14	-19	-16	-31	-12	-8	-25	-27	-8	-14	-3	-14	-16	-2	-5	-9	-26	-15	-18
8 L	4	-14	-18	-11	-31	-7	-4	-21	-24	-17	-15	-7	-13	-25	5	-4	-9	-27	-17	-20
9 A	-1	-10	-15	-5	-33	-2	4	-22	-22	-21	-13	1	-10	-28	0	-4	-8	-30	-27	-23
10 D	0	-16	-22	-7	-33	-5	3	-20	-24	-23	-6	2	-3	-34	0	-5	-7	-39	-34	-21
11 A	4	-11	-15	-8	-27	0	5	-19	-20	-21	-8	9	-1	-33	-8	-2	1	-39	-33	-19
12 H	6	-5	-10	-5	-32	3	3	-14	-17	-26	-16	16	-6	-37	-7	-4	4	-40	-34	-22
13 L	1	3	-9	-6	-34	5	5	-16	-16	-27	-17	25	-5	-39	-9	-4	-2	-41	-34	-24
14 T	3	0	-15	-13	-25	2	2	-20	-18	-19	-9	20	-7	-36	-14	-8	-2	-40	-33	-19
15 K	-1	-2	-17	-18	-25	1	-2	-23	-16	-16	-1	16	-7	-31	-21	-10	-1	-31	-24	-17
16 K	-1	-3	-16	-16	-24	1	-3	-23	-19	-16	-4	17	-11	-31	-20	-9	0	-31	-22	-16
17 L	1	-1	-16	-15	-19	0	-1	-24	-17	-16	-4	14	-13	-29	-19	-10	0	-29	-22	-14
18 L	1	-5	-23	-15	-22	-6	-4	-23	-21	-10	1	8	-9	-24	-22	-12	-6	-28	-21	-7
19 D	0	-5	-23	-16	-22	-5	-4	-26	-21	-12	0	8	-9	-24	-20	-10	-9	-28	-21	-7
20 L	-1	-8	-21	-16	-22	-3	-4	-26	-19	-9	1	7	-10	-24	-20	-9	-8	-28	-21	-5
21 V	4	-7	-20	-17	-27	0	-4	-24	-10	-14	-5	10	-11	-26	-16	-5	-7	-28	-19	-8
22 Q	7	-4	-18	-17	-21	-2	-5	-23	-14	-12	-9	13	-12	-27	-15	0	-6	-32	-21	-5
23 Q	7	-2	-16	-20	-20	-3	-6	-21	-9	-8	-6	11	-10	-26	-16	-1	-5	-31	-22	-2

Figure 4.1: Examples of (A) standard PSSM and (B) smoothed PSSM generated by PSI-BLAST (e-value = 0.001, iteration number = 3).

standard PSSM of a protein with e-value as 0.001 and iteration number as 3 in PSI-BLAST.

2. Smoothed PSSM encoding scheme

In addition to the consideration of neighbors of a residue α_i, we propose a new encoding scheme to incorporate the dependency of surrounding residues. In a standard PSSM profile, the log-likelihood at each position is calculated based on an assumption that each position is independent from the others. However, Terribilini *et al.* (Terribilini, et al., 2006) observed that RNA binding residues tend to occur in clusters. Their analysis revealed that 95% of interacting residues in the RBP109 data set have at least one additional interacting residue among the four amino acids on either side, and 49% of those have at least four. Inspired by the consideration of adjacent pixels used in the spatial domain method from the research field of image processing (Gonzalez and Woods, 2002), we present a new encoding scheme to model the dependency or correlation among surrounding neighbors of a central residue. Similar to the feature representation in standard PSSM encoding, we use a sliding window of size *w* to incorporate the evolutionary information from upstream and downstream residues. In the construction of a smoothed PSSM, each row vector of a residue α_i is represented and smoothed by the summation of *ws* surrounding row vectors ($V_{\text{smoothed\_i}} = V_{\text{i-(ws-1)/2}} + \ldots + V_{\text{i}} + \ldots + V_{\text{i+(ws-1)/2}}$). For the N-terminal and C-terminal of a protein, (w-1)/2 ZERO vectors, are appended to the hand or tail of a smoothed PSSM profile. Using the smoothed PSSM encoding scheme, the feature vector of a residue α_i is represented by ($V_{\text{smoothed\_i-(w-1)/2}}$, …, $V_{\text{smoothed\_i}}$, …, $V_{\text{smoothed\_i+(w-1)/2}}$). The feature values in each vector are normalized to a range between -1 and 1. Here, we apply different smoothing window sizes from 3 to 11 with a step as 2 (i.e., *ws* = 3, 5, …, 11). Figure 4.1 (B) illustrates an example of a smoothed PSSM profile. At position 9, the corresponding value of amino acid ‘A’ represented by a smoothed PSSM encoding is the sum of [(-2)+(-2)+ (-3)+5+(-2)+3+0].

4.1.4 Window size selection and parameter optimization

In order to optimize the performance of RNAProB, we have to determine the best combination of several parameters, including the sliding window size *w*, cost parameter *C* and kernel parameter γ in the SVM classifier, the smoothing window size *ws*, and the weight parameters w_1 and w_{-1} in SVM. Table 4.2 shows the workflow of window size selection and parameter optimization. In our study, the best parameters are optimized with respect to overall accuracy. First, we test the performance of different sliding window sizes *w* from 3, 5, 7, …, 41 in standard PSSM encoding scheme using default *C* and γ parameters in SVM, and initial weight parameter w_1 as 1 and w_{-1} as the ratio of the number of non-interacting residues to that of interacting residues in a data set. As shown in Table 4.1, the ratios of the numbers of non-interacting residues to those of interacting residues in the RBP86, RBP109, and RBP107 data sets are 1:3.39, 1:6.01, and 1:7.63, respectively. Second, based on the optimized sliding window size *w* selected from the first step, the best combination of cost parameter *C* and kernel parameter γ is determined with initial weight parameters. The $\log_2$C and $\log_2\gamma$ ranged from -3 to 12 and -3 to -15, respectively. Third, the prediction performance of different smoothing window sizes *ws* ranged from 3 to 11 with a step 2 is evaluated using initial weight parameters and previously selected parameters (i.e., *w*, *C*, and γ). Fourth, due to data set imbalance, the weight parameters w_1 and w_{-1} are tuned with optimized *w*, *C*, γ, and *ws*. After these steps, the optimal parameters, including sliding window size *w*, cost parameter *C*, kernel parameter γ, smoothing window size *ws*, and weight parameters w_1 and w_{-1}, are determined.

Table 4.2: The workflow of window size selection and parameter optimization.

	Sliding window size (*w*)	*C* and *γ*	Smoothing window size (*ws*)	Weight parameter (w_1 and w_{-1})
Step 1	$3 \leq w \leq 41$ (step = 2)	Default	-	Default ratio
Step 2	Optimized *w* from step 1	$-3 \leq \log_2 C \leq 12$ (step = 1) $-3 \leq \log_2 \gamma \leq -15$ (step = -1)	-	Default ratio
Step 3	Optimized *w* from step 1	Optimized *C* and *γ* from step 2	$3 \leq ws \leq 11$ (step = 2)	Default ratio
Step 4	Optimized *w* from step 1	Optimized *C* and *γ* from step 2	Optimized *ws* from step 3	$1 \leq w_1 \leq 8$[#] (step = 1), $w_{-1} = 1$
Final	Optimized *w* from step 1	Optimized *C* and *γ* from step 2	Optimized *ws* from step 3	Optimized w_1 and w_{-1} from step4

[#] In the RBP107 data set, we test w_1 ranged from 1 to 10 (step = 1).

4.1.5 System architecture of RNAProB

The system architecture of RNAProB is shown in Figure 4.2. Given a protein sequence, RNAProB performs the following steps:

1. Apply PSI-BLAST to generate a standard PSSM of the protein.

2. Generate a smoothed PSSM of the protein using an optimized smoothing window size.

3. Construct a feature vector for each residue in the protein sequence by an optimized sliding window size, and normalize all feature values in the vector into a range of -1 and 1.

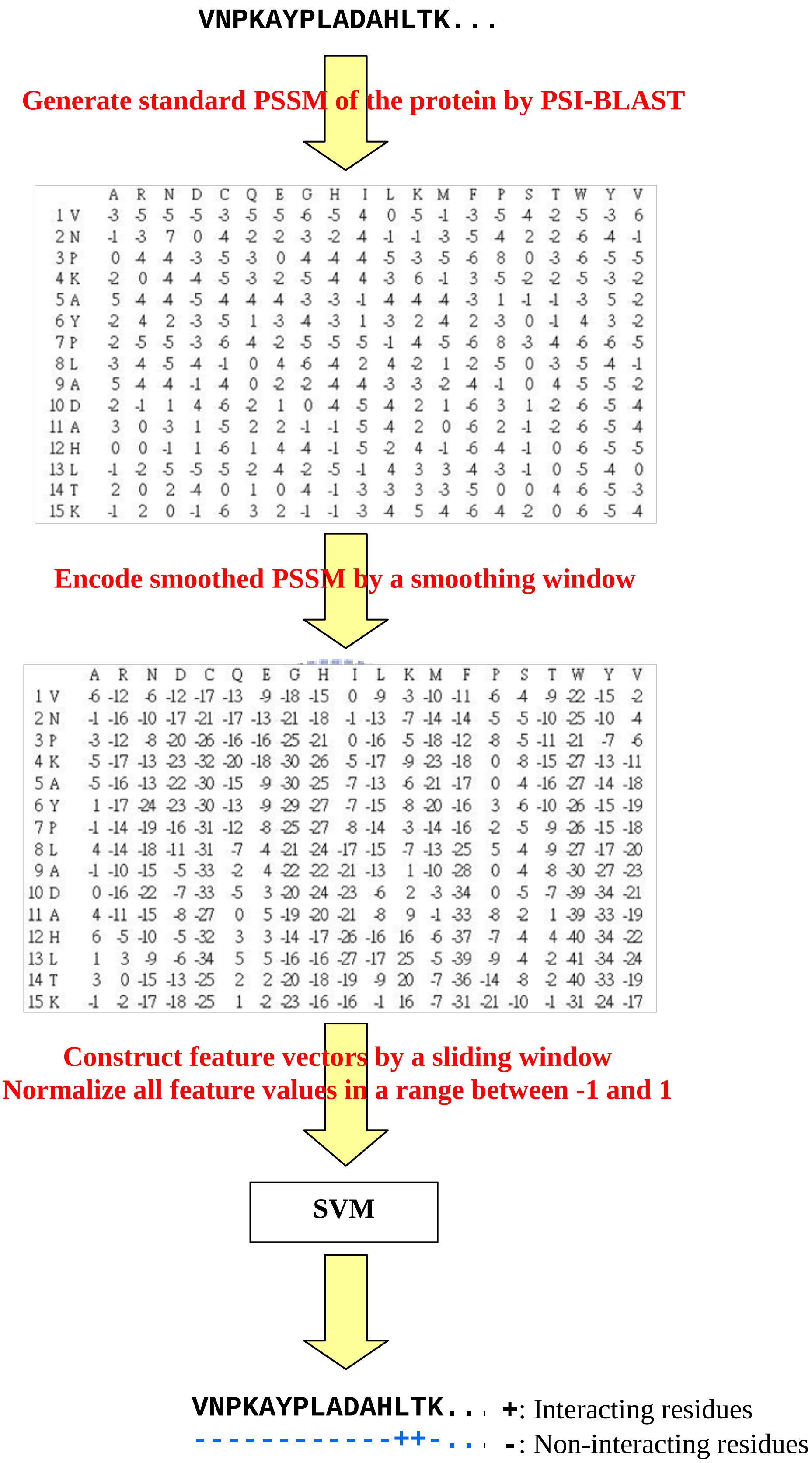
4. Use a trained SVM classifier with optimized parameters (C, γ, w_{1}, w_{-1}) to predict the interacting and non-interacting residues in the protein.

After the above steps, RNAProB outputs the corresponding interacting or non-interacting state of each residue in the protein.

4.1.6 Training and testing

The performance of RNAProB is assessed by *n*-fold cross-validation and three-way data split. To compare with other approaches, we use five-fold cross-validation to evaluate the performance of RNAProB. However, to prevent data-overfitting, a three-way data split procedure is applied to assess our predictor. The performance of RNAProB is evaluated as follows.

1. *n*-fold cross-validation: A data set is randomly divided into five distinct non-overlapping sets of positive and negative instances (i.e., n = 5), four of which are used to train the predictor and the accuracy of the predictor is evaluated on the remaining set. This procedure is repeated five times.



VNPKAYPLADAHLTK...

Generate standard PSSM of the protein by PSI-BLAST

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
1 V	-3	-5	-5	-5	-3	-5	-5	-6	-5	4	0	-5	-1	-3	-5	-4	-2	-5	-3	6
2 N	-1	-3	7	0	-4	-2	-2	-3	-2	-4	-1	-1	-3	-5	-4	2	-2	-6	-4	-1
3 P	0	-4	-4	-3	-5	-3	0	-4	-4	-4	-5	-3	-5	-6	8	0	-3	-6	-5	-5
4 K	-2	0	-4	-4	-5	-3	-2	-5	-4	4	-3	6	-1	3	-5	-2	-2	-5	-3	-2
5 A	5	-4	-4	-5	-4	-4	-4	-3	-3	-1	-4	-4	-4	-3	1	-1	-1	-3	5	-2
6 Y	-2	4	2	-3	-5	1	-3	-4	-3	1	-3	2	-4	2	-3	0	-1	4	3	-2
7 P	-2	-5	-5	-3	-6	-4	-2	-5	-5	-5	-1	-4	-5	-6	8	-3	-4	-6	-6	-5
8 L	-3	-4	-5	-4	-1	0	4	-6	-4	2	4	-2	1	-2	-5	0	-3	-5	-4	-1
9 A	5	-4	-4	-1	-4	0	-2	-2	-4	-4	-3	-3	-2	-4	-1	0	4	-5	-5	-2
10 D	-2	-1	1	4	-6	-2	1	0	-4	-5	-4	2	1	-6	3	1	-2	-6	-5	-4
11 A	3	0	-3	1	-5	2	2	-1	-1	-5	-4	2	0	-6	2	-1	-2	-6	-5	-4
12 H	0	0	-1	1	-6	1	4	-4	-1	-5	-2	4	-1	-6	-4	-1	0	-6	-5	-5
13 L	-1	-2	-5	-5	-5	-2	-4	-2	-5	-1	4	3	3	-4	-3	-1	0	-5	-4	0
14 T	2	0	2	-4	0	1	0	-4	-1	-3	-3	3	-3	-5	0	0	4	-6	-5	-3
15 K	-1	2	0	-1	-6	3	2	-1	-1	-3	-4	5	-4	-6	-4	-2	0	-6	-5	-4

Encode smoothed PSSM by a smoothing window

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
1 V	-6	-12	-6	-12	-17	-13	-9	-18	-15	0	-9	-3	-10	-11	-6	-4	-9	-22	-15	-2
2 N	-1	-16	-10	-17	-21	-17	-13	-21	-18	-1	-13	-7	-14	-14	-5	-5	-10	-25	-10	-4
3 P	-3	-12	-8	-20	-26	-16	-16	-25	-21	0	-16	-5	-18	-12	-8	-5	-11	-21	-7	-6
4 K	-5	-17	-13	-23	-32	-20	-18	-30	-26	-5	-17	-9	-23	-18	0	-8	-15	-27	-13	-11
5 A	-5	-16	-13	-22	-30	-15	-9	-30	-25	-7	-13	-6	-21	-17	0	-4	-16	-27	-14	-18
6 Y	1	-17	-24	-23	-30	-13	-9	-29	-27	-7	-15	-8	-20	-16	3	-6	-10	-26	-15	-19
7 P	-1	-14	-19	-16	-31	-12	-8	-25	-27	-8	-14	-3	-14	-16	-2	-5	-9	-26	-15	-18
8 L	4	-14	-18	-11	-31	-7	-4	-21	-24	-17	-15	-7	-13	-25	5	-4	-9	-27	-17	-20
9 A	-1	-10	-15	-5	-33	-2	4	-22	-22	-21	-13	1	-10	-28	0	-4	-8	-30	-27	-23
10 D	0	-16	-22	-7	-33	-5	3	-20	-24	-23	-6	2	-3	-34	0	-5	-7	-39	-34	-21
11 A	4	-11	-15	-8	-27	0	5	-19	-20	-21	-8	9	-1	-33	-8	-2	1	-39	-33	-19
12 H	6	-5	-10	-5	-32	3	3	-14	-17	-26	-16	16	-6	-37	-7	-4	4	-40	-34	-22
13 L	1	3	-9	-6	-34	5	5	-16	-16	-27	-17	25	-5	-39	-9	-4	-2	-41	-34	-24
14 T	3	0	-15	-13	-25	2	2	-20	-18	-19	-9	20	-7	-36	-14	-8	-2	-40	-33	-19
15 K	-1	-2	-17	-18	-25	1	-2	-23	-16	-16	-1	16	-7	-31	-21	-10	-1	-31	-24	-17

Construct feature vectors by a sliding window
Normalize all feature values in a range between -1 and 1

SVM

Figure 4.2: System architecture of RNAProB.

2. Three-way data split: To avoid overfitting, we use a more stringent three-way data split procedure (Ritchie, et al., 2003; Su, et al., 2007) to evaluate the performance of RNAProB. A data set is randomly partitioned into three non-overlapping sets: a training set for classifier learning, a validation set for parameter selection, and a test set for performance evaluation. In this paper, we divide a data set into five distinct sets, three for training, one for validation, and one for testing. The procedure is also iterated 5 times.

4.1.7 Performance evaluation measures

For comparison with other approaches, we follow the measures used in previous work (Kumar, et al., 2008; Wang and Brown, 2006; Wang and Brown, 2006), including specificity (Spec.), sensitivity (Sens.), *MCC* (Matthews, 1975), and overall accuracy (Acc). Specificity and sensitivity measure how well the binary classifier recognizes negative and positive cases, respectively. A specificity of 100% and a sensitivity of 100% imply that the classifier identifies all non-interacting residues as non-interacting and all interacting residues as interacting, correspondingly. When a predictor's specificity increases, its sensitivity often decreases. On the other hand, *MCC*, which considers both under- and over-predictions, gives a complementary measure of the prediction performance, where *MCC* = 1 denotes a perfect prediction, *MCC* = 0 indicates a completely random assignment, and *MCC* = -1 means a perfectly reverse correlation. Moreover, overall accuracy presents how well the classifier distinguishes true positives and true negatives, and 100% overall accuracy denotes a perfect prediction. The definitions of specificity, sensitivity, *MCC*, and overall accuracy are defined in Equations (4.3), (4.4), (4.5), and (4.6), respectively. In the equations, *TP*, *TN*, *FP*, and *FN* denote the numbers of true positives, true negatives, false positives, and false negatives, correspondingly.

$$\text{Specificity} = \text{TN}/(\text{TN}+\text{FP}) \times 100 \tag{4.3}$$

$$\text{Sensitivity} = \text{TP}/(\text{TP}+\text{FN}) \times 100 \tag{4.4}$$

$$\text{MCC} = (\text{TP}\times\text{TN}-\text{FP}\times\text{FN})/\sqrt{(\text{TP}+\text{FP})\times(\text{TP}+\text{FN})\times(\text{TN}+\text{FP})\times(\text{TN}+\text{FN})} \tag{4.5}$$

$$\text{Acc} = (\text{TP}+\text{TN})/(\text{TP}+\text{TN}+\text{FP}+\text{FN}) \times 100 \tag{4.6}$$

In addition to the above measures, we also use the receiver operating characteristic (ROC) curve (Swets, 1988) and area under the ROC curve (AUC) (Bradley, 1997) to evaluate the performance of standard and smoothed PSSM encoding schemes. In an ROC curve plot, the X-axis represents false positive rate (i.e., 1-specificity) and Y-axis denotes true positive rate (i.e., sensitivity). We incorporate different thresholds in the SVM classifier to plot the true positive rates against false positive rates in an ROC curve. Moreover, AUC calculates the area under an ROC curve and the maximum value of AUC is 1, which denotes a perfect prediction. A random guess results in an AUC value close to 0.5.

To determine the thresholds in the SVM classifiers, we follow the criteria used in the previous work. We notice that the thresholds in other approaches are optimized with respect to different measures. For example, Kumar *et al.* (Kumar, et al., 2008) and Jeong and Miyano (Jeong and Miyano, 2006) both optimized their results in the RBP86 data set based on *MCC*. In addition, Terribilini *et al.* (Terribilini, et al., 2006) also selected the thresholds with the best *MCC* for the RBP109 data set. On the other hand, Wang and Brown (Wang and Brown, 2006) determined the best thresholds in the RBP107 data set based on the average of specificity and sensitivity. Therefore, the thresholds in RNAProB are optimized with respect to *MCC* for RBP86 and RBP109 data sets, while the threshold is determined by the average of sensitivity and specificity for the RBP107 data set.

4.2 Results

4.2.1 Effect of smoothed PSSM encoding scheme

Here we compare the performance of smoothed PSSM and standard PSSM encoding scheme in terms of *MCC*, overall accuracy, ROC curve, and AUC for the benchmark data sets. Table 4.3 shows the performance comparison of standard PSSM and smoothed PSSM using five-fold cross-validation and three-way data split. Evaluated by five-fold cross-validation, smoothed PSSM encoding scheme attains overall accuracy of 87.99%, 89.70%, and 80.44% compared to 83.39%, 87.38%, and 77.80% by standard PSSM encoding for the RBP86, RBP109, and RBP107 data sets, respectively. Moreover, smoothed PSSM encoding scheme achieves improvements of 0.06~0.178 in *MCC* compared to standard PSSM. Similarly, assessed by three-way data split, smoothed PSSM encoding also performs better than standard PSSM in terms of both overall accuracy and *MCC* in the three data sets.

Table 4.3: Performance comparison of standard PSSM and smoothed PSSM.

Data set	Smoothed PSSM		Standard PSSM	
	Acc (%)	*MCC*	Acc (%)	*MCC*
RBP86	87.99 (87.65)	0.68 (0.67)	83.39 (83.35)	0.502 (0.496)
RBP109	89.70 (89.36)	0.58 (0.56)	87.38 (86.95)	0.45 (0.43)
RBP107	80.44 (79.84)	0.42 (0.40)	77.80 (77.55)	0.36 (0.35)

[§] The performance of incorporating a three-way data split procedure is shown in the parentheses.

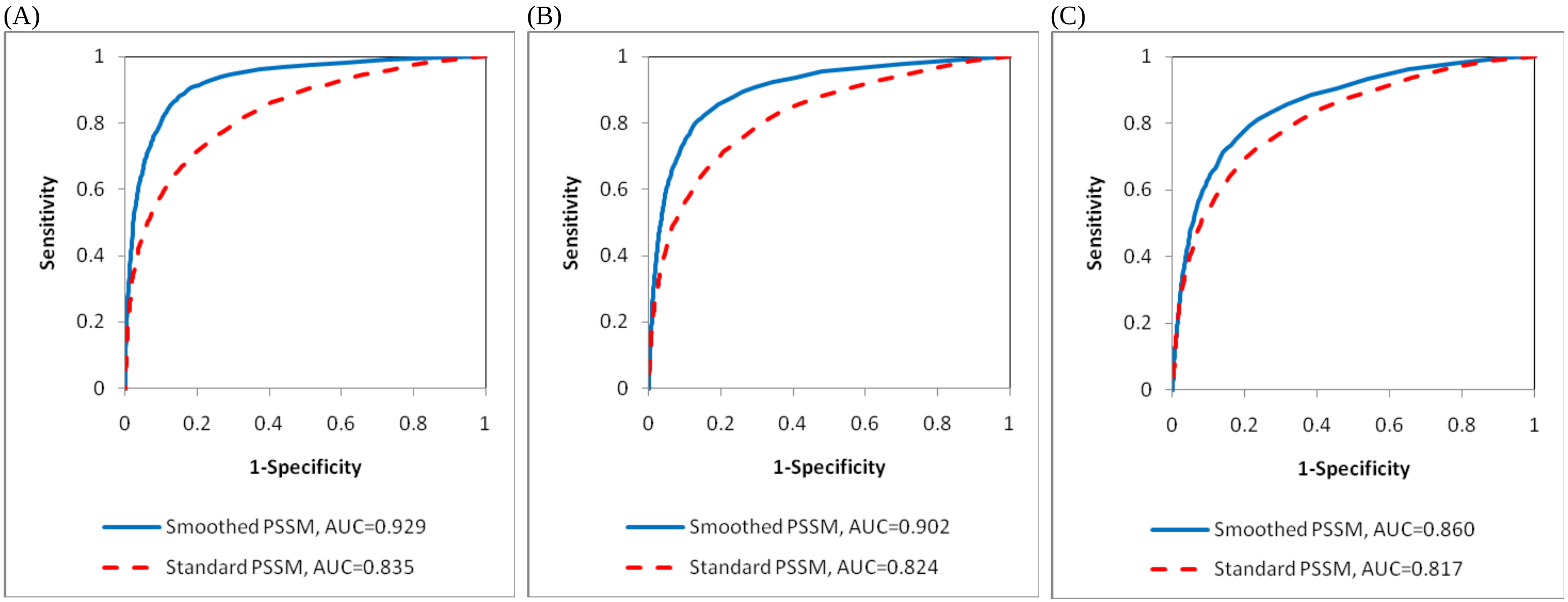


Figure 4.3: ROC curves and AUC of the (A) RBP86, (B) RBP109, and (C) RBP107 data sets.

Figure 4.3 (A), (B), and (C) illustrate the ROC curves and AUC of smoothed PSSM and standard PSSM encoding schemes for the three benchmark data sets. The solid blue line and dotted red line represent the ROC curves plotted according to the performance of smoothed PSSM and standard PSSM encoding schemes, respectively. When smoothed PSSM encoding scheme is used to represent the proteins, AUC achieve 0.929, 0.902, and 0.860 on the RBP86, RBP109, and RBP107 data sets, respectively; on the other hand, standard PSSM only attains AUC of 0.835, 0.824, and 0.817

Experiment results demonstrate that our proposed smoothed PSSM encoding scheme not only achieves good prediction performance, but also yields a significant improvement over standard PSSM encoding. Smoothed PSSM encoding scheme outperforms standard PSSM by 2.32%~4.60% in overall accuracy and 0.06~0.178 in *MCC*. The consideration of dependency among neighboring residues works well in distinguishing interacting residues from non-interacting ones; accordingly, the prediction performance of smoothed PSSM encoding scheme is substantially improved. This supports our assumption that the incorporation of the correlation between surrounding residues in PSSM profiles can significantly enhance the performance of RNA-binding site prediction.

4.2.2 RNAProB prediction performance on the benchmark data sets

For each data set, we used five-fold cross-validation and three-way data split to evaluate the prediction performance, which is detailed below and summarized in Table 4.4.

Table 4.4: Performance of five-fold cross-validation and three-way data split for the benchmark data sets.

Data set	Measurements	Spec. (%)	Sens. (%)	Acc (%)	*MCC*	Threshold
RBP86	5-fold CV	90.36	79.95	87.99	0.68	0.36
	3-way data split	90.01	79.64	87.65	0.67	0.36
RBP109	5-fold CV	93.88	64.62	89.70	0.58	0.35
	3-way data split	94.14	60.63	89.36	0.56	0.35
RBP107	5-fold CV	80.87	77.14	80.44	0.42	0.11
	3-way data split	80.65	73.62	79.84	0.40	0.12

1. Performance comparison with other approaches on the RBP86 data set

The window sizes, including the sliding window size *w* and smoothing window size *ws*, and other parameters in RNAProB are selected with respect to overall accuracy. First, Figure 4.4 (A) shows the overall accuracy of applying different sliding window sizes on the RBP86 data set. The overall accuracy evaluated by both five-fold cross-validation and three-way data split grows rapidly before it reaches 77%. However, a slow growth in the overall accuracy is observed as the size of sliding window is greater than 25. Thus, the sliding window size *w* is set as 25 for the RBP86 data set. Next the prediction performance of different smoothing window sizes based on previously determined sliding window size (i.e. *w* = 25) is illustrated in Figure 4.4 (B) and (C). In Figure 4.4 (B), although there is a very slow growth in the overall accuracy, we observe that *MCC* is improved from 0.50 to 0.67 when the size of smoothing window is increased from 1 to 7. Nevertheless, the performance improvement in *MCC* (i.e. improvement < 0.01) is not significant as the size of smoothing window is greater than 7. Similar trends in *MCC* and overall accuracy are observed in Figure 4.4 (C).

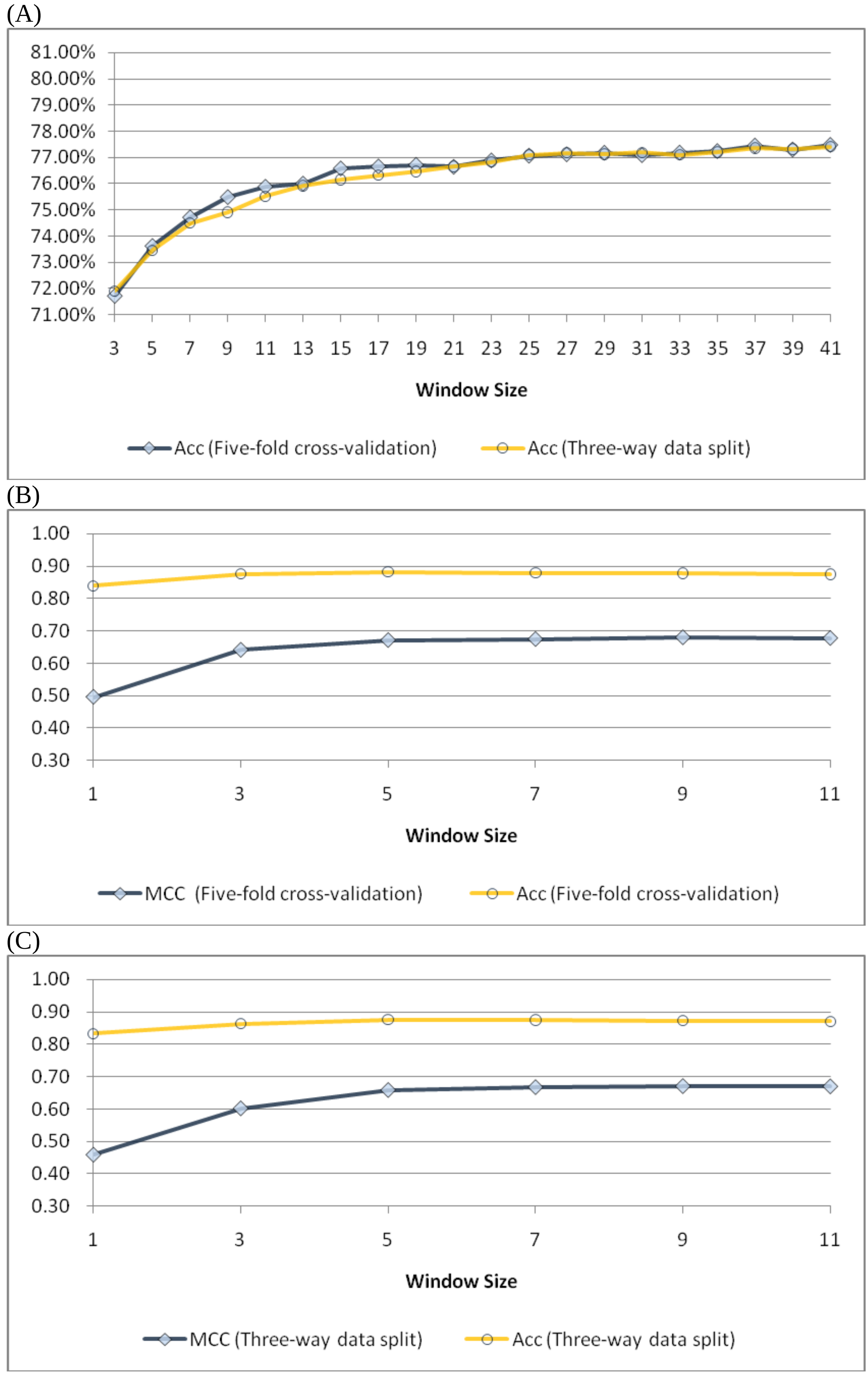


Figure 4.4: (A) Accuracy with respect to different sliding window sizes using five-fold cross-validation and three-way data split for the RBP86 data set, respectively. (B) The performance of the RBP86 data set with different smoothing window sizes by five-fold cross-validation. (C) The performance of the RBP86 data set with different smoothing window sizes by three-way data split.

Therefore, we use 7 as the smoothing window size ws in our method. As shown in Table 4.4, the performance of RNAProB evaluated by five-fold cross-validation achieves *MCC*, overall accuracy, specificity, and sensitivity of 0.68, 87.99%, 90.36%, and 79.95%, (with sliding window size $w = 25$, smoothing window size $ws = 7$, cost parameter $C = 4$, kernel function parameter $\gamma = 0.015625$, weight parameter $w_1 = 4$, $w_{-1} = 1$, and threshold value = 0.36), respectively. Besides, using a more rigorous three-way data split procedure, our method also attains *MCC*, overall accuracy, specificity, and sensitivity of 0.67, 87.65%, 90.01%, and 79.64%, (with $w = 25$, $ws = 7$, $C = 1$, $\gamma = 0.03125$, $w_1 = 4$, $w_{-1} = 1$, and threshold value = 0.36), correspondingly. The experiment results of window size selection and parameter optimization on the RBP86 data set are shown in the supplementary material [see Appendix 2.4].

The performance comparison with two other approaches developed on the same data set is shown in Table 4.5. Jeong and Miyano (Jeong and Miyano, 2006) used an ANN to incorporate evolutionary information and obtained *MCC*, overall accuracy, specificity, and sensitivity of 0.39, 80.20%, 91.04%, and 43.40%, respectively. The *MCC* of their proposed method was further improved to 0.41 based on a weighted profile approach. In addition, Kumar *et al*. developed PPRint (Kumar, et al., 2008), which incorporated PSSM profiles in an SVM model, and attained *MCC*, overall accuracy, specificity, and sensitivity of 0.45, 81.16%, 89.55%, and 53.05%, respectively. Compared to these approaches, our method not only achieves high overall accuracy but also significantly improves the sensitivity by 26.90%~36.55% using five-fold cross-validation. Moreover, RNAProB achieves 0.68 in *MCC*, compared to 0.45 by PPRint and 0.41 by Jeong and Miyano.

Table 4.5: Performance comparison of different approaches using five-fold cross-validation for the benchmark data sets.

Data set	Method	Spec. (%)	Sens. (%)	Acc (%)	*MCC*	Threshold
RBP86	Jeong 2006	91.04	43.4	80.2	0.39 (0.41)[*]	--
	PPRint	89.55	53.05	81.16	0.45	--
	RNAProB [§]	**90.36**	**79.95**	**87.99**	**0.68**	**0.36**
	RNAProB [#]	**90.01**	**79.64**	**87.65**	**0.67**	**0.36**
RBP109	RNABindR	93.00	38.00	84.80	0.35	--
	RNAProB [§]	**93.88**	**64.62**	**89.70**	**0.58**	**0.35**
	RNAProB [#]	**94.14**	**60.63**	**89.36**	**0.56**	**0.35**
RBP107	BindN-PCP[&]	69.84	66.28	69.32	0.27	--
	BindN-ALL[&]	75.70	65.78	74.25	--	--
	PPRint	75.54	70.09	75.43	0.32	--
	RNAProB [§]	**80.87**	**77.14**	**80.44**	**0.42**	**0.11**
	RNAProB [#]	**80.65**	**73.62**	**79.84**	**0.40**	**0.12**

[§] presents the performance by five-fold cross-validation.
[#] denotes the performance by a three-way data split procedure.
* indicates the performance of weighted profiles by Jeong and Miyano (Jeong and Miyano, 2006).
[&] BindN-PCP represents the results based only on physicochemical properties, while BindN-ALL shows the performance using physicochemical properties, relative solvent accessible surface area, and BLAST results.

2. Performance comparison with RNABindR on the RBP109 data set

Figure 4.5 illustrates the experiment results of different sliding and smoothing window sizes on the RBP109 data set. Similar to the RBP86 data set, the RBP109 data set exhibits a slow growth in the prediction performance when sliding window size *w* is greater than 25 or smoothing window size *ws* is larger than 7. Thus, we also select *w* as 25 and *ws* as 7 for this data set. Table 4.4 shows that RNAProB attains 0.58, 89.70%, 93.88%, and 64.62% in *MCC*, overall accuracy, specificity, and sensitivity using five-fold cross-validation (with $w = 25$, $ws = 7$, $C = 4$, $\gamma = 0.015625$, $w_1 = 4$, $w_{-1} = 1$, and threshold value = 0.35), respectively. Besides, evaluated by three-way data split, our method obtains *MCC*, overall accuracy, specificity, and sensitivity of 0.56, 89.36%, 94.14%, and 60.63% (with $w = 25$, $ws = 7$, $C = 8$, $\gamma = 0.015625$, $w_1 = 4$, $w_{-1} = 1$, and threshold value = 0.35), respectively. The prediction performance of different window sizes and parameters on the RBP109 data set is detailed in the supplementary material [see Appendix 2.5].

Table 4.5 illustrates the performance comparison with RNABindR (Terribilini, et al., 2006; Terribilini, et al., 2007), a Naïve Bayes based method developed on the same data set. Using five-fold cross-validation, RNAProB achieves 0.58, 89.70%, 93.88%, and 64.62% in *MCC*, overall accuracy, specificity, and sensitivity, respectively, compared favourably to 0.35, 84.80%, 93.00%, and 38.00% by RNABindR. Particularly, our method significantly outperforms RNABindR by 26.62% in terms of sensitivity.

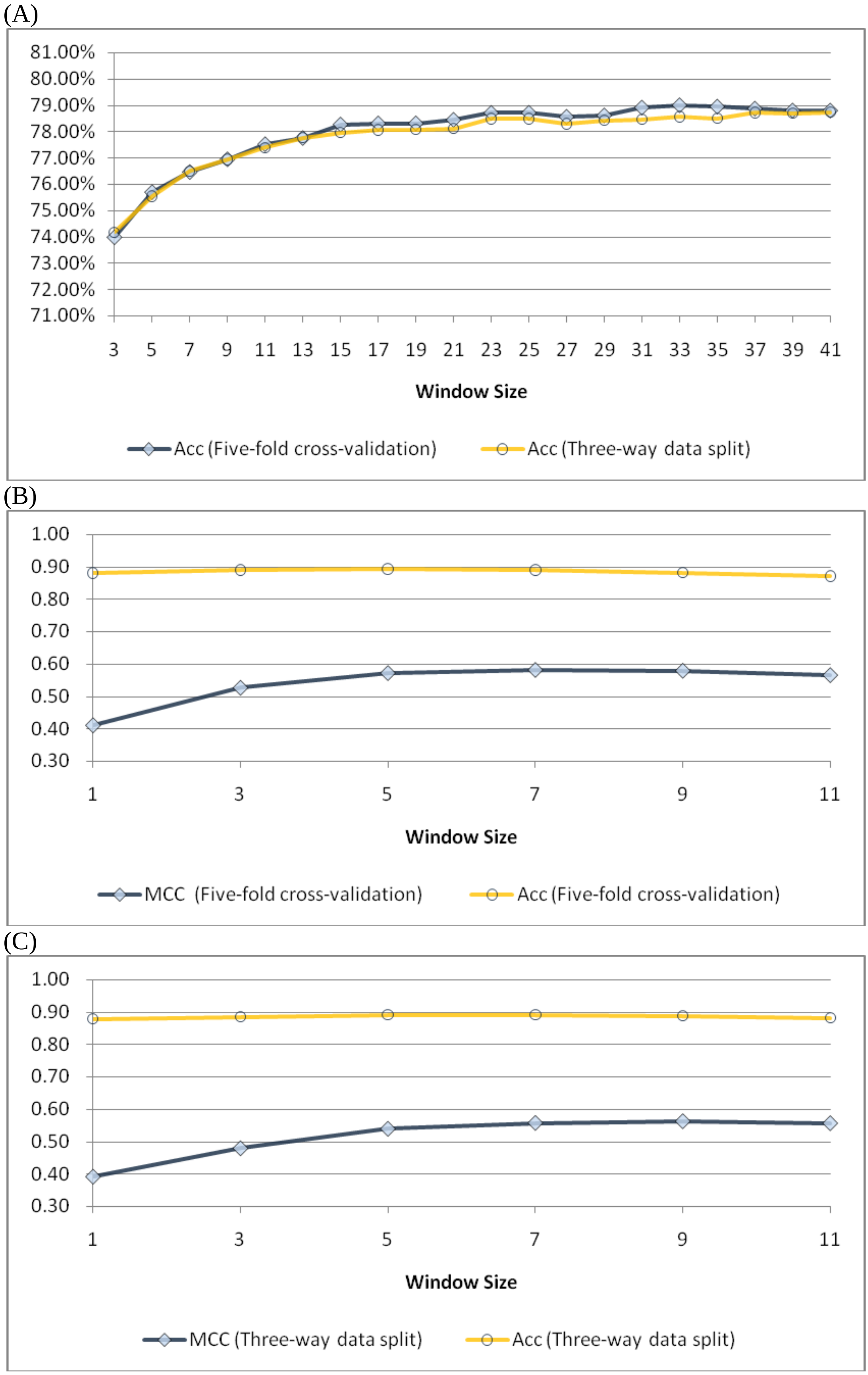


Figure 4.5: (A) Accuracy with respect to different sliding window sizes using five-fold cross-validation and three-way data split for the RBP109 data set, respectively. (B) The performance of the RBP109 data set with different smoothing window sizes by five-fold cross-validation. (C) The performance of the RBP109 data set with different smoothing window sizes by three-way data split.

3. Performance comparison with other approaches on the RBP107 data set

The prediction performance of different sliding and smoothing window sizes on the RBP107 data set is demonstrated in Figure 4.6. Similar to the RBP86 data set, we observe that the overall accuracy converges as sliding window size is greater than 25 on the RBP107 data set in Figure 4.6 (B). Moreover, the *MCC* shows a slight peak when the smoothing window size reaches 7 in Figure 4.6 (C). Thus RNAProB also selects *w* as 25 and *ws* as 7 for this data set. As illustrated in Table 4.4, our method reaches 0.42, 80.44%, 80.87%, and 77.14% in *MCC*, overall accuracy, specificity, and sensitivity by five-fold cross-validation (with w = 25, ws = 7, C = 4, γ = 0.015625, w_1 = 4, w_{-1} =1, and threshold value = 0.11), respectively. In addition, RNAProB also attains *MCC*, overall accuracy, specificity, and sensitivity of 0.40, 79.84%, 80.65%, and 73.62% by three-way data split (with w = 25, ws = 7, C = 8, γ = 0.015625, w_1 = 4, w_{-1} = 1, and threshold value = 0.12), correspondingly. The detailed experiment results on the RBP109 data set are summarized in the supplementary material [see Appendix 2.6].

Table 4.5 compares the performance of RNAProB with other approaches on the RBP107 data set. Based on physicochemical properties, BindN (i.e. referred to as BindN-PCP in Table 4.5) attains *MCC*, overall accuracy, specificity, and sensitivity of 0.27, 69.32%, 69.84%, and 66.28%, respectively (Wang and Brown, 2006). Incorporated with more biological features, BindN (i.e. denoted as BindN-ALL in Table 4.5) further improves specificity and accuracy by 5.86% and 4.93% with a slight decrease in sensitivity (Wang and Brown, 2006). PPRint improves sensitivity to 70.09% with the other measures performed comparable to those of BindN-ALL. Our method significantly outperforms the-state-of-the-art approaches by 0.10, 5.10%, 5.33%, and 7.05% in *MCC*, overall accuracy, specificity, and sensitivity, respectively. This demonstrates that RNAProB not only achieves accurate performance, but also substantially improves sensitivity in the prediction of RNA-binding sites.

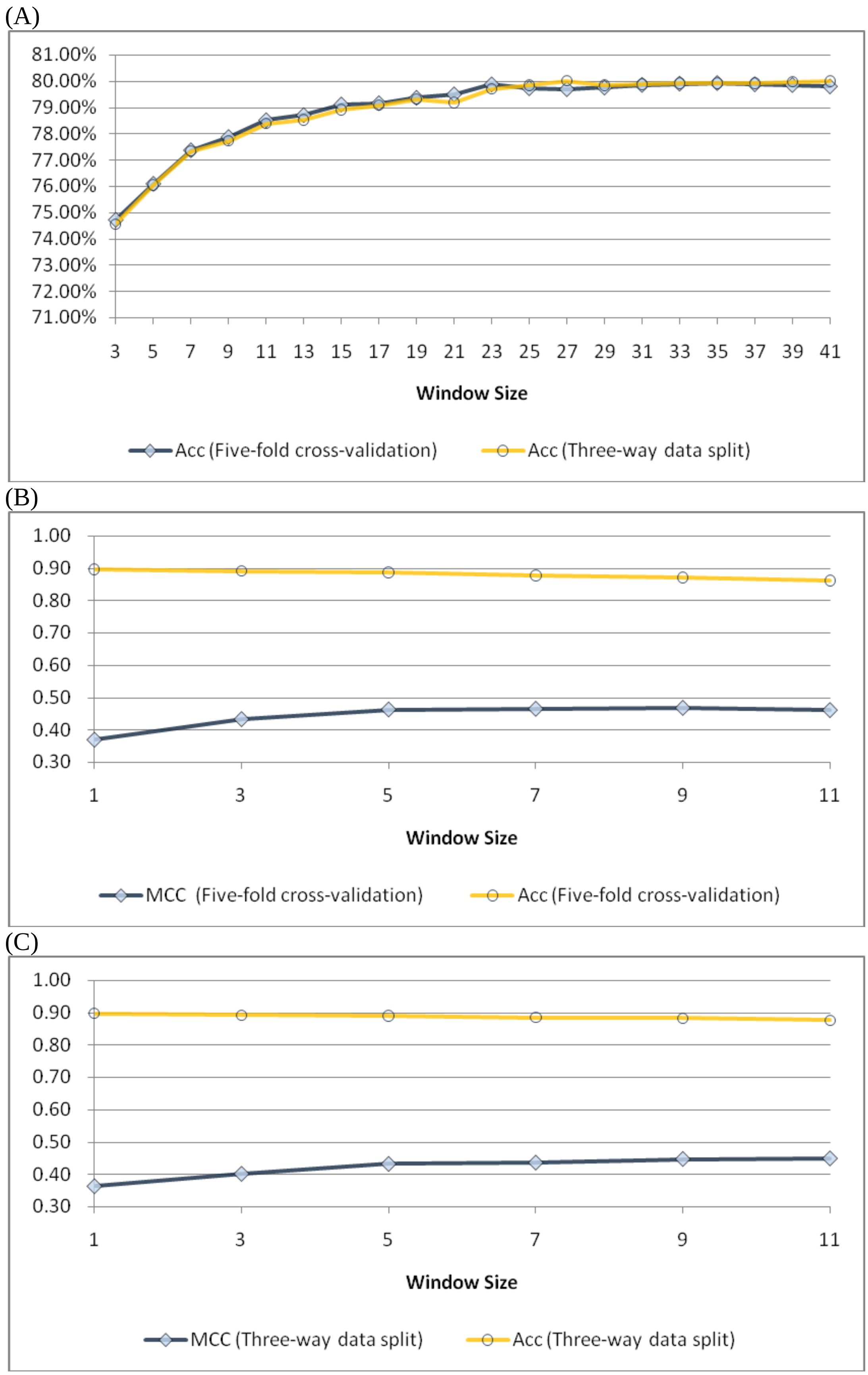
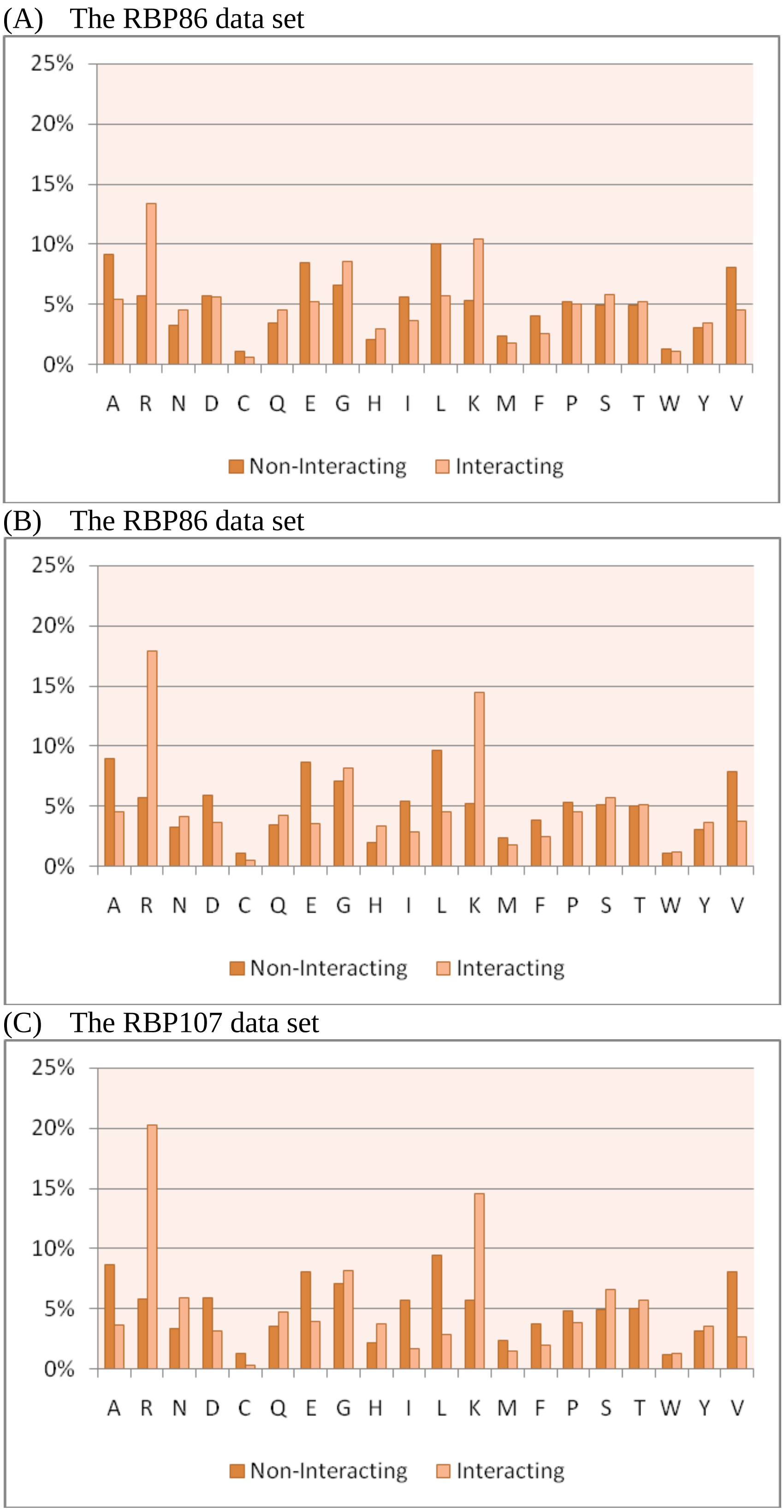


Figure 4.6: (A) Accuracy with respect to different sliding window sizes using five-fold cross-validation and three-way data split for the RBP107 data set, respectively. (B) The performance of the RBP107 data set with different smoothing window sizes by five-fold cross-validation. (C) The performance of the RBP107 data set with different smoothing window sizes by three-way data split.

4.3 Discussion

4.3.1 Physicochemical preferences of interacting and non-interacting residues

In this section, we examine the physicochemical properties of RNA interacting and non-interacting residues. Figure 4.7 (A), (B), and (C) show the amino acid compositions of interacting and non-interacting residues in the RBP86, RBP109, and RBP107 data sets, respectively. It is observed that interacting and non-interacting residues show preferences for different amino acids. RNA interacting residues tend to have high compositions for Arginine (R), Asparagine (N), Glutamine (Q), Glycine (G), Histidine (H), and Lysine (K). For example, there are relatively high proportions for Arginine (R) and Lysine (K), which may interact with negatively charged RNA with their positive side chains. In addition, the smallest amino acid, Glycine (G), also has a high composition in interacting residues because it rotates easily and provides flexibility to interact with RNA molecules. Moreover, positively charged Histidine (H) can have an aromatic interaction with RNA molecules due to its specific pKa value and imidazole ring. On the other hand, non-interacting residues show slight preferences for Alanine (A), Aspartic acid (D), Glutamic acid (E), Isoleucine (I), Leucine (L), Phenylalanine (F), and Valine (V). Cysteine (C), Aspartic acid (D), and Glutamic acid (E) are favoured by non-interacting residues because of their negatively charged side chains. In addition, although Kumar *et al.* (Kumar, et al., 2008) reported that Aspartic acid (D) showed no preference for interacting or non-interacting residues in their main data set (i.e., the RBP86 data set in our study), we observed that the Aspartic acid (D)


(A) The RBP86 data set

(B) The RBP86 data set

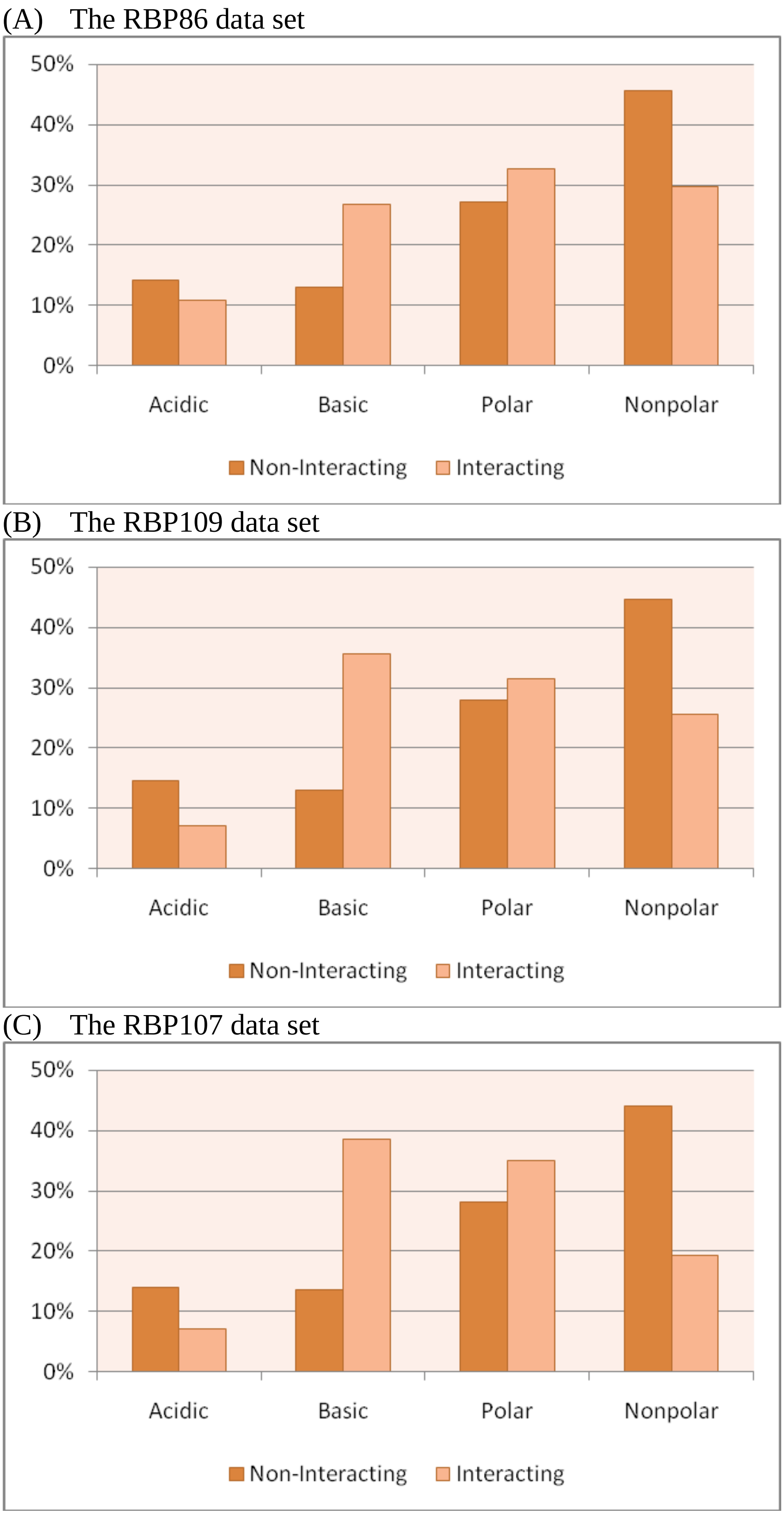
(C) The RBP107 data set

Figure 4.7: Amino acid compositions of interacting and non-interacting residues in the benchmark data sets.

composition of non-interacting residues is significantly higher than that of interacting residues in both of the RBP109 and RBP107 data sets. Our analysis indicates that the finding from Kumar *et al*. could be a bias from the data set.

To further analyze the physicochemical properties of the RNA interacting and non-interacting residues, each amino acid is classified into one of the four groups: acidic (DE), basic (HKR), polar (CGNQSTY), and non-polar (AFILMPVW) (Yu, et al., 2006). Figure 4.8 shows the grouped amino acid compositions of interacting and non-interacting residues for the benchmark data sets. It is observed among the three data sets that basic and polar amino acids tend to interact with RNA, and acidic and non-polar amino acids are not favoured by RNA molecules. Particularly, our analysis shows that the compositions of basic amino acids exhibit significantly over-represented patterns for interacting residues.

Furthermore, we inspect the amino acid compositions of proteins that interact with different RNA molecules. The proteins in the RBP109 data set are divided into four categories according to the definition in Terribilini *et al* (Terribilini, et al., 2006). Figure 4.9 (A), (B), (C), and (D) show the amino acid compositions of (A) rRNA, (B) mRNA, snRNA, dsRNA, and siRNA, (C) tRNA, and (D) viralRNA, respectively. It is observed that viralRNA group shows a different amino acid composition compared to the other groups. Proteins that interact with viralRNA evolve fast and induce conformational changes in the active sites. Thus, these proteins exhibit a specific mechanism to interact with viralRNA.

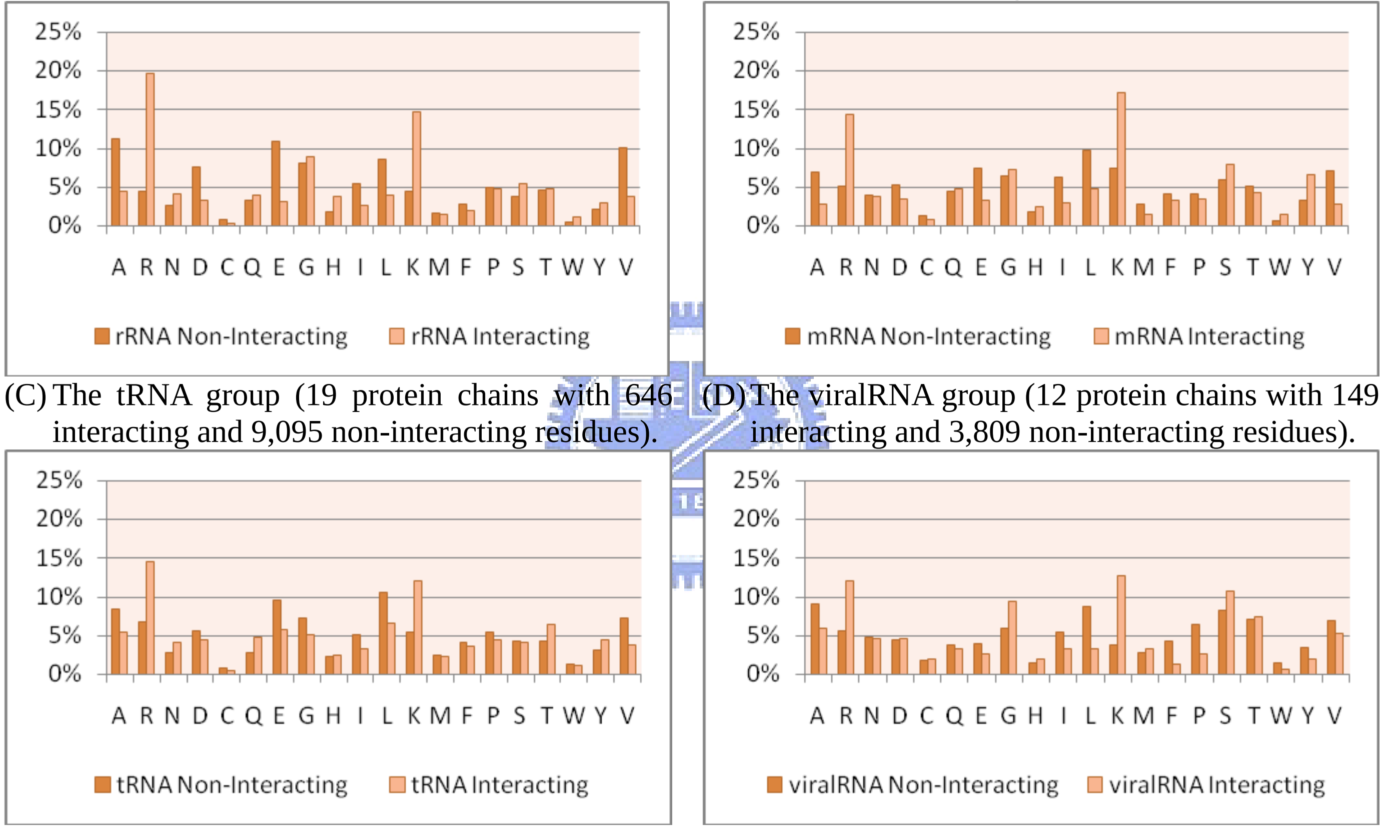


(A) The RBP86 data set

(B) The RBP109 data set

(C) The RBP107 data set

Figure 4.8: Grouped amino acid compositions of interacting and non-interacting residues in the benchmark data sets.



(A) The rRNA group (55 protein chains with 2,392 interacting and 5,302 non-interacting residues).

(B) The mRNA, snRNA, dsRNA, and siRNA group (23 protein chains with 394 interacting and 3,320 non-interacting residues).

(C) The tRNA group (19 protein chains with 646 interacting and 9,095 non-interacting residues).

(D) The viralRNA group (12 protein chains with 149 interacting and 3,809 non-interacting residues).

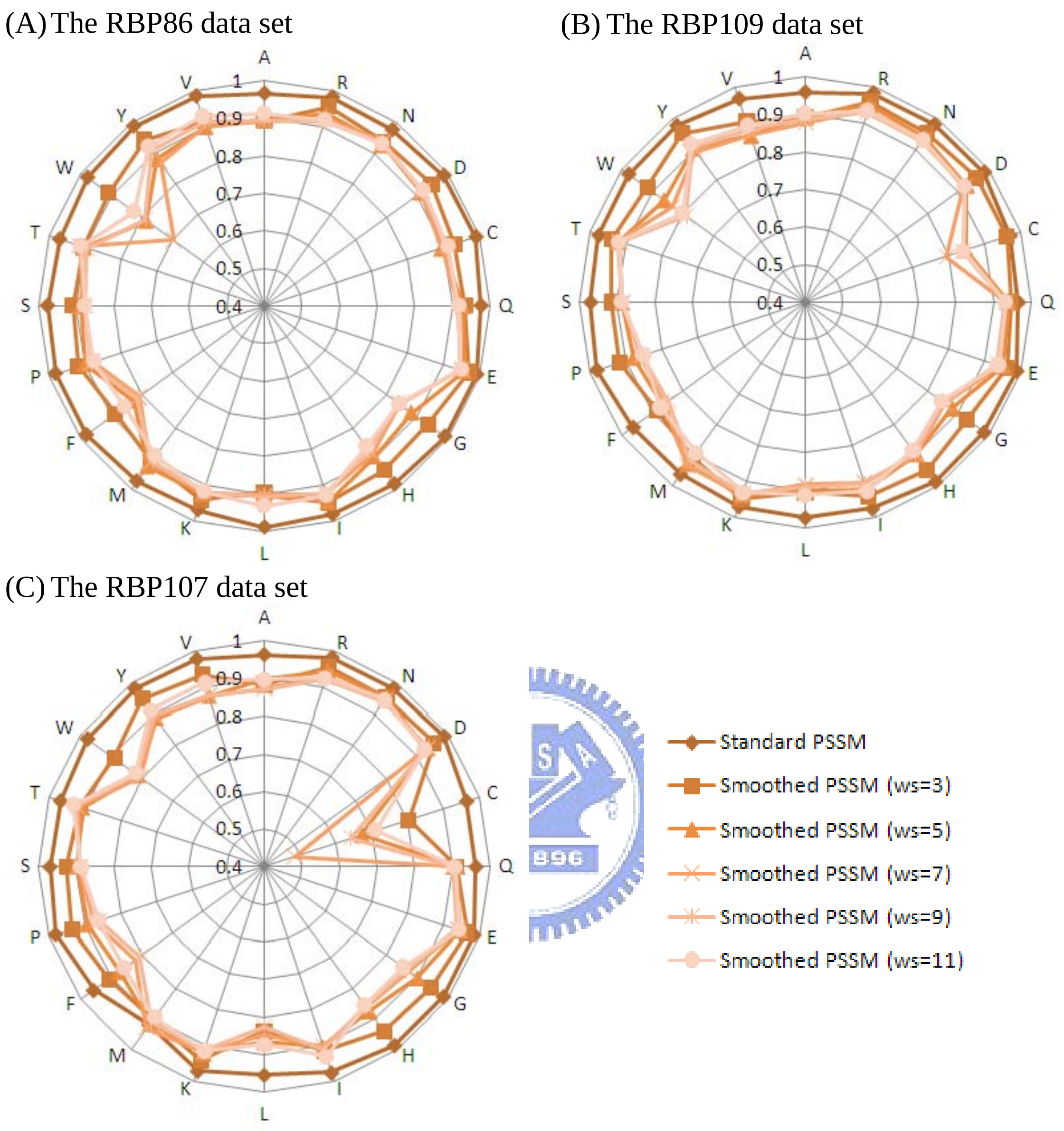
Figure 4.9: Amino acid compositions of interacting and non-interacting residues in four different RNA groups of the RBP109 data set.

4.3.2 Comparison of smoothed PSSM and standard PSSM

Here we examine the correlation between interacting and non-interacting residues for both smoothed PSSM and standard PSSM encoding schemes. We incorporate Pearson correlation coefficient (PCC) (Chang, et al., 2008) to measure the correlation between the evolutionary information of interacting and non-interacting for an amino acid. For each amino acid *a*, we use two vectors, *X* and *Y*, to present the sum of PSSM evolutionary information vectors for interacting and non-interacting amino acid *a*, respectively. The Pearson correlation coefficient for a series of *n* measurements for variables *X* and *Y* is defined in Equation (4.7).

$$\mathrm{PCC} = \mathrm{r_{xy}} = (\mathrm{n}\sum \mathrm{x_i y_i} - \sum \mathrm{x_i} \sum \mathrm{y_i}) / \sqrt{\mathrm{n}\sum \mathrm{x_i}^2 - (\sum \mathrm{x_i})^2}\sqrt{\mathrm{n}\sum \mathrm{y_i}^2 - (\sum \mathrm{y_i})^2} \quad (4.7)$$

Figure 4.10 shows the Pearson correlation coefficient between interacting and non-interacting evolutionary information vectors based on different PSSM encoding schemes in the benchmark data sets. It is observed that the correlation coefficients calculated from smoothed PSSM encoding scheme are lower than those from standard PSSM, especially for Cysteine (C) and Tryptophan (W). In Figure 4.10 (A), smoothed PSSM encoding attains lower correlation coefficients not only in interacting residues, such as Arginine (R), Asparagine (N), Glutamine (Q), Glycine (G), Histidine (H), and Lysine (K), but also in non-interacting residues, including Alanine (A), Aspartic acid (D), Glutamic acid (E), Isoleucine (I), Leucine (L), Phenylalanine (F), and Valine (V). Similarly, Figure 4.10 (B) and (C) also show lower correlation coefficients between interacting and non-interacting residues based on smoothed PSSM encoding. Furthermore, it is observed that the correlation coefficients calculated with smoothing window size *ws* = 7 are usually lower than those generated by other smoothing window sizes. If an encoding scheme leads to a lower Pearson correlation coefficient, it indicates that the encoding scheme can better resolve ambiguity in discriminating interacting residues from non-interacting ones. Our analysis lends support to our assumption that smoothed PSSM encoding scheme can improve the recognition RNA interacting and non-interacting sites by modelling the dependency from surrounding residues.


(A) The RBP86 data set

(B) The RBP109 data set

(C) The RBP107 data set

Figure 4.10: Pearson correlation coefficient between interacting and non-interacting evolutionary vectors generated by different PSSM encoding schemes in the benchmark data sets.

4.4 Conclusion

In this chapter, we present RNAProB, which combines a new smoothed PSSM encoding scheme with a SVM model for prediction of RNA-binding sites in proteins. In a standard PSSM profile, evolutionary information is calculated based on an assumption that each position is independent of others. However, the correlation or dependency from surrounding residues is incorporated in the proposed smoothed PSSM encoding. Experiment results show that the prediction performance of smoothed PSSM encoding performs better than the state-of-the-art approaches on the benchmark data sets. Evaluated by five-fold cross-validation, RNAProB outperforms the other approaches by 0.10~0.23 in *MCC*, 4.90%~6.83% in overall accuracy, and 0.88%~5.33% in specificity. Most notably, our method significantly improves sensitivity by 26.90%, 26.62%, and 7.05% for the RBP86, RBP109, and RBP107 data sets, respectively. Performance improvement in RNAProB not only demonstrates that smoothed PSSM can better resolve the ambiguity in discriminating RNA interacting and non-interacting residues, but also supports our assumption that consideration of correlation between neighboring residues can significantly enhance prediction accuracy. To prevent data overfitting, a rigorous three-way data split procedure is incorporated to evaluate our prediction performance. The proposed method can be used in other research topics, such as DNA-binding site prediction, protein-protein interaction, and prediction of post-translational modification sites.

References

1. Adamczak, R., Porollo, A. and Meller, J. (2005) Combining prediction of secondary structure and solvent accessibility in proteins, *Proteins*, **59**, 467-475.

2. Altschul, S.F., Madden, T.L., Schaffer, A.A., Zhang, J., Zhang, Z., Miller, W. and Lipman, D.J. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs, *Nucleic Acids Res*, **25**, 3389-3402.

3. Andrade, M.A., O'Donoghue, S.I. and Rost, B. (1998) Adaptation of protein surfaces to subcellular location, *J Mol Biol*, **276**, 517-525.

4. Bannai, H., Tamada, Y., Maruyama, O., Nakai, K. and Miyano, S. (2002) Extensive feature detection of N-terminal protein sorting signals, *Bioinformatics*, **18**, 298-305.

5. Baran, V., Colonna, M., Di Toro, M. and Greco, V. (2001) Nuclear fragmentation: Sampling the instabilities of binary systems, *Phys Rev Lett*, **86**, 4492-4495.

6. Baran, V., Colonna, M., Di Toro, M. and Larionov, A.B. (1998) Spinodal decomposition of low-density asymmetric nuclear matter, *Nucl Phys A*, **632**, 287-303.

7. Barranco, M. and Buchler, J.R. (1980) Thermodynamic Properties of Hot Nucleonic Matter, *Phys Rev C*, **22**, 1729-1737.

8. Bechara, E., Davidovic, L., Melko, M., Bensaid, M., Tremblay, S., Grosgeorge, J., Khandjian, E.W., Lalli, E. and Bardoni, B. (2007) Fragile X related protein 1 isoforms differentially modulate the affinity of fragile X mental retardation protein for G-quartet RNA structure, *Nucleic Acids Res*, **35**, 299-306.

9. Bendtsen, J.D., Jensen, L.J., Blom, N., Von Heijne, G. and Brunak, S. (2004) Feature-based prediction of non-classical and leaderless protein secretion, *Protein Eng Des Sel*, **17**, 349-356.

10. Bendtsen, J.D., Kiemer, L., Fausboll, A. and Brunak, S. (2005) Non-classical protein secretion in bacteria, *BMC Microbiol*, **5**, 58.

11. Bendtsen, J.D., Nielsen, H., von Heijne, G. and Brunak, S. (2004) Improved prediction of signal peptides: SignalP 3.0, *J Mol Biol*, **340**, 783-795.

12. Bendtsen, J.D., Nielsen, H., Widdick, D., Palmer, T. and Brunak, S. (2005) Pre-

diction of twin-arginine signal peptides, *BMC Bioinformatics*, **6**, 167.

13. Berks, B.C. (1996) A common export pathway for proteins binding complex redox cofactors?, *Mol Microbiol*, **22**, 393-404.

14. Berman, H.M., Battistuz, T., Bhat, T.N., Bluhm, W.F., Bourne, P.E., Burkhardt, K., Feng, Z., Gilliland, G.L., Iype, L., Jain, S., Fagan, P., Marvin, J., Padilla, D., Ravichandran, V., Schneider, B., Thanki, N., Weissig, H., Westbrook, J.D. and Zardecki, C. (2002) The Protein Data Bank, *Acta Crystallogr D Biol Crystallogr*, **58**, 899-907.

15. Bhasin, M., Garg, A. and Raghava, G.P.S. (2005) PSLpred: prediction of subcellular localization of bacterial proteins, *Bioinformatics*, **21**, 2522-2524.

16. Bradley, A.P. (1997) The use of the area under the ROC curve in the evaluation of machine learning algorithms *Pattern Recognition*, **30**, 1145-1159.

17. Cedano, J., Aloy, P., PerezPons, J.A. and Querol, E. (1997) Relation between amino acid composition and cellular location of proteins, *J Mol Biol*, **266**, 594-600.

18. Chang, C.C. and Lin, C.J. (2001) LIBSVM: a library for support vector machines, [http://www.csie.ntu.edu.tw/~cjlin/libsvm/].

19. Chang, J.M., Su, E.C., Lo, A., Chiu, H.S., Sung, T.Y. and Hsu, W.L. (2008) PSLDoc: Protein subcellular localization prediction based on gapped-dipeptides and probabilistic latent semantic analysis, *Proteins*, **72**, 693-710.

20. Cheng, B.Y., Carbonell, J.G. and Klein-Seetharaman, J. (2005) Protein classification based on text document classification techniques, *Proteins*, **58**, 955-970.

21. Chou, K.C. and Cai, Y.D. (2005) Predicting protein localization in budding yeast, *Bioinformatics*, **21**, 944-950.

22. Chou, K.C. and Elrod, D.W. (1999) Protein subcellular location prediction, *Protein Eng*, **12**, 107-118.

23. Chou, K.C. and Shen, H.B. (2006) Hum-PLoc: A novel ensemble classifier for predicting human protein subcellular localization, *Biochem Bioph Res Co*, **347**, 150-157.

24. Crick, F. (1970) Central dogma of molecular biology, *Nature*, **227**, 561-563.

25. Cserzo, M., Eisenhaber, F., Eisenhaber, B. and Simon, I. (2002) On filtering false positive transmembrane protein predictions, *Protein Eng*, **15**, 745-752.

26. Cuff, J.A. and Barton, G.J. (1999) Evaluation and improvement of multiple sequence methods for protein secondary structure prediction, *Proteins-Structure Function and Genetics*, **34**, 508-519.

27. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K. and Harshman, R. (1990) Indexing by Latent Semantic Analysis, *J Am Soc Inform Sci*, **41**, 391-407.

28. Dubchak, I., Muchnik, I., Holbrook, S.R. and Kim, S.H. (1995) Prediction of protein folding class using global description of amino acid sequence, *Proc Natl Acad Sci U S A*, **92**, 8700-8704.

29. Emanuelsson, O., Nielsen, H., Brunak, S. and von Heijne, G. (2000) Predicting subcellular localization of proteins based on their N-terminal amino acid sequence, *J Mol Biol*, **300**, 1005-1016.

30. Fontana, P., Bindewald, E., Toppo, S., Velasco, R., Valle, G. and Tosatto, S.C. (2005) The SSEA server for protein secondary structure alignment, *Bioinformatics*, **21**, 393-395.

31. Gardy, J.L. and Brinkman, F.S.L. (2006) Methods for predicting bacterial protein subcellular localization, *Nat Rev Microbiol*, **4**, 741-751.

32. Gardy, J.L., Laird, M.R., Chen, F., Rey, S., Walsh, C.J., Ester, M. and Brinkman, F.S.L. (2005) PSORTb v.2.0: Expanded prediction of bacterial protein subcellular localization and insights gained from comparative proteome analysis, *Bioinformatics*, **21**, 617-623.

33. Gardy, J.L., Spencer, C., Wang, K., Ester, M., Tusnady, G.E., Simon, I., Hua, S., deFays, K., Lambert, C., Nakai, K. and Brinkman, F.S.L. (2003) PSORT-B: improving protein subcellular localization prediction for Gram-negative bacteria, *Nucleic Acids Research*, **31**, 3613-3617.

34. Garg, A., Bhasin, M. and Raghava, G.P.S. (2005) Support vector machine-based method for subcellular localization of human proteins using amino acid compositions, their order, and similarity search, *J Biol Chem*, **280**, 14427-14432.

35. Garrow, A.G., Agnew, A. and Westhead, D.R. (2005) TMB-Hunt: a web server to screen sequence sets for transmembrane beta-barrel proteins, *Nucleic Acids Res*, **33**, W188-192.

36. Garrow, A.G., Agnew, A. and Westhead, D.R. (2005) TMB-Hunt: an amino acid composition based method to screen proteomes for beta-barrel transmembrane proteins, *BMC Bioinformatics*, **6**, 56.

37. Gonzalez, R.C. and Woods, R.E. (2002) *Digital Image Processing*. Prentice Hall.

38. Henikoff, S. and Henikoff, J.G. (1992) Amino acid substitution matrices from protein blocks, *Proc Natl Acad Sci U S A*, **89**, 10915-10919.

39. Henikoff, S. and Henikoff, J.G. (1992) Amino-Acid Substitution Matrices from Protein Blocks, *P Natl Acad Sci USA*, **89**, 10915-10919.

40. Hofmann, T. (2001) Unsupervised learning by probabilistic latent semantic analysis, *Mach Learn*, **42**, 177-196.

41. Hoglund, A., Donnes, P., Blum, T., Adolph, H.W. and Kohlbacher, O. (2006) MultiLoc: prediction of protein subcellular localization using N-terminal targeting sequences, sequence motifs and amino acid composition, *Bioinformatics*, **22**, 1158-1165.

42. Holland, I.B., Schmitt, L. and Young, J. (2005) Type 1 protein secretion in bacteria, the ABC-transporter dependent pathway (review), *Mol Membr Biol*, **22**, 29-39.

43. Horton, P., Park, K.J., Obayashi, T. and Nakai, K. (2006) Protein subcellular localization prediction with WoLF PSORT, *In Proceedings of the 4th Annual Asia Pacific Bioinformatics Conference (APBC'06): 13-16 February 2006; Taipei, Taiwan.*, 39-48.

44. Hua, S.J. and Sun, Z.R. (2001) A novel method of protein secondary structure prediction with high segment overlap measure: Support vector machine approach, *J Mol Biol*, **308**, 397-407.

45. Hua, S.J. and Sun, Z.R. (2001) Support vector machine approach for protein subcellular localization prediction, *Bioinformatics*, **17**, 721-728.

46. Jeong, E., Chung, I.F. and Miyano, S. (2004) A neural network method for identification of RNA-interacting residues in protein, *Genome Inform*, **15**, 105-116.

47. Jeong, E. and Miyano, S. (2006) A Weighted Profile Based Method for Protein-RNA Interacting Residue Prediction. *Transactions on Computational Systems Biology*. 123-139.

48. Jones, D.T. (1999) Protein secondary structure prediction based on position-specific scoring matrices, *J Mol Biol*, **292**, 195-202.

49. Krogh, A., Larsson, B., von Heijne, G. and Sonnhammer, E.L. (2001) Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes, *J Mol Biol*, **305**, 567-580.

50. Kumar, C.A., Gupta, A., Batool, M. and Trehan, S. (2006) Latent Semantic Indexing-Based Intelligent Information Retrieval System for Digital Libraries, *Journal of Computing and Information Technology*, **14**, 191-196.

51. Kumar, M., Gromiha, M.M. and Raghava, G.P. (2008) Prediction of RNA binding sites in a protein using SVM and PSSM profile, *Proteins*, **71**, 189-194.

52. Lee, K., Kim, D.W., Na, D., Lee, K.H. and Lee, D. (2006) PLPD: reliable protein localization prediction from imbalanced and overlapped datasets, *Nucleic Acids Res*, **34**, 4655-4666.

53. Li, W. and Godzik, A. (2006) Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences, *Bioinformatics*, **22**, 1658-1659.

54. Liang, H.K., Huang, C.M., Ko, M.T. and Hwang, J.K. (2005) Amino acid coupling patterns in thermophilic proteins, *Proteins*, **59**, 58-63.

55. Lin, C.J. and Chang, C.C. (2001) LIBSVM: a library for support vector machines.

56. Lin, H.N., Chang, J.M., Wu, K.P., Sung, T.Y. and Hsu, W.L. (2005) HYPROSP II - A knowledge-based hybrid method for protein secondary structure prediction based on local prediction confidence, *Bioinformatics*, **21**, 3227-3233.

57. Lin, H.N., Chang, J.M., Wu, K.P., Sung, T.Y. and Hsu, W.L. (2005) A knowledge-based hybrid method for protein secondary structure prediction based on local prediction confidence, *Bioinformatics*, **21**, 3227-3233.

58. Lu, Z., Szafron, D., Greiner, R., Lu, P., Wishart, D.S., Poulin, B., Anvik, J., Macdonell, C. and Eisner, R. (2004) Predicting subcellular localization of proteins using machine-learned classifiers, *Bioinformatics*, **20**, 547-556.

59. Manning, C.D. and Schütze, H. (1999) *Foundations of statistical natural language processing*. MIT Press, Cambridge, Mass.

60. Marcotte, E.M., Xenarios, I., van der Bliek, A.M. and Eisenberg, D. (2000) Localizing proteins in the cell from their phylogenetic profiles, *P Natl Acad Sci USA*, **97**, 12115-12120.

61. Matthews, B.W. (1975) Comparison of the predicted and observed secondary structure of T4 phage lysozyme, *Biochim Biophys Acta*, **405**, 442-451.

62. McGuffin, L.J., Bryson, K. and Jones, D.T. (2000) The PSIPRED protein structure prediction server, *Bioinformatics*, **16**, 404-405.

63. McKnight, K.L. and Heinz, B.A. (2003) RNA as a target for developing antivirals, *Antivir Chem Chemother*, **14**, 61-73.

64. Mott, R., Schultz, J., Bork, P. and Ponting, C.P. (2002) Predicting protein cellular localization using a domain projection method, *Genome Res*, **12**, 1168-1174.

65. Muller, H. and Serot, B.D. (1995) Phase-Transitions in Warm, Asymmetric Nuclear-Matter, *Phys Rev C*, **52**, 2072-2091.

66. Myers, E.W. and Miller, W. (1988) Optimal Alignments in Linear-Space, *Comput Appl Biosci*, **4**, 11-17.

67. Nair, R. and Rost, B. (2002) Sequence conserved for subcellular localization, *Protein Sci*, **11**, 2836-2847.

68. Nair, R. and Rost, B. (2003) Better prediction of sub-cellular localization by combining evolutionary and structural information, *Proteins-Structure Function and Genetics*, **53**, 917-930.

69. Nair, R. and Rost, B. (2003) LOC3D: annotate sub-cellular localization for protein structures, *Nucleic Acids Res*, **31**, 3337-3340.

70. Nair, R. and Rost, B. (2005) Mimicking cellular sorting improves prediction of subcellular localization, *J Mol Biol*, **348**, 85-100.

71. Nakai, K. and Horton, P. (1999) PSORT: a program for detecting sorting signals in proteins and predicting their subcellular localization, *Trends Biochem Sci*, **24**, 34-35.

72. Nakai, K. and Kanehisa, M. (1991) Expert system for predicting protein localization sites in gram-negative bacteria, *Proteins*, **11**, 95-110.

73. Namburu, S.M., Tu, H., Luo, J. and Pattipati, K.R. (2005) Experiments on Su-

pervised Learning Algorithms for Text Categorization. *Aerospace, 2005 IEEE Conference*. 1-8.

74. Nickel, W. (2003) The mystery of nonclassical protein secretion. A current view on cargo proteins and potential export routes, *Eur J Biochem*, **270**, 2109-2119.

75. Park, K.J. and Kanehisa, M. (2003) Prediction of protein subcellular locations by support vector machines using compositions of amino acids and amino acid pairs, *Bioinformatics*, **19**, 1656-1663.

76. Pautsch, A. and Schulz, G.E. (1998) Structure of the outer membrane protein A transmembrane domain, *Nat Struct Biol*, **5**, 1013-1017.

77. Pierleoni, A., Martelli, P.L., Fariselli, P. and Casadio, R. (2006) BaCelLo: a balanced subcellular localization predictor, *Bioinformatics*, **22**, e408-416.

78. Pugsley, A.P. (1993) The complete general secretory pathway in gram-negative bacteria, *Microbiol Rev*, **57**, 50-108.

79. Reinhardt, A. and Hubbard, T. (1998) Using neural networks for prediction of the subcellular location of proteins, *Nucleic Acids Research*, **26**, 2230-2236.

80. Rey, S., Acab, M., Gardy, J.L., Laird, M.R., DeFays, K., Lambert, C. and Brinkman, F.S.L. (2005) PSORTdb: a protein subcellular localization database for bacteria, *Nucleic Acids Research*, **33**, D164-D168.

81. Ritchie, M.D., White, B.C., Parker, J.S., Hahn, L.W. and Moore, J.H. (2003) Optimization of neural network architecture using genetic programming improves detection and modeling of gene-gene interactions in studies of human diseases, *BMC Bioinformatics*, **4**, 28.

82. Rost, B. and Sander, C. (1993) Prediction of Protein Secondary Structure at Better Than 70-Percent Accuracy, *J Mol Biol*, **232**, 584-599.

83. Salton, G. and Buckley, C. (1988) Term-Weighting Approaches in Automatic Text Retrieval, *Inform Process Manag*, **24**, 513-523.

84. Salton, G., Wong, A. and Yang, C.S. (1975) Vector-Space Model for Automatic Indexing, *Commun Acm*, **18**, 613-620.

85. Schneider, G. and Fechner, U. (2004) Advances in the prediction of protein targeting signals, *Proteomics*, **4**, 1571-1580.

86. Scott, M.S., Calafell, S.J., Thomas, D.Y. and Hallett, M.T. (2005) Refining protein subcellular localization, *Plos Comput Biol*, **1**, 518-528.

87. Sebastiani, F. (2002) Machine learning in automated text categorization, *Acm Comput Surv*, **34**, 1-47.

88. Su, C.Y., Lo, A., Chiu, H.S., Sung, T.Y. and Hsu, W.L. (2006) Protein subcellular localization prediction based on compartment-specific biological features. *IEEE Computational Systems Bioinformatics Conference (CSB'06).* Stanford, California, 325-330.

89. Su, C.Y., Lo, A., Chiu, H.S., Sung, T.Y. and Hsu, W.L. (2006) Protein subcellular localization prediction based on compartment-specific biological features, *In Proceedings of IEEE Computational Systems Bioinformatics Conference (CSB'06): 14-18 August 2006; Stanford, California*, 325-330.

90. Su, E.C., Chiu, H.S., Lo, A., Hwang, J.K., Sung, T.Y. and Hsu, W.L. (2007) Protein subcellular localization prediction based on compartment-specific features and structure conservation, *BMC Bioinformatics*, **8**, 330.

91. Sunita, S., Purta, E., Durawa, M., Tkaczuk, K.L., Swaathi, J., Bujnicki, J.M. and Sivaraman, J. (2007) Functional specialization of domains tandemly duplicated within 16S rRNA methyltransferase RsmC, *Nucleic Acids Res*, **35**, 4264-4274.

92. Swets, J.A. (1988) Measuring the accuracy of diagnostic systems, *Science*, **240**, 1285-1293.

93. Terribilini, M., Lee, J.H., Yan, C., Jernigan, R.L., Honavar, V. and Dobbs, D. (2006) Prediction of RNA binding sites in proteins from amino acid sequence, *RNA*, **12**, 1450-1462.

94. Terribilini, M., Sander, J.D., Lee, J.H., Zaback, P., Jernigan, R.L., Honavar, V. and Dobbs, D. (2007) RNABindR: a server for analyzing and predicting RNA-binding sites in proteins, *Nucleic Acids Res*, **35**, W578-584.

95. Thompson, J.D., Higgins, D.G. and Gibson, T.J. (1994) CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice, *Nucleic Acids Res*, **22**, 4673-4680.

96. Tsai, R.T., Sung, C.L., Dai, H.J., Hung, H.C., Sung, T.Y. and Hsu, W.L. (2006) NERBio: using selected word conjunctions, term normalization, and global pat-

terns to improve biomedical named entity recognition, *BMC Bioinformatics*, **7 Suppl 5**, S11.

97. Valdes-Perez, R.E., Pereira, F. and Pericliev, V. (2000) Concise, intelligible, and approximate profiling of multiple classes, *Int J Hum-Comput St*, **53**, 411-436.

98. Vapnik, V.N. (1995) *The Nature of Statistical Learning Theory*. Springer-Verlag, New York.

99. Vapnik, V.N. (1995) The Nature of Statistical Learning Theory, *New York: Springer-Verlag*.

100. Wang, J., Sung, W.K., Krishnan, A. and Li, K.B. (2005) Protein subcellular localization prediction for Gram-negative bacteria using amino acid subalphabets and a combination of multiple support vector machines, *BMC Bioinformatics*, **6**, 174.

101. Wang, L. and Brown, S.J. (2006) BindN: a web-based tool for efficient prediction of DNA and RNA binding sites in amino acid sequences, *Nucleic Acids Res*, **34**, W243-248.

102. Wang, L. and Brown, S.J. (2006) Prediction of RNA-binding residues in protein sequences using support vector machines, *Conf Proc IEEE Eng Med Biol Soc*, **1**, 5830-5833.

103. Wheeler, D.L., Barrett, T., Benson, D.A., Bryant, S.H., Canese, K., Chetvernin, V., Church, D.M., DiCuccio, M., Edgar, R., Federhen, S., Geer, L.Y., Kapustin, Y., Khovayko, O., Landsman, D., Lipman, D.J., Madden, T.L., Maglott, D.R., Ostell, J., Miller, V., Pruitt, K.D., Schuler, G.D., Sequeira, E., Sherry, S.T., Sirotkin, K., Souvorov, A., Starchenko, G., Tatusov, R.L., Tatusova, T.A., Wagner, L. and Yaschenko, E. (2007) Database resources of the National Center for Biotechnology Information, *Nucleic Acids Res*, **35**, D5-12.

104. Wickner, W. and Schekman, R. (2005) Protein translocation across biological membranes, *Science*, **310**, 1452-1456.

105. Wu, K.P., Lin, H.N., Chang, J.M., Sung, T.Y. and Hsu, W.L. (2004) HYPROSP: a hybrid protein secondary structure prediction algorithm - a knowledge-based approach, *Nucleic Acids Research*, **32**, 5059-5065.

106. Wu, T.F., Lin, C.J. and Weng, R.C. (2004) Probability estimates for multi-class classification by pairwise coupling, *J Mach Learn Res*, **5**, 975-1005.

107. Yu, C.S., Chen, Y.C., Lu, C.H. and Hwang, J.K. (2006) Prediction of protein subcellular localization, *Proteins*, **64**, 643-651.

108. Yu, C.S., Lin, C.J. and Hwang, J.K. (2004) Predicting subcellular localization of proteins for Gram-negative bacteria by support vector machines based on n-peptide compositions, *Protein Sci*, **13**, 1402-1406.

Appendix 1.1

The Second Encoding Scheme for Secondary Structure Elements

To depict secondary structure elements (SSE), i.e., *α*-helix (H), *β*-strand (E), and loop (L), in a protein, three descriptors, composition (C), transition (T), and distribution (D), are used to encode predictions from HYPROSP II using Equations (1), (2), and (3), respectively [1].

$$C_i = (n_i / N) \times 100\%,\ \sum_i C_i = 100\%,\ i \in \{\mathrm{H}, \mathrm{E}, \mathrm{L}\} \tag{1}$$

$$T_{i \leftrightarrow j} = \left[t_{i \leftrightarrow j} / \left(N - 1 \right) \right] \times 100\%,\ i, j \in \{\mathrm{H}, \mathrm{E}, \mathrm{L}\},\ i \neq j \tag{2}$$

$$D_i^{p\%} = \left(d_i^{p\%} / N \right) \times 100\%,\ i \in \{\mathrm{H}, \mathrm{E}, \mathrm{L}\},\ p = \{1, 25, 50, 75, 100\}, \tag{3}$$

where C_i represents the composition of SSE type *i*; n_i is the number of SSE type *i* in a protein; *N* is the total number of amino acid residues in a protein; $T_{i \leftrightarrow j}$ measures the transition between SSE type *i* and *j*; $t_{i \leftrightarrow j}$ is the number of transitions between SSE type *i* and *j*; $D_i^{p\%}$ gives the distribution of *p%* located SSE type *i*; and $d_i^{p\%}$ is the position of *p%* located SSE type *i* in a protein.

For illustration purposes, a hypothetical SSE sequence is shown in Figure 1.1S. The sequence includes 12 *α*-helix residues (n_H = 12) and 8 *β*-strand residues (n_E = 8). The percent compositions are calculated as follows: n_H / (n_H + n_E + n_L) × 100% = 60.0% for H, n_E / (n_H + n_E + n_L) × 100% = 40.0% for E, and n_L / (n_H + n_E + n_L) × 100% = 0.0% for L. These three numbers represent the first descriptor, C. The second descriptor, T, characterizes the percent frequency that amino acids of a particular secondary structure element type are followed by a different type. In this case, there are 4 transitions of H to E or E to H, $T_{H\leftrightarrow E}$ is represented by (4 / 19) × 100.0% = 21.1%. $T_{H\leftrightarrow L}$ and $T_{E\leftrightarrow L}$ are equal to 0.0%, respectively. The third descriptor, D, measures the chain length within which the first, 25%, 50%, 75%, and 100% of the amino acids of a particular secondary structure element type are located. In Figure 2.1S, the first percent residue of α-helices coincides with the beginning of the chain, so the $D_H^{1\%}$ descriptor equals 0.0%. Twenty-five percent of α-helix residues are contained within the first 3 residues of the protein chain, so the second number equals (3 / 20) × 100.0% = 15.0%. Fifty percent of α-helix residues are within the first 11 residues of the chain; thus, the third number is (11 / 20) × 100.0% = 55.0%. The fourth and fifth numbers of the distribution descriptor are 70.0% and 100.0%, respectively. Similarly, analogous numbers for β-strand and loop residues are calculated.

SSE sequence	**H**	**H**	**H**	**H**	**E**	**E**	**E**	**E**	**E**	**H**	**H**	**H**	**H**	**H**	**E**	**E**	**E**	**H**	**H**	**H**
SSE index	1				5					10					15					20
Index for **H**	1	2	3	4						5	6	7	8	9				10	11	12
Index for **E**					1	2	3	4	5						6	7	8			
H ↔ **E** transitions					|					|				|			|		

Figure 1.1S: A hypothetical SSE sequence that illustrates the derivation of the SSE2 feature vector for a protein.

Appendix 1.2

Data Sets

1. We provide the data sets for protein subcellular localization prediction used in the training and testing of the proposed method. Table 1.1S lists the links to download the benchmark data sets, the non-redundant data set, and the evaluation data sets described in the main paper.

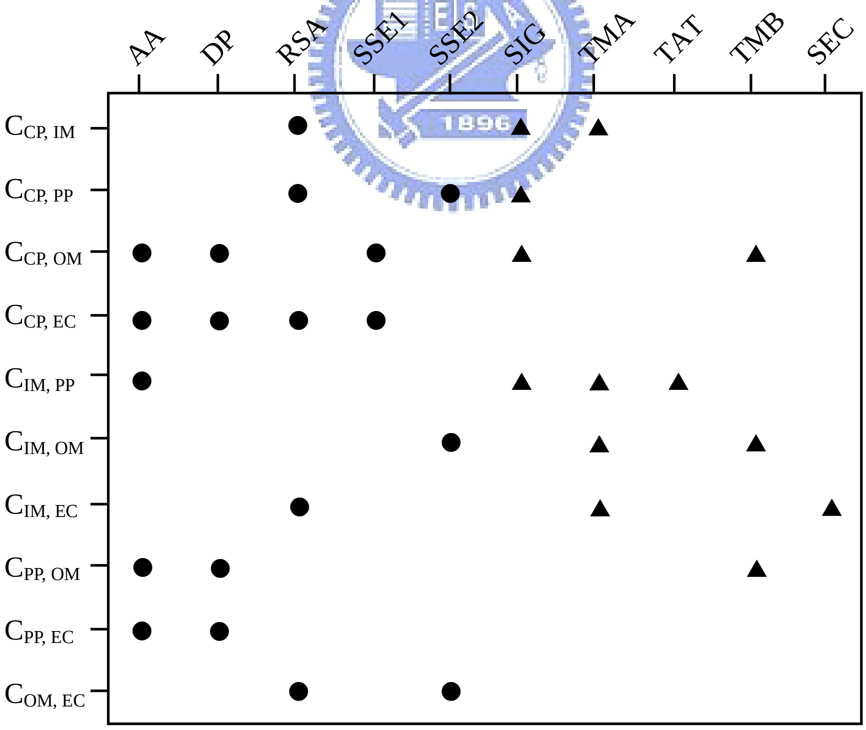
Table 1.1S: The benchmark, non-redundant, and evaluation data sets used in PSL101.

Data Set	Number of Proteins	Link
PS1302	1302	http://bio-cluster.iis.sinica.edu.tw/~bioapp/PSL101/dataset/PS1302.fasta
PS1444	1444	http://bio-cluster.iis.sinica.edu.tw/~bioapp/PSL101/dataset/PS1444.fasta
NR755	755	http://bio-cluster.iis.sinica.edu.tw/~bioapp/PSL101/dataset/NR755.fasta
NR828	828	http://bio-cluster.iis.sinica.edu.tw/~bioapp/PSL101/dataset/NR828.fasta
EV90_high	90	http://bio-cluster.iis.sinica.edu.tw/~bioapp/PSL101/dataset/EV90_high.fasta
EV153_low	153	http://bio-cluster.iis.sinica.edu.tw/~bioapp/PSL101/dataset/EV153_low.fasta
EV243_all	243	http://bio-cluster.iis.sinica.edu.tw/~bioapp/PSL101/dataset/EV243_all.fasta

Appendix 1.3

Selected Feature Combinations

The features selected from PSL101 for the PS1302 data set using a three-way data split procedure are shown in Figure 1.2S. The selected features correspond well to those discussed in the main paper. In Figure 1.2S, PSL101 also selects SIG, TMA, and RSA as the optimal features to distinguish cytoplasmic and inner membrane proteins. In addition, RSA, TMA, and SEC are used in the discrimination of proteins localized in the inner membrane and extracellular space. The results indicate that the selected features are highly correlated with the biological insights derived from Gram-negative bacteria translocation pathways.



	AA	DP	RSA	SSE1	SSE2	SIG	TMA	TAT	TMB	SEC
$C_{CP, IM}$			●			▲	▲			
$C_{CP, PP}$			●		●	▲				
$C_{CP, OM}$	●	●		●		▲			▲	
$C_{CP, EC}$	●	●	●	●						
$C_{IM, PP}$	●					▲	▲	▲		
$C_{IM, OM}$					●		▲		▲	
$C_{IM, EC}$			●				▲			▲
$C_{PP, OM}$	●	●							▲	
$C_{PP, EC}$	●	●								
$C_{OM, EC}$			●		●					

Figure 1.2S. Feature combinations derived from the PS1302 data set using a three-way data split procedure. Selected general and compartment-specific features are represented by filled circles and triangles, respectively.

REFERENCE

1. Dubchak I, Muchnik I, Holbrook SR, Kim SH: **Prediction of protein folding class using global description of amino acid sequence**. *Proc Natl Acad Sci USA* 1995, **92**(19):8700-8704.

Appendix 2.1

The RBP86 Data Set

2BBV_C	1F7U_A	1I6U_A	1JJ2_O	1JJ2_S
1A9N_D	1F7Y_A	1IVS_B	1JJ2_P	1JJ2_U
1ASZ_B	1F8V_A	1JBR_A	1JJ2_Q	1JJ2_W
1AV6_A	1FEU_D	1JID_A	1JJ2_R	1JJ2_Y
1B23_P	1FJG_B	1JJ2_1	1JJ2_T	1K8W_A
1B7F_B	1FJG_D	1JJ2_2	1JJ2_V	1KQ2_M
1C0A_A	1FJG_L	1JJ2_A	1JJ2_X	1M5O_F
1C9S_L	1FJG_M	1JJ2_B	1JJ2_Z	1M8Y_B
1CVJ_H	1FJG_S	1JJ2_C	1KNZ_A	1MMS_A
1DDL_C	1FJG_T	1JJ2_D	1LNG_A	1N35_A
1DFU_P	1G1X_F	1JJ2_E	1M8V_M	1NB7_B
1DI2_B	1G1X_G	1JJ2_H	1MJI_A	1QTQ_A
1DRZ_A	1G1X_H	1JJ2_I	1MZP_A	1SER_B
1DUL_A	1G2E_A	1JJ2_J	1N78_B	2A8V_B
1E6T_C	1H38_D	1JJ2_K	1QF6_A	1JJ2_S
1E7K_B	1H3E_A	1JJ2_L	1QU2_A	1JJ2_U
1EC6_A	1H4S_B	1JJ2_M	1URN_C	1JJ2_W
1EFW_B	1HQ1_A	1JJ2_N	2FMT_B	1JJ2_Y

Appendix 2.2

The RBP109 Data Set

1A34_A	1FJG_D	1H3E_A	1JJ2_O	1N78_A
1A9N_A	1FJG_E	1HR0_W	1JJ2_P	1NB7_A
1ASY_A	1FJG_G	1I6U_A	1JJ2_Q	1OOA_A
1AV6_A	1FJG_I	1J1U_A	1JJ2_R	1PGL_2
1B23_P	1FJG_J	1J2B_A	1JJ2_S	1Q2R_A
1COA_A	1FJG_K	1JBR_A	1JJ2_T	1QF6_A
1DDL_A	1FJG_L	1JID_A	1JJ2_U	1QTQ_A
1DFU_P	1FJG_M	1JJ2_1	1JJ2_V	1R3E_A
1DI2_A	1FJG_N	1JJ2_2	1JJ2_W	1RC7_A
1E6T_A	1FJG_P	1JJ2_A	1JJ2_X	1RMV_A
1E7K_A	1FJG_Q	1JJ2_B	1JJ2_Y	1RPU_A
1E8O_A	1FJG_S	1JJ2_C	1JJ2_Z	1S03_G
1E8O_B	1FJG_T	1JJ2_D	1K8W_A	1SER_A
1EC6_A	1FJG_V	1JJ2_E	1KNZ_A	1SI3_A
1EIY_A	1FXL_A	1JJ2_F	1KQ2_A	1UN6_B
1EIY_B	1G1X_A	1JJ2_G	1LAJ_A	1URN_A
1F7U_A	1G1X_B	1JJ2_H	1LNG_A	1UVJ_A
1F8V_A	1G1X_C	1JJ2_I	1M8V_A	2A8V_A
1FEU_A	1G1X_G	1JJ2_J	1MFQ_C	2BBV_C
1FFY_A	1GAX_A	1JJ2_K	1MMS_A	2BBV_F
1FJG_B	1GTF_Q	1JJ2_L	1MZP_A	2FMT_A
1FJG_C	1H2C_A	1JJ2_M	1N35_A	1N78_A

Appendix 2.3

The RBP107 Data Set

1A1V_A	1GAX_A	1JJ2_O	1RMV_A	1XMQ_M
1A34_A	1H38_A	1JJ2_P	1RPU_A	1XMQ_N
1A9N_A	1HC8_A	1JJ2_Q	1S72_H	1XMQ_P
1APG_A	1HQ1_A	1JJ2_R	1SER_B	1XMQ_Q
1AQ3_A	1I6U_A	1JJ2_S	1SI2_A	1XMQ_R
1ASY_A	1JBR_A	1JJ2_T	1TTT_A	1XMQ_S
1BMV_2	1JID_A	1JJ2_U	1U0B_B	1XMQ_T
1C0A_A	1JJ2_Y	1JJ2_V	1URN_A	1XMQ_V
1C9S_A	1JJ2_z	1JJ2_W	1UVJ_A	1XOK_C
1CVJ_A	1JJ2_A	1JJ2_X	1WMQ_A	1XOK_D
1CWP_A	1JJ2_1	1JJ2_G	1WNE_A	1YTU_A
1DDL_A	1JJ2_B	1K8W_A	1XMQ_B	1YVP_A
1DFU_P	1JJ2_2	1KNZ_A	1XMQ_C	1ZE2_A
1DI2_A	1JJ2_C	1KOG_A	1XMQ_D	1ZE2_B
1E7K_A	1JJ2_D	1L9A_A	1XMQ_E	1ZJW_A
1E8O_A	1JJ2_E	1M8Y_A	1XMQ_F	2A8V_A
1E8O_B	1JJ2_I	1MFQ_C	1XMQ_G	2BBV_F
1EC6_A	1JJ2_J	1MZP_A	1XMQ_H	2BH2_A
1F8V_A	1JJ2_K	1OOA_A	1XMQ_I	1XMQ_M
1FFY_A	1JJ2_L	1P6V_A	1XMQ_J	
1FKA_O	1JJ2_M	1Q2R_A	1XMQ_K	
1G59_A	1JJ2_N	1QF6_A	1XMQ_L	

Appendix 2.4

Experiment results of the RBP86

Table A 1: The detail performance of RBP86 with (A) different sliding window size under five-fold cross-validation (w1 = 3.39, w-1 = 1, other parameters: default value) and (B) different sliding window size under three-way data split (w1 = 3.39, w-1 = 1, other parameters: default value).

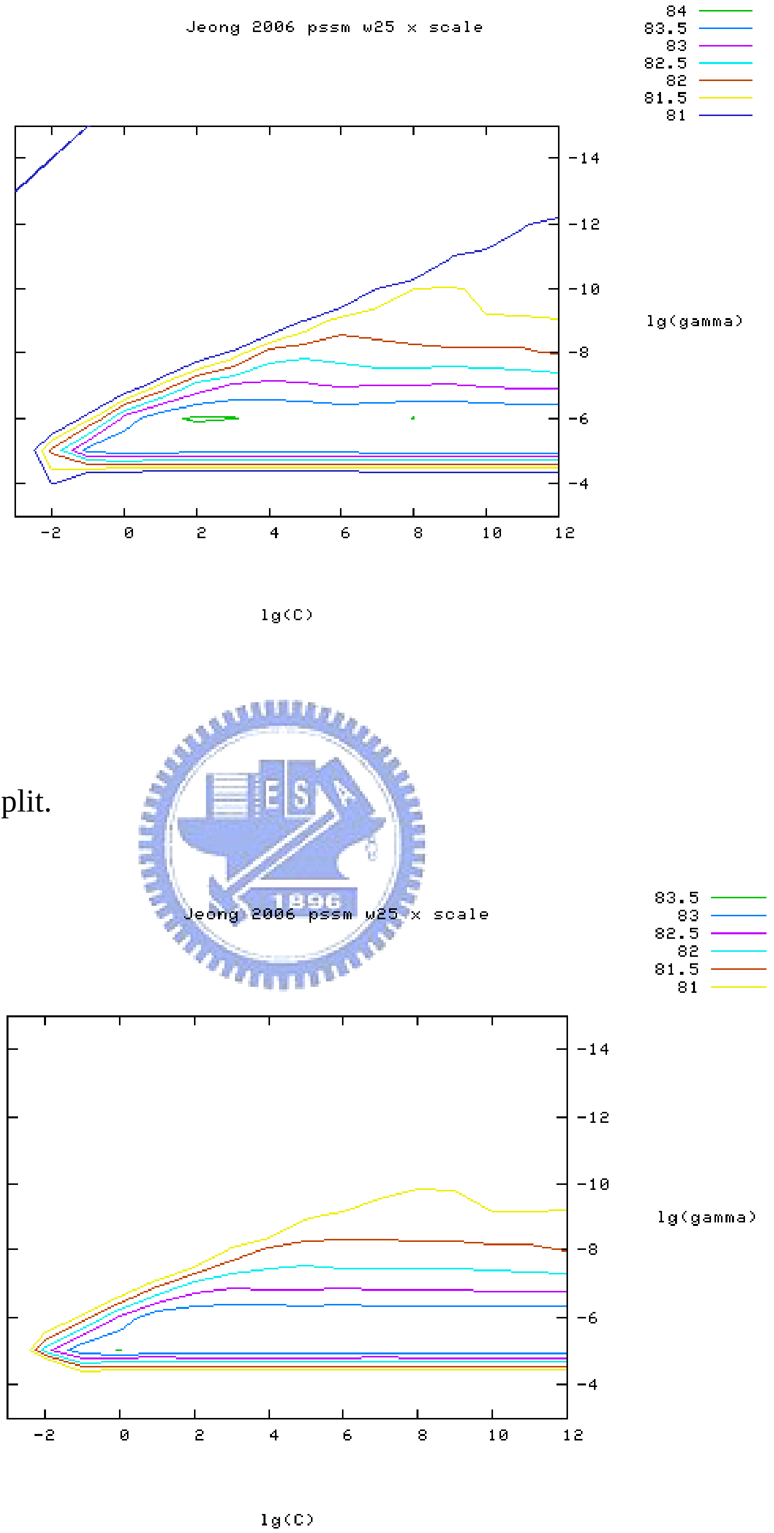
(A) Five-fold cross-validation

Window Size	Spec.	Sens.	MCC	Acc
3	73.38%	66.16%	0.35	71.74%
5	75.60%	66.94%	0.38	73.63%
7	76.81%	67.62%	0.40	74.72%
9	77.30%	69.33%	0.41	75.49%
11	77.92%	68.96%	0.42	75.88%
13	78.06%	69.00%	0.42	76.00%
15	78.71%	69.33%	0.43	76.58%
17	78.77%	69.48%	0.43	76.65%
19	78.87%	69.40%	0.43	76.71%
21	78.75%	69.46%	0.43	76.64%
23	78.93%	69.94%	0.44	76.88%
25	79.22%	69.75%	0.44	77.06%
27	79.24%	69.88%	0.44	77.11%
29	79.38%	69.66%	0.44	77.17%
31	79.26%	69.70%	0.44	77.08%
33	79.36%	69.77%	0.44	77.18%
35	79.41%	69.86%	0.44	77.24%
37	79.77%	69.55%	0.45	77.45%
39	79.50%	69.81%	0.44	77.30%
41	79.66%	70.05%	0.45	77.47%

(B) Three-way data split

Window Size	Spec.	Sens.	MCC	Acc
3	73.59%	66.20%	0.35	71.91%
5	75.59%	66.24%	0.37	73.46%
7	76.77%	66.81%	0.39	74.50%
9	76.93%	68.08%	0.40	74.91%
11	77.70%	68.13%	0.41	75.52%
13	78.17%	68.35%	0.42	75.93%
15	78.44%	68.43%	0.42	76.16%
17	78.60%	68.59%	0.42	76.32%
19	78.77%	68.65%	0.43	76.46%
21	78.85%	69.24%	0.43	76.66%
23	79.20%	68.78%	0.43	76.83%
25	79.38%	69.35%	0.44	77.10%
27	79.47%	69.40%	0.44	77.18%
29	79.45%	69.26%	0.44	77.13%
31	79.41%	69.70%	0.44	77.20%
33	79.40%	69.33%	0.44	77.11%
35	79.58%	69.13%	0.44	77.21%
37	79.82%	69.00%	0.44	77.36%
39	79.67%	69.33%	0.44	77.32%
41	79.78%	69.37%	0.44	77.42%

(A) Five-fold cross-validation.



(B) Three-way data split.

Figure A 1: The performance with different combination of C and γ in the RBP86 data set under (A) five-fold cross-validation and (B) three-way data split.

Table A 2. The detail performance of the RBP86 with (A) different smoothing window size under five-fold cross-validation (w = 25, log C = 2, log γ = -6, w1 = 3.39, w-1 = 1, other parameters: default value) and (B) different smoothing window size under three-way data split (w = 25, log C = 0, log γ = -5, w1 = 3.39, w-1 = 1, other parameters: default value).

(A) Five-fold cross-validation.

Smoothing Window Size	Spec.	Sens.	MCC	Acc
1	94.37%	48.18%	0.50	83.86%
3	92.95%	69.64%	0.64	87.64%
5	91.67%	76.58%	0.67	88.23%
7	90.12%	80.36%	0.67	87.90%
9	89.28%	82.82%	0.68	87.81%
11	88.34%	84.37%	0.68	87.44%

(B) Three-way data split.

Smoothing Window Size	Spec.	Sens.	MCC	Acc
1	96.40%	38.68%	0.46	83.26%
3	92.14%	66.53%	0.60	86.31%
5	90.72%	76.95%	0.66	87.58%
7	89.65%	80.34%	0.67	87.53%
9	88.62%	82.84%	0.67	87.31%
11	88.04%	83.87%	0.67	87.09%

Table A 3. The detail performance of the RBP86 with (A) different weight parameter w1 under five-fold cross-validation (w = 25, log C = 2, log γ = -6, ws = 7, w-1 = 1, other parameters: default value) and (B) different weight parameter w1 under three-way data split (w = 25, log C = 0, log γ = -5, ws = 7, w-1 = 1, other parameters: default value).

(A) Five-fold cross-validation.

W1	Spec.	Sens.	MCC	Acc
1	96.01%	62.26%	0.65	88.33%
2	92.33%	75.28%	0.67	88.45%
3	90.60%	79.73%	0.68	88.13%
4	89.80%	81.37%	0.68	87.88%
5	89.42%	81.74%	0.67	87.67%
6	89.23%	82.05%	0.67	87.60%
7	89.12%	82.20%	0.67	87.55%
8	89.04%	82.18%	0.67	87.48%

(B) Three-way data split.

W1	Spec.	Sens.	MCC	Acc
1	96.79%	56.41%	0.62	87.60%
2	92.17%	73.53%	0.66	87.93%
3	90.18%	78.94%	0.66	87.62%
4	89.07%	81.68%	0.67	87.38%
5	88.56%	82.44%	0.67	87.17%
6	88.22%	82.73%	0.66	86.97%
7	88.08%	82.86%	0.66	86.89%
8	88.02%	82.97%	0.66	86.87%

Table A 4. The RBP86 data set experiment results with –b option in SVM for (A) smoothed PSSM by five-fold cross-validation, (B) standard PSSM by five-fold cross-validation, (C) smoothed PSSM by three-way data split, and (D) standard PSSM by three-way data split.

(A) The experiment result of smoothed PSSM with –b option in SVM by five-fold cross-validation.

Threshold	Spec.	Sens.	MCC	Threshold	Spec.	Sens.	MCC
0	0.00%	100.00%	0.00	**0.51**	93.83%	71.21%	0.67
0.01	13.08%	99.54%	0.17	**0.52**	93.96%	70.71%	0.67
0.02	27.49%	99.01%	0.27	**0.53**	94.18%	70.07%	0.67
0.03	39.48%	98.16%	0.34	**0.54**	94.43%	69.29%	0.67
0.04	48.89%	97.31%	0.40	**0.55**	94.61%	68.78%	0.67
0.05	56.51%	96.67%	0.45	**0.56**	94.74%	68.13%	0.66
0.06	62.70%	96.06%	0.49	**0.57**	94.90%	67.21%	0.66
0.07	67.71%	95.10%	0.53	**0.58**	95.08%	66.35%	0.66
0.08	70.92%	94.53%	0.55	**0.59**	95.23%	65.41%	0.65
0.09	73.43%	93.76%	0.57	**0.6**	95.35%	64.89%	0.65
0.1	75.29%	93.15%	0.59	**0.61**	95.58%	64.14%	0.65
0.11	77.03%	92.64%	0.60	**0.62**	95.79%	63.35%	0.65
0.12	78.46%	92.05%	0.61	**0.63**	95.94%	62.70%	0.65
0.13	79.69%	91.46%	0.62	**0.64**	96.10%	61.93%	0.65
0.14	80.71%	90.98%	0.63	**0.65**	96.32%	60.68%	0.64
0.15	81.67%	90.56%	0.64	**0.66**	96.52%	59.63%	0.64
0.16	82.48%	90.06%	0.64	**0.67**	96.72%	58.67%	0.64
0.17	83.23%	89.45%	0.65	**0.68**	96.88%	57.42%	0.63
0.18	83.76%	88.88%	0.65	**0.69**	97.06%	56.44%	0.63
0.19	84.33%	88.53%	0.65	**0.7**	97.17%	55.36%	0.62
0.2	84.87%	88.03%	0.66	**0.71**	97.37%	54.07%	0.62
0.21	85.42%	87.52%	0.66	**0.72**	97.53%	52.74%	0.61
0.22	85.89%	87.00%	0.66	**0.73**	97.67%	51.29%	0.60
0.23	86.34%	86.67%	0.67	**0.74**	97.79%	50.00%	0.59
0.24	86.73%	86.25%	0.67	**0.75**	97.96%	48.71%	0.59
0.25	87.14%	85.73%	0.67	**0.76**	98.06%	46.61%	0.57
0.26	87.48%	85.22%	0.67	**0.77**	98.20%	44.51%	0.56
0.27	87.86%	84.63%	0.67	**0.78**	98.37%	42.78%	0.55
0.28	88.20%	84.15%	0.67	**0.79**	98.52%	41.20%	0.54
0.29	88.50%	83.65%	0.67	**0.8**	98.61%	39.01%	0.53
0.3	88.76%	83.17%	0.67	**0.81**	98.75%	37.41%	0.52
0.31	89.06%	82.71%	0.68	**0.82**	98.87%	35.49%	0.50
0.32	89.37%	82.14%	0.68	**0.83**	98.95%	33.69%	0.49
0.33	89.54%	81.87%	0.68	**0.84**	99.03%	31.94%	0.48
0.34	89.77%	81.33%	0.68	**0.85**	99.18%	29.62%	0.46
0.35	90.06%	80.71%	0.68	**0.86**	99.34%	27.50%	0.45
0.36	90.36%	79.95%	0.68	**0.87**	99.44%	25.15%	0.43
0.37	90.66%	79.18%	0.67	**0.88**	99.55%	23.05%	0.41
0.38	90.94%	78.57%	0.67	**0.89**	99.63%	21.04%	0.39

0.39	91.20%	77.87%	0.67	**0.9**	99.71%	18.76%	0.37
0.4	91.47%	77.32%	0.67	**0.91**	99.77%	16.62%	0.35
0.41	91.78%	76.86%	0.68	**0.92**	99.81%	14.45%	0.33
0.42	91.99%	76.29%	0.67	**0.93**	99.86%	12.52%	0.31
0.43	92.22%	75.83%	0.68	**0.94**	99.90%	10.66%	0.28
0.44	92.40%	75.15%	0.67	**0.95**	99.94%	8.91%	0.26
0.45	92.67%	74.54%	0.67	**0.96**	99.95%	7.25%	0.23
0.46	92.85%	73.99%	0.67	**0.97**	99.97%	5.58%	0.21
0.47	92.99%	73.42%	0.67	**0.98**	99.98%	3.96%	0.17
0.48	93.21%	72.77%	0.67	**0.99**	99.99%	1.88%	0.12
0.49	93.38%	72.22%	0.67	**1**	100.00%	0.00%	0.00
0.5	93.70%	71.41%	0.67				

(B) The experiment result of standard PSSM with –b option in SVM by five-fold cross-validation.

Threshold	**Spec.**	**Sens.**	**MCC**	**Threshold**	**Spec.**	**Sens.**	**MCC**
0	0.00%	100.00%	0.00	**0.51**	95.84%	43.91%	0.49
0.01	0.73%	100.00%	0.04	**0.52**	96.08%	43.06%	0.49
0.02	4.15%	99.65%	0.09	**0.53**	96.37%	42.16%	0.49
0.03	8.98%	99.08%	0.13	**0.54**	96.54%	40.94%	0.48
0.04	14.75%	98.29%	0.17	**0.55**	96.74%	40.11%	0.48
0.05	20.42%	97.31%	0.20	**0.56**	96.90%	39.43%	0.48
0.06	26.66%	95.93%	0.23	**0.57**	97.07%	38.59%	0.48
0.07	33.99%	94.48%	0.27	**0.58**	97.19%	37.63%	0.47
0.08	39.66%	92.93%	0.29	**0.59**	97.36%	36.91%	0.47
0.09	44.46%	91.51%	0.31	**0.6**	97.48%	36.21%	0.47
0.1	49.17%	90.30%	0.34	**0.61**	97.64%	35.25%	0.46
0.11	53.23%	88.68%	0.35	**0.62**	97.83%	34.30%	0.46
0.12	56.49%	87.22%	0.37	**0.63**	97.94%	33.60%	0.46
0.13	59.52%	86.08%	0.38	**0.64**	98.08%	32.82%	0.45
0.14	62.39%	84.41%	0.39	**0.65**	98.26%	31.59%	0.45
0.15	64.93%	83.10%	0.40	**0.66**	98.34%	30.91%	0.44
0.16	67.26%	81.66%	0.41	**0.67**	98.42%	30.06%	0.44
0.17	69.34%	80.43%	0.42	**0.68**	98.51%	29.07%	0.43
0.18	71.17%	78.59%	0.43	**0.69**	98.58%	28.04%	0.42
0.19	73.13%	77.30%	0.43	**0.7**	98.65%	27.04%	0.42
0.2	75.02%	76.03%	0.44	**0.71**	98.71%	26.36%	0.41
0.21	76.55%	74.58%	0.45	**0.72**	98.79%	25.48%	0.41
0.22	77.71%	73.47%	0.45	**0.73**	98.86%	24.54%	0.40
0.23	79.02%	72.29%	0.46	**0.74**	98.94%	23.82%	0.39
0.24	80.35%	71.28%	0.47	**0.75**	99.05%	23.14%	0.39
0.25	81.46%	70.16%	0.47	**0.76**	99.09%	22.53%	0.39
0.26	82.46%	69.11%	0.48	**0.77**	99.13%	21.63%	0.38
0.27	83.42%	67.78%	0.48	**0.78**	99.19%	20.86%	0.37
0.28	84.42%	66.97%	0.48	**0.79**	99.23%	20.14%	0.37
0.29	85.33%	65.78%	0.49	**0.8**	99.27%	19.40%	0.36

0.3	86.04%	64.65%	0.49	**0.81**	99.32%	18.43%	0.35
0.31	86.89%	63.42%	0.49	**0.82**	99.37%	17.64%	0.34
0.32	87.61%	62.26%	0.49	**0.83**	99.40%	16.97%	0.34
0.33	88.05%	61.62%	0.49	**0.84**	99.48%	16.00%	0.33
0.34	88.76%	60.55%	0.50	**0.85**	99.55%	15.11%	0.32
0.35	89.44%	59.52%	0.50	**0.86**	99.60%	14.19%	0.31
0.36	90.03%	58.38%	0.50	**0.87**	99.64%	13.05%	0.30
0.37	90.55%	57.27%	0.50	**0.88**	99.65%	12.13%	0.29
0.38	91.10%	56.30%	0.50	**0.89**	99.70%	10.75%	0.27
0.39	91.66%	55.34%	0.50	**0.9**	99.74%	9.52%	0.26
0.4	92.10%	54.33%	0.50	**0.91**	99.79%	8.12%	0.24
0.41	92.59%	53.06%	0.50	**0.92**	99.84%	7.05%	0.22
0.42	92.92%	52.21%	0.50	**0.93**	99.86%	6.22%	0.21
0.43	93.27%	51.34%	0.50	**0.94**	99.94%	5.30%	0.20
0.44	93.65%	50.35%	0.50	**0.95**	99.95%	4.29%	0.18
0.45	94.11%	49.43%	0.50	**0.96**	99.97%	3.46%	0.16
0.46	94.53%	48.27%	0.50	**0.97**	99.99%	2.50%	0.14
0.47	94.83%	47.48%	0.50	**0.98**	100.00%	1.44%	0.11
0.48	95.08%	46.56%	0.50	**0.99**	100.00%	0.46%	0.06
0.49	95.35%	45.84%	0.50	**1**	100.00%	0.00%	0.00
0.5	95.70%	44.44%	0.49				

(C) The experiment result of smoothed PSSM with –b option in SVM by three-way data split

Threshold	**Spec.**	**Sens.**	**MCC**	**Threshold**	**Spec.**	**Sens.**	**MCC**
0	0.00%	100.00%	0.00	**0.51**	93.44%	70.88%	0.66
0.01	6.91%	99.87%	0.13	**0.52**	93.58%	70.14%	0.66
0.02	19.24%	99.39%	0.22	**0.53**	93.72%	69.59%	0.66
0.03	32.53%	98.69%	0.30	**0.54**	93.88%	68.87%	0.65
0.04	45.75%	97.75%	0.38	**0.55**	94.07%	68.30%	0.65
0.05	55.62%	96.80%	0.44	**0.56**	94.28%	67.40%	0.65
0.06	63.25%	96.19%	0.50	**0.57**	94.45%	66.86%	0.65
0.07	69.07%	95.29%	0.54	**0.58**	94.64%	66.07%	0.65
0.08	72.24%	94.61%	0.57	**0.59**	94.81%	65.46%	0.65
0.09	74.63%	93.98%	0.58	**0.6**	94.99%	64.62%	0.64
0.1	76.52%	93.30%	0.60	**0.61**	95.17%	63.97%	0.64
0.11	77.96%	92.51%	0.61	**0.62**	95.39%	63.09%	0.64
0.12	79.00%	92.05%	0.62	**0.63**	95.60%	62.19%	0.64
0.13	79.93%	91.40%	0.62	**0.64**	95.77%	61.49%	0.64
0.14	80.77%	90.67%	0.63	**0.65**	95.99%	60.49%	0.63
0.15	81.58%	89.91%	0.63	**0.66**	96.19%	59.39%	0.63
0.16	82.31%	89.14%	0.63	**0.67**	96.35%	58.63%	0.63
0.17	83.00%	88.64%	0.64	**0.68**	96.52%	57.33%	0.62
0.18	83.62%	88.11%	0.64	**0.69**	96.69%	56.35%	0.62
0.19	84.15%	87.57%	0.64	**0.7**	96.90%	54.97%	0.61
0.2	84.65%	86.82%	0.64	**0.71**	97.08%	53.48%	0.60

0.21	85.10%	86.32%	0.65	**0.72**	97.26%	52.08%	0.60
0.22	85.56%	85.95%	0.65	**0.73**	97.44%	50.88%	0.59
0.23	85.92%	85.35%	0.65	**0.74**	97.65%	49.17%	0.58
0.24	86.36%	84.92%	0.65	**0.75**	97.83%	47.59%	0.58
0.25	86.80%	84.46%	0.66	**0.76**	98.02%	45.86%	0.57
0.26	87.18%	84.06%	0.66	**0.77**	98.18%	43.72%	0.55
0.27	87.45%	83.54%	0.66	**0.78**	98.32%	41.81%	0.54
0.28	87.81%	83.25%	0.66	**0.79**	98.49%	39.62%	0.53
0.29	88.09%	82.82%	0.66	**0.8**	98.65%	37.24%	0.51
0.3	88.36%	82.51%	0.66	**0.81**	98.79%	35.09%	0.50
0.31	88.69%	82.07%	0.67	**0.82**	98.96%	32.47%	0.48
0.32	88.92%	81.63%	0.67	**0.83**	99.06%	29.95%	0.46
0.33	89.14%	81.41%	0.67	**0.84**	99.19%	26.99%	0.44
0.34	89.44%	80.93%	0.67	**0.85**	99.37%	23.86%	0.41
0.35	89.75%	80.36%	0.67	**0.86**	99.46%	20.84%	0.38
0.36	90.01%	79.64%	0.67	**0.87**	99.60%	18.19%	0.36
0.37	90.29%	79.36%	0.67	**0.88**	99.70%	15.63%	0.34
0.38	90.58%	78.72%	0.67	**0.89**	99.76%	13.20%	0.31
0.39	90.80%	78.04%	0.67	**0.9**	99.85%	11.12%	0.29
0.4	91.05%	77.47%	0.67	**0.91**	99.88%	9.33%	0.26
0.41	91.30%	77.06%	0.67	**0.92**	99.92%	7.53%	0.24
0.42	91.58%	76.38%	0.67	**0.93**	99.95%	6.35%	0.22
0.43	91.73%	75.79%	0.67	**0.94**	99.96%	5.34%	0.20
0.44	91.92%	75.39%	0.67	**0.95**	99.98%	4.23%	0.18
0.45	92.10%	74.80%	0.67	**0.96**	99.99%	3.26%	0.16
0.46	92.38%	74.10%	0.66	**0.97**	99.99%	2.21%	0.13
0.47	92.60%	73.53%	0.66	**0.98**	99.99%	1.27%	0.10
0.48	92.76%	72.70%	0.66	**0.99**	100.00%	0.42%	0.06
0.49	92.95%	72.09%	0.66	**1**	100.00%	0.00%	0.00
0.5	93.30%	71.32%	0.66				

(D) The experiment result of standard PSSM with –b option in SVM by three-way data split

Threshold	**Spec.**	**Sens.**	**MCC**	**Threshold**	**Spec.**	**Sens.**	**MCC**
0	0.00%	100.00%	0.00	**0.51**	95.58%	42.27%	0.47
0.01	1.92%	99.87%	0.06	**0.52**	95.84%	41.53%	0.47
0.02	5.82%	99.50%	0.11	**0.53**	96.05%	40.89%	0.47
0.03	10.48%	99.01%	0.14	**0.54**	96.21%	40.15%	0.47
0.04	15.75%	98.29%	0.18	**0.55**	96.42%	39.14%	0.46
0.05	21.92%	97.09%	0.21	**0.56**	96.59%	38.46%	0.46
0.06	28.18%	95.91%	0.24	**0.57**	96.74%	37.48%	0.46
0.07	34.69%	94.81%	0.28	**0.58**	96.90%	36.67%	0.45
0.08	39.32%	93.41%	0.29	**0.59**	97.09%	36.01%	0.45
0.09	43.60%	92.40%	0.32	**0.6**	97.28%	35.49%	0.45
0.1	47.60%	91.24%	0.33	**0.61**	97.43%	34.92%	0.45
0.11	51.42%	89.75%	0.35	**0.62**	97.56%	34.24%	0.45

0.12	54.90%	88.35%	0.36	**0.63**	97.66%	33.49%	0.45
0.13	58.21%	86.87%	0.38	**0.64**	97.74%	32.92%	0.44
0.14	61.21%	85.31%	0.39	**0.65**	97.86%	31.79%	0.44
0.15	63.92%	84.04%	0.40	**0.66**	97.93%	31.20%	0.43
0.16	66.37%	82.53%	0.41	**0.67**	98.02%	30.67%	0.43
0.17	68.89%	81.22%	0.42	**0.68**	98.10%	29.71%	0.42
0.18	70.99%	79.84%	0.43	**0.69**	98.17%	28.98%	0.42
0.19	73.11%	78.31%	0.44	**0.7**	98.30%	28.31%	0.42
0.2	74.95%	76.66%	0.45	**0.71**	98.44%	27.36%	0.41
0.21	76.51%	74.98%	0.45	**0.72**	98.46%	26.69%	0.41
0.22	78.14%	73.66%	0.46	**0.73**	98.55%	26.03%	0.40
0.23	79.54%	72.37%	0.47	**0.74**	98.65%	25.44%	0.40
0.24	80.78%	71.06%	0.47	**0.75**	98.74%	24.72%	0.40
0.25	82.00%	69.68%	0.47	**0.76**	98.82%	24.10%	0.39
0.26	83.18%	68.83%	0.48	**0.77**	98.90%	23.38%	0.39
0.27	84.20%	67.38%	0.48	**0.78**	98.94%	22.57%	0.38
0.28	85.22%	66.22%	0.49	**0.79**	99.01%	21.89%	0.38
0.29	86.13%	64.75%	0.49	**0.8**	99.06%	21.23%	0.37
0.3	86.92%	63.46%	0.49	**0.81**	99.12%	20.56%	0.37
0.31	87.67%	62.30%	0.49	**0.82**	99.17%	19.81%	0.36
0.32	88.23%	61.01%	0.49	**0.83**	99.21%	19.18%	0.35
0.33	88.62%	60.29%	0.49	**0.84**	99.24%	18.45%	0.35
0.34	89.26%	58.80%	0.49	**0.85**	99.27%	17.86%	0.34
0.35	89.85%	57.51%	0.49	**0.86**	99.34%	17.14%	0.34
0.36	90.43%	56.48%	0.49	**0.87**	99.36%	16.42%	0.33
0.37	90.97%	55.54%	0.49	**0.88**	99.41%	15.59%	0.32
0.38	91.52%	54.38%	0.49	**0.89**	99.45%	14.71%	0.31
0.39	92.07%	53.74%	0.50	**0.9**	99.52%	14.05%	0.31
0.4	92.47%	52.50%	0.49	**0.91**	99.56%	13.31%	0.30
0.41	92.79%	51.44%	0.49	**0.92**	99.61%	12.52%	0.29
0.42	93.12%	50.46%	0.49	**0.93**	99.67%	11.49%	0.28
0.43	93.46%	49.47%	0.49	**0.94**	99.69%	10.25%	0.26
0.44	93.81%	48.42%	0.49	**0.95**	99.76%	8.65%	0.24
0.45	94.09%	47.22%	0.48	**0.96**	99.84%	6.81%	0.22
0.46	94.39%	46.50%	0.48	**0.97**	99.93%	4.84%	0.19
0.47	94.64%	45.73%	0.48	**0.98**	99.98%	3.02%	0.15
0.48	94.92%	44.86%	0.48	**0.99**	99.99%	1.34%	0.10
0.49	95.19%	43.94%	0.48	**1**	100.00%	0.00%	0.00
0.5	95.50%	42.54%	0.47				

Appendix 2.5

Experiment results of the RBP109

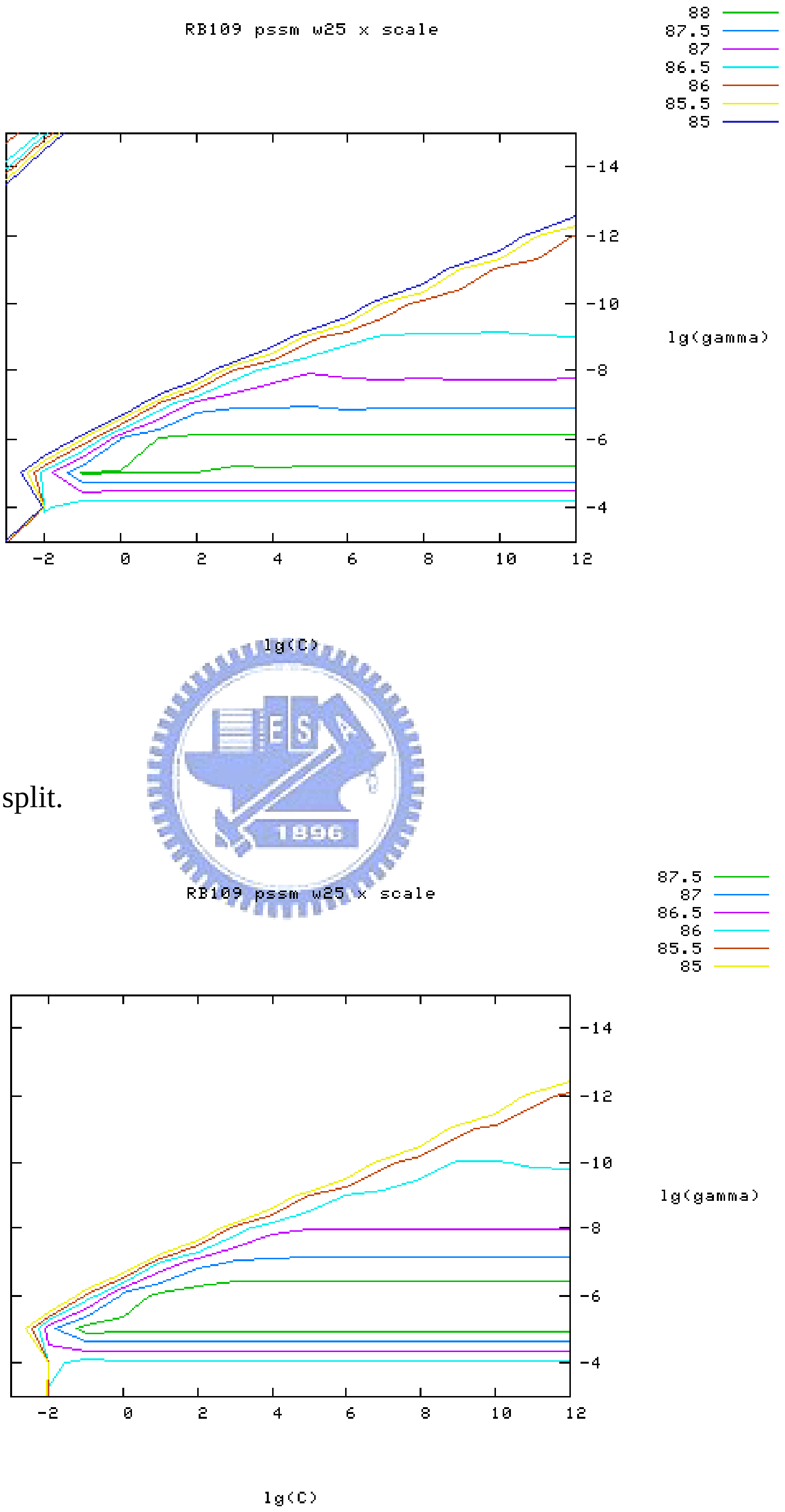
Table B 1. The detail performance of the RBP109 with different sliding window size under (A) five-fold cross-validation (w1 = 6.01, w-1 = 1, other parameters: default value) and (B) three-way data split (w1 = 6.01, w-1 = 1, other parameters: default value).

(A) Five-fold cross validation.

Window Size	Spec.	Sens.	MCC	Acc
3	74.99%	67.89%	0.32	73.98%
5	76.93%	68.22%	0.35	75.69%
7	77.72%	68.89%	0.36	76.46%
9	78.14%	69.73%	0.37	76.94%
11	78.77%	69.98%	0.38	77.52%
13	79.14%	69.42%	0.38	77.75%
15	79.62%	70.09%	0.39	78.27%
17	79.67%	70.15%	0.39	78.31%
19	79.73%	69.79%	0.39	78.31%
21	79.86%	69.95%	0.39	78.44%
23	80.21%	69.65%	0.39	78.71%
25	80.24%	69.51%	0.39	78.71%
27	80.08%	69.42%	0.39	78.56%
29	80.18%	69.20%	0.39	78.61%
31	80.53%	69.17%	0.39	78.91%
33	80.67%	68.92%	0.39	79.00%
35	80.53%	69.45%	0.40	78.95%
37	80.42%	69.56%	0.40	78.87%
39	80.47%	68.75%	0.39	78.80%
41	80.45%	68.84%	0.39	78.79%

(B) Three-way data split.

Window Size	Spec.	Sens.	MCC	Acc
3	75.22%	67.89%	0.33	74.17%
5	76.72%	68.44%	0.35	75.54%
7	77.82%	68.53%	0.36	76.50%
9	78.24%	69.14%	0.37	76.94%
11	78.73%	69.31%	0.37	77.38%
13	79.16%	69.39%	0.38	77.76%
15	79.38%	69.39%	0.38	77.95%
17	79.55%	69.09%	0.38	78.05%
19	79.54%	69.28%	0.38	78.07%
21	79.67%	68.75%	0.38	78.11%
23	80.09%	68.75%	0.39	78.47%
25	80.05%	69.06%	0.39	78.48%
27	79.87%	68.81%	0.38	78.29%
29	80.09%	68.33%	0.38	78.42%
31	80.12%	68.50%	0.38	78.46%
33	80.26%	68.36%	0.38	78.56%
35	80.13%	68.70%	0.39	78.50%
37	80.25%	69.48%	0.39	78.71%
39	80.34%	68.75%	0.39	78.69%
41	80.43%	68.47%	0.39	78.72%


(A) Five-fold cross-validation.

(B) Three-way data split.

Figure B 1. The performance with different combination of C and γ in the RBP109 data set under (A) five-fold cross-validation and (B) three-way data split.

Table B 2. The detail performance of the RBP109 with (A) different smoothing window size under five-fold cross-validation (w = 25, log C = 2, log γ = -6, w1 = 6.01, w-1 = 1, other parameters: default value) and (B) different smoothing window size under three-way data split (w = 25, log C = 3, log γ = -6, w1 = 6.01, w-1 = 1, other parameters: default value).

(A) Five-fold cross-validation.

Smoothing Window Size	**Spec.**	**Sens.**	**MCC**	**Acc**
1	97.32%	32.62%	0.41	88.09%
3	94.71%	54.82%	0.53	89.02%
5	93.31%	65.04%	0.57	89.28%
7	91.94%	70.37%	0.58	88.86%
9	90.62%	73.67%	0.58	88.20%
11	88.99%	75.96%	0.56	87.13%

(B) Three-way data split.

Smoothing Window Size	**Spec.**	**Sens.**	**MCC**	**Acc**
1	97.30%	30.80%	0.39	87.81%
3	95.53%	46.47%	0.48	88.53%
5	94.47%	57.27%	0.54	89.16%
7	93.56%	62.47%	0.56	89.13%
9	92.66%	65.88%	0.56	88.84%
11	91.53%	68.28%	0.56	88.21%

Table B 3. The detail performance of the RBP109 with (A) different weight parameter w1 under five-fold cross-validation (w = 25, log C = 2, log γ = -6, ws = 7, w-1 = 1, other parameters: default value) and (B) different weight parameter w1 under three-way data split (w = 25, log C = 3, log γ = -6, ws = 7, w-1 = 1, other parameters: default value).

(A) Five-fold cross-validation.

W1	Spec.	Sens.	MCC	Acc
1	97.63%	41.58%	0.51	89.63%
2	94.96%	58.84%	0.56	89.81%
3	93.45%	66.07%	0.58	89.54%
4	92.57%	68.72%	0.58	89.17%
5	92.19%	69.84%	0.58	89.00%
6	91.94%	70.37%	0.58	88.86%
7	91.71%	70.76%	0.58	88.72%
8	91.59%	70.96%	0.58	88.64%

(B) Three-way data split.

W1	Spec.	Sens.	MCC	Acc
1	96.46%	48.53%	0.53	89.62%
2	94.66%	59.09%	0.56	89.59%
3	93.98%	61.55%	0.56	89.36%
4	93.73%	62.22%	0.56	89.24%
5	93.63%	62.36%	0.56	89.17%
6	93.56%	62.47%	0.56	89.13%
7	93.57%	62.58%	0.56	89.15%
8	93.57%	62.58%	0.56	89.15%

Table B 4. The RBP109 data set experiment results with –b option in SVM for (A) smoothed PSSM by five-fold cross-validation, (B) standard PSSM by five-fold cross-validation, (C) smoothed PSSM by three-way data split, and (D) standard PSSM by three-way data split.

(A) The experiment result of smoothed PSSM with –b option in SVM by five-fold cross-validation.

Threshold	Spec.	Sens.	MCC	Threshold	Spec.	Sens.	MCC
0	0.00%	100.00%	0.00	**0.51**	96.58%	51.05%	0.55
0.01	13.58%	99.16%	0.14	**0.52**	96.70%	50.43%	0.55
0.02	29.55%	97.88%	0.22	**0.53**	96.79%	49.34%	0.54
0.03	42.20%	96.51%	0.28	**0.54**	96.91%	48.59%	0.54
0.04	52.11%	95.36%	0.33	**0.55**	97.07%	47.39%	0.54
0.05	59.37%	93.61%	0.37	**0.56**	97.20%	46.66%	0.53
0.06	65.68%	92.24%	0.41	**0.57**	97.36%	45.71%	0.53
0.07	70.75%	90.65%	0.44	**0.58**	97.47%	44.85%	0.53
0.08	74.08%	89.39%	0.47	**0.59**	97.57%	43.95%	0.52
0.09	76.63%	87.82%	0.48	**0.6**	97.68%	43.00%	0.52
0.1	78.88%	86.65%	0.50	**0.61**	97.75%	41.80%	0.51
0.11	80.84%	85.42%	0.51	**0.62**	97.82%	40.83%	0.51
0.12	82.20%	84.25%	0.52	**0.63**	97.91%	39.74%	0.50
0.13	83.51%	83.30%	0.53	**0.64**	98.01%	38.87%	0.50
0.14	84.55%	82.27%	0.54	**0.65**	98.14%	37.11%	0.49
0.15	85.63%	81.35%	0.55	**0.66**	98.26%	36.41%	0.49
0.16	86.47%	80.68%	0.56	**0.67**	98.34%	35.27%	0.48
0.17	87.14%	79.73%	0.56	**0.68**	98.41%	34.07%	0.47
0.18	87.73%	78.75%	0.56	**0.69**	98.53%	32.95%	0.47
0.19	88.33%	77.72%	0.57	**0.7**	98.66%	31.36%	0.46
0.2	88.79%	76.63%	0.57	**0.71**	98.74%	30.35%	0.45
0.21	89.31%	75.82%	0.57	**0.72**	98.83%	28.79%	0.44
0.22	89.82%	75.03%	0.57	**0.73**	98.95%	27.67%	0.43
0.23	90.31%	74.00%	0.57	**0.74**	98.99%	26.42%	0.42
0.24	90.76%	73.05%	0.58	**0.75**	99.08%	24.91%	0.41
0.25	91.14%	72.30%	0.58	**0.76**	99.15%	23.01%	0.39
0.26	91.52%	71.49%	0.58	**0.77**	99.20%	21.50%	0.38
0.27	91.86%	70.57%	0.58	**0.78**	99.25%	19.99%	0.37
0.28	92.14%	69.84%	0.58	**0.79**	99.29%	18.82%	0.36
0.29	92.43%	69.09%	0.58	**0.8**	99.35%	17.56%	0.34
0.3	92.69%	68.39%	0.58	**0.81**	99.41%	16.03%	0.33
0.31	92.93%	67.75%	0.58	**0.82**	99.47%	14.94%	0.32
0.32	93.23%	67.02%	0.58	**0.83**	99.52%	13.35%	0.30
0.33	93.36%	66.43%	0.58	**0.84**	99.58%	12.20%	0.29
0.34	93.67%	65.43%	0.58	**0.85**	99.64%	10.89%	0.27
0.35	93.88%	64.62%	0.58	**0.86**	99.69%	9.86%	0.26
0.36	94.08%	63.75%	0.58	**0.87**	99.74%	8.99%	0.25
0.37	94.34%	63.03%	0.58	**0.88**	99.80%	7.82%	0.24
0.38	94.56%	62.27%	0.58	**0.89**	99.83%	6.98%	0.22

0.39	94.75%	61.63%	0.58	**0.9**	99.85%	5.81%	0.20
0.4	94.94%	60.96%	0.58	**0.91**	99.87%	4.86%	0.19
0.41	95.13%	60.09%	0.58	**0.92**	99.90%	4.08%	0.17
0.42	95.31%	59.23%	0.58	**0.93**	99.91%	3.55%	0.16
0.43	95.50%	58.59%	0.58	**0.94**	99.94%	2.60%	0.14
0.44	95.65%	57.61%	0.57	**0.95**	99.95%	1.84%	0.11
0.45	95.82%	56.44%	0.57	**0.96**	99.98%	1.37%	0.10
0.46	95.96%	55.52%	0.57	**0.97**	99.99%	0.84%	0.08
0.47	96.08%	54.59%	0.56	**0.98**	100.00%	0.34%	0.05
0.48	96.20%	53.62%	0.56	**0.99**	100.00%	0.11%	0.03
0.49	96.34%	52.83%	0.56	**1**	100.00%	0.00%	0.00
0.5	96.52%	51.49%	0.55				

(B) The experiment result of standard PSSM with –b option in SVM by five-fold cross-validation.

Threshold	**Spec.**	**Sens.**	**MCC**	**Threshold**	**Spec.**	**Sens.**	**MCC**
0	0.00%	100.00%	0.00	**0.51**	98.00%	28.20%	0.39
0.01	2.66%	99.80%	0.06	**0.52**	98.08%	27.42%	0.39
0.02	10.49%	98.63%	0.11	**0.53**	98.20%	26.84%	0.39
0.03	19.59%	96.65%	0.15	**0.54**	98.31%	26.19%	0.39
0.04	28.65%	94.69%	0.19	**0.55**	98.41%	25.55%	0.38
0.05	36.92%	92.52%	0.22	**0.56**	98.49%	24.57%	0.38
0.06	44.75%	90.51%	0.25	**0.57**	98.56%	23.93%	0.37
0.07	52.95%	87.88%	0.29	**0.58**	98.61%	23.37%	0.37
0.08	58.54%	85.70%	0.31	**0.59**	98.69%	22.95%	0.37
0.09	62.91%	83.55%	0.33	**0.6**	98.74%	22.54%	0.37
0.1	66.68%	81.32%	0.34	**0.61**	98.84%	22.06%	0.37
0.11	69.92%	79.14%	0.36	**0.62**	98.89%	21.28%	0.36
0.12	72.72%	76.60%	0.36	**0.63**	98.94%	20.75%	0.36
0.13	75.16%	74.70%	0.37	**0.64**	98.99%	20.08%	0.35
0.14	77.37%	72.69%	0.38	**0.65**	99.10%	19.27%	0.35
0.15	79.16%	71.32%	0.39	**0.66**	99.18%	18.68%	0.35
0.16	80.96%	69.03%	0.40	**0.67**	99.22%	17.87%	0.34
0.17	82.41%	67.61%	0.41	**0.68**	99.26%	17.20%	0.33
0.18	83.96%	66.13%	0.42	**0.69**	99.32%	16.56%	0.33
0.19	85.16%	64.28%	0.42	**0.7**	99.35%	15.97%	0.32
0.2	86.37%	62.83%	0.42	**0.71**	99.41%	15.44%	0.32
0.21	87.43%	61.13%	0.43	**0.72**	99.46%	15.02%	0.32
0.22	88.26%	59.45%	0.43	**0.73**	99.50%	14.33%	0.31
0.23	89.10%	57.69%	0.43	**0.74**	99.54%	13.80%	0.31
0.24	89.89%	56.46%	0.43	**0.75**	99.55%	13.07%	0.30
0.25	90.57%	55.07%	0.44	**0.76**	99.57%	12.59%	0.29
0.26	91.24%	53.70%	0.44	**0.77**	99.60%	12.15%	0.29
0.27	91.88%	52.33%	0.44	**0.78**	99.63%	11.51%	0.28
0.28	92.42%	50.99%	0.44	**0.79**	99.67%	11.09%	0.28
0.29	92.90%	50.07%	0.44	**0.8**	99.69%	10.58%	0.27

0.3	93.38%	48.90%	0.44	**0.81**	99.73%	10.16%	0.27
0.31	93.74%	48.12%	0.45	**0.82**	99.74%	9.69%	0.26
0.32	94.11%	46.89%	0.45	**0.83**	99.75%	9.16%	0.25
0.33	94.37%	45.96%	0.44	**0.84**	99.78%	8.63%	0.25
0.34	94.74%	44.76%	0.44	**0.85**	99.81%	7.90%	0.24
0.35	95.03%	43.42%	0.44	**0.86**	99.86%	7.34%	0.23
0.36	95.29%	42.11%	0.44	**0.87**	99.88%	6.65%	0.22
0.37	95.51%	40.80%	0.43	**0.88**	99.90%	6.03%	0.21
0.38	95.76%	39.77%	0.43	**0.89**	99.92%	5.17%	0.20
0.39	96.04%	38.73%	0.43	**0.9**	99.93%	4.66%	0.19
0.4	96.29%	37.70%	0.42	**0.91**	99.94%	4.19%	0.18
0.41	96.52%	36.41%	0.42	**0.92**	99.96%	2.68%	0.14
0.42	96.71%	35.41%	0.42	**0.93**	99.98%	2.01%	0.13
0.43	96.91%	34.54%	0.42	**0.94**	99.99%	1.51%	0.11
0.44	97.09%	33.73%	0.41	**0.95**	99.99%	1.23%	0.10
0.45	97.25%	33.04%	0.41	**0.96**	99.99%	0.98%	0.09
0.46	97.35%	32.20%	0.41	**0.97**	100.00%	0.70%	0.08
0.47	97.48%	31.50%	0.41	**0.98**	100.00%	0.31%	0.05
0.48	97.62%	30.58%	0.40	**0.99**	100.00%	0.03%	0.02
0.49	97.75%	29.80%	0.40	**1**	100.00%	0.00%	0.00
0.5	97.95%	28.71%	0.40				

(C) The experiment result of smoothed PSSM with –b option in SVM by three-way data split.

Threshold	**Spec.**	**Sens.**	**MCC**	**Threshold**	**Spec.**	**Sens.**	**MCC**
0	0.00%	100.00%	0.00	**0.51**	96.72%	46.10%	0.51
0.01	9.61%	99.72%	0.12	**0.52**	96.82%	45.38%	0.51
0.02	22.97%	98.49%	0.19	**0.53**	96.96%	44.68%	0.51
0.03	34.95%	97.01%	0.24	**0.54**	97.07%	43.84%	0.51
0.04	44.47%	95.48%	0.29	**0.55**	97.24%	42.81%	0.50
0.05	52.20%	93.83%	0.32	**0.56**	97.32%	42.06%	0.50
0.06	58.64%	92.40%	0.36	**0.57**	97.43%	41.30%	0.50
0.07	65.30%	90.48%	0.39	**0.58**	97.58%	40.32%	0.49
0.08	69.44%	89.03%	0.42	**0.59**	97.69%	39.15%	0.49
0.09	72.87%	87.49%	0.44	**0.6**	97.76%	38.26%	0.48
0.1	75.75%	85.95%	0.46	**0.61**	97.86%	37.36%	0.48
0.11	78.11%	84.86%	0.48	**0.62**	97.98%	36.30%	0.47
0.12	80.21%	83.55%	0.49	**0.63**	98.07%	35.33%	0.47
0.13	81.86%	82.44%	0.50	**0.64**	98.16%	34.18%	0.46
0.14	83.20%	81.15%	0.51	**0.65**	98.33%	32.73%	0.45
0.15	84.42%	79.78%	0.52	**0.66**	98.40%	31.75%	0.45
0.16	85.44%	78.58%	0.53	**0.67**	98.48%	30.44%	0.44
0.17	86.34%	77.41%	0.53	**0.68**	98.54%	29.43%	0.43
0.18	87.16%	76.68%	0.54	**0.69**	98.64%	28.09%	0.42
0.19	87.82%	75.31%	0.54	**0.7**	98.70%	27.12%	0.42
0.2	88.43%	74.25%	0.54	**0.71**	98.76%	25.94%	0.41

0.21	89.00%	73.22%	0.55	**0.72**	98.84%	25.13%	0.40
0.22	89.56%	72.27%	0.55	**0.73**	98.96%	23.93%	0.39
0.23	90.13%	71.32%	0.55	**0.74**	99.05%	22.76%	0.39
0.24	90.53%	70.43%	0.55	**0.75**	99.13%	21.47%	0.38
0.25	90.92%	69.76%	0.56	**0.76**	99.24%	20.36%	0.37
0.26	91.39%	68.61%	0.56	**0.77**	99.33%	19.18%	0.36
0.27	91.81%	67.50%	0.56	**0.78**	99.41%	17.76%	0.35
0.28	92.10%	66.85%	0.56	**0.79**	99.45%	16.28%	0.33
0.29	92.46%	66.07%	0.56	**0.8**	99.51%	15.22%	0.32
0.3	92.83%	65.15%	0.56	**0.81**	99.58%	14.16%	0.32
0.31	93.11%	64.09%	0.56	**0.82**	99.62%	12.85%	0.30
0.32	93.45%	63.22%	0.56	**0.83**	99.65%	11.59%	0.28
0.33	93.62%	62.55%	0.56	**0.84**	99.71%	10.56%	0.27
0.34	93.87%	61.71%	0.56	**0.85**	99.74%	9.63%	0.26
0.35	94.14%	60.63%	0.56	**0.86**	99.78%	8.49%	0.25
0.36	94.36%	59.76%	0.56	**0.87**	99.81%	7.48%	0.23
0.37	94.56%	58.73%	0.55	**0.88**	99.84%	6.42%	0.21
0.38	94.73%	58.06%	0.55	**0.89**	99.85%	5.67%	0.20
0.39	94.93%	57.00%	0.55	**0.9**	99.87%	4.55%	0.18
0.4	95.10%	56.35%	0.55	**0.91**	99.92%	3.60%	0.16
0.41	95.28%	55.32%	0.55	**0.92**	99.94%	2.76%	0.14
0.42	95.46%	54.54%	0.54	**0.93**	99.95%	1.98%	0.12
0.43	95.60%	53.76%	0.54	**0.94**	99.97%	1.54%	0.11
0.44	95.75%	52.86%	0.54	**0.95**	99.97%	1.09%	0.09
0.45	95.94%	52.00%	0.54	**0.96**	99.97%	0.64%	0.06
0.46	96.12%	50.91%	0.53	**0.97**	99.98%	0.39%	0.05
0.47	96.25%	49.85%	0.53	**0.98**	99.99%	0.28%	0.04
0.48	96.36%	48.81%	0.52	**0.99**	100.00%	0.11%	0.03
0.49	96.48%	47.81%	0.52	**1**	100.00%	0.00%	0.00
0.5	96.66%	46.58%	0.52				

(D) The experiment result of standard PSSM with –b option in SVM by three-way data split.

Threshold	Spec.	Sens.	MCC	Threshold	Spec.	Sens.	MCC
0	0.00%	100.00%	0.00	**0.51**	98.10%	26.81%	0.38
0.01	2.53%	99.89%	0.06	**0.52**	98.24%	25.80%	0.38
0.02	9.94%	99.05%	0.11	**0.53**	98.32%	24.94%	0.37
0.03	18.61%	97.18%	0.15	**0.54**	98.41%	24.24%	0.37
0.04	27.52%	95.09%	0.18	**0.55**	98.49%	23.65%	0.37
0.05	35.52%	92.96%	0.21	**0.56**	98.55%	23.07%	0.36
0.06	42.99%	90.92%	0.24	**0.57**	98.63%	22.26%	0.36
0.07	50.87%	88.19%	0.27	**0.58**	98.69%	21.59%	0.35
0.08	56.31%	85.79%	0.29	**0.59**	98.79%	20.66%	0.35
0.09	60.97%	83.13%	0.31	**0.6**	98.83%	20.11%	0.34
0.1	65.08%	80.54%	0.32	**0.61**	98.89%	19.58%	0.34

0.11	68.78%	78.02%	0.34	**0.62**	98.97%	18.96%	0.34
0.12	71.88%	75.87%	0.35	**0.63**	99.02%	18.35%	0.33
0.13	74.40%	73.95%	0.36	**0.64**	99.08%	17.70%	0.33
0.14	76.78%	72.02%	0.37	**0.65**	99.17%	16.95%	0.32
0.15	78.70%	70.06%	0.38	**0.66**	99.24%	16.48%	0.32
0.16	80.60%	68.25%	0.39	**0.67**	99.27%	15.75%	0.31
0.17	82.10%	66.60%	0.39	**0.68**	99.34%	15.25%	0.31
0.18	83.56%	64.73%	0.40	**0.69**	99.37%	14.69%	0.31
0.19	84.84%	63.25%	0.41	**0.7**	99.42%	14.24%	0.30
0.2	86.06%	61.32%	0.41	**0.71**	99.44%	13.52%	0.30
0.21	87.09%	59.73%	0.41	**0.72**	99.48%	12.96%	0.29
0.22	88.01%	58.42%	0.42	**0.73**	99.51%	12.20%	0.28
0.23	88.77%	56.80%	0.42	**0.74**	99.55%	11.64%	0.28
0.24	89.59%	55.38%	0.42	**0.75**	99.59%	11.11%	0.27
0.25	90.29%	53.98%	0.42	**0.76**	99.62%	10.42%	0.26
0.26	90.99%	52.42%	0.42	**0.77**	99.64%	9.77%	0.25
0.27	91.59%	51.33%	0.43	**0.78**	99.66%	9.49%	0.25
0.28	92.10%	49.93%	0.42	**0.79**	99.71%	8.96%	0.25
0.29	92.67%	48.81%	0.43	**0.8**	99.72%	8.43%	0.24
0.3	93.08%	47.78%	0.43	**0.81**	99.76%	7.93%	0.23
0.31	93.49%	46.58%	0.43	**0.82**	99.78%	7.48%	0.23
0.32	93.90%	45.16%	0.43	**0.83**	99.81%	7.07%	0.22
0.33	94.11%	44.48%	0.42	**0.84**	99.85%	6.42%	0.22
0.34	94.53%	43.54%	0.43	**0.85**	99.88%	6.00%	0.21
0.35	94.84%	42.22%	0.42	**0.86**	99.91%	5.45%	0.20
0.36	95.22%	41.30%	0.43	**0.87**	99.92%	5.03%	0.20
0.37	95.51%	40.13%	0.42	**0.88**	99.94%	4.64%	0.19
0.38	95.83%	39.35%	0.43	**0.89**	99.94%	4.22%	0.18
0.39	96.03%	38.23%	0.42	**0.9**	99.95%	3.71%	0.17
0.4	96.29%	37.11%	0.42	**0.91**	99.97%	3.38%	0.16
0.41	96.52%	35.94%	0.42	**0.92**	99.98%	2.23%	0.13
0.42	96.72%	34.99%	0.41	**0.93**	100.00%	1.56%	0.11
0.43	96.91%	34.29%	0.41	**0.94**	100.00%	1.20%	0.10
0.44	97.10%	33.04%	0.41	**0.95**	100.00%	0.92%	0.09
0.45	97.26%	32.23%	0.41	**0.96**	100.00%	0.73%	0.08
0.46	97.44%	31.25%	0.40	**0.97**	100.00%	0.50%	0.07
0.47	97.59%	30.27%	0.40	**0.98**	100.00%	0.17%	0.04
0.48	97.75%	29.29%	0.39	**0.99**	100.00%	0.03%	0.02
0.49	97.87%	28.57%	0.39	**1**	100.00%	0.00%	0.00
0.5	98.05%	27.23%	0.39				

Appendix 2.6

Experiment results of the RBP107

Table C 1. The detail performance of the RBP107 with different sliding window size under (A) five-fold cross-validation (w1 = 7.63, w-1 = 1, other parameters: default value) and three-way data split (w1 = 7.63, w-1 = 1, other parameters: default value).

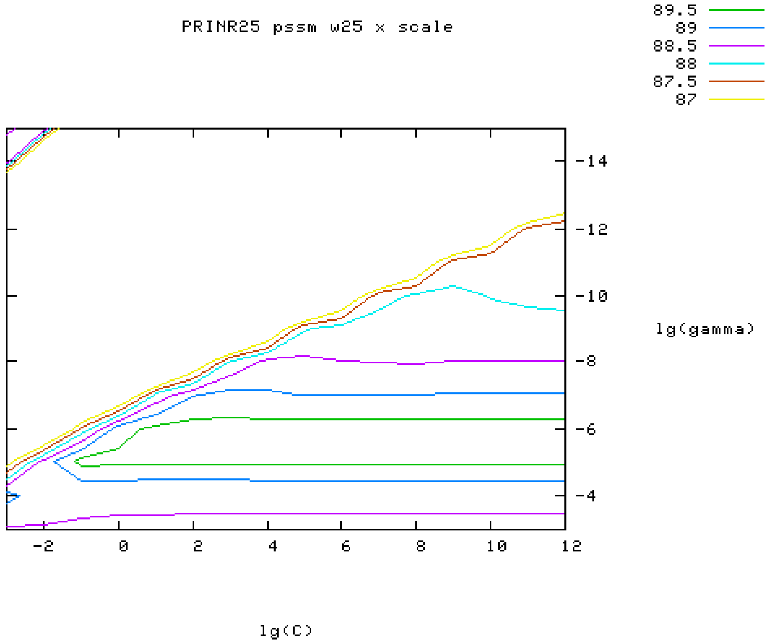
(A) Five-fold cross-validation.

Window Size	Spec.	Sens.	MCC	Acc
3	75.28%	70.57%	0.32	74.73%
5	76.77%	70.96%	0.34	76.10%
7	78.08%	71.90%	0.36	77.37%
9	78.58%	72.41%	0.37	77.86%
11	79.42%	71.51%	0.37	78.50%
13	79.65%	71.39%	0.37	78.69%
15	80.09%	71.51%	0.38	79.10%
17	80.17%	71.35%	0.38	79.15%
19	80.43%	71.12%	0.38	79.35%
21	80.60%	70.96%	0.38	79.48%
23	80.95%	71.55%	0.39	79.86%
25	80.83%	71.19%	0.38	79.72%
27	80.80%	71.12%	0.38	79.68%
29	80.87%	71.08%	0.38	79.74%
31	80.95%	71.39%	0.39	79.84%
33	81.05%	71.00%	0.38	79.89%
35	81.08%	71.00%	0.39	79.91%
37	81.07%	70.68%	0.38	79.86%
39	80.99%	71.04%	0.38	79.84%
41	81.07%	69.98%	0.38	79.78%

(B) Three-way data split.

Window Size	Spec.	Sens.	MCC	Acc
3	75.18%	69.98%	0.32	74.58%
5	76.75%	70.65%	0.34	76.05%
7	78.18%	70.96%	0.35	77.34%
9	78.53%	71.62%	0.36	77.73%
11	79.32%	71.23%	0.37	78.38%
13	79.49%	71.12%	0.37	78.52%
15	79.93%	71.12%	0.37	78.91%
17	80.14%	70.88%	0.37	79.07%
19	80.46%	70.57%	0.37	79.31%
21	80.34%	70.22%	0.37	79.17%
23	80.96%	70.06%	0.38	79.70%
25	81.04%	70.53%	0.38	79.82%
27	81.24%	70.49%	0.38	79.99%
29	81.13%	69.94%	0.38	79.83%
31	81.18%	69.90%	0.38	79.87%
33	81.21%	69.90%	0.38	79.90%
35	81.27%	69.47%	0.38	79.90%
37	81.27%	69.43%	0.38	79.90%
39	81.37%	69.28%	0.38	79.96%
41	81.36%	69.47%	0.38	79.98%

(A) Five-fold cross-validation.



(B) Three-way data split.

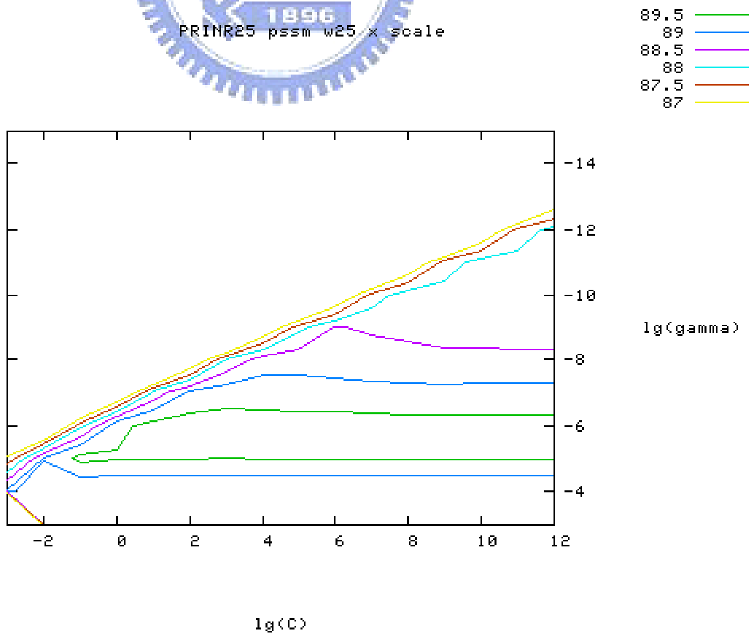


Figure C 1. The performance with different combination of C and γ in the RBP107 data set under (A) five-fold cross-validation and (B) three-way data split.

Table C 2. The detail performance of the RBP107 with (A) different smoothing window size under five-fold cross-validation (w = 25, log C = 2, log γ = -6, w1 = 7.63, w-1 = 1, other parameters: default value) and (B) different smoothing window size under three-way data split (w = 25, log C = 3, log γ = -6, w1 = 7.63, w-1 = 1, other parameters: default value).

(A) Five-fold cross-validation.

Smoothing Window Size	Spec.	Sens.	MCC	Acc
1	97.80%	27.83%	0.37	89.69%
3	94.91%	45.21%	0.43	89.15%
5	93.09%	54.64%	0.46	88.63%
7	91.44%	59.65%	0.46	87.76%
9	90.22%	63.29%	0.47	87.10%
11	88.85%	65.83%	0.46	86.18%

(B) Three-way data split.

Smoothing Window Size	Spec.	Sens.	MCC	Acc
1	98.04%	26.18%	0.36	89.71%
3	95.91%	38.20%	0.40	89.23%
5	94.64%	45.95%	0.43	89.00%
7	93.51%	49.98%	0.44	88.47%
9	92.68%	53.86%	0.45	88.18%
11	91.60%	57.10%	0.45	87.61%

Table C 3. The detail performance of the RBP107 with (A) different weight parameter w1 under five-fold cross-validation (w = 25, log C = 2, log γ = -6, ws = 7, w-1 = 1, other parameters: default value) and (B) different weight parameter w1 under three-way data split (w = 25, log C = 3, log γ = -6, ws = 7, w-1 = 1, other parameters: default value).

(A) Five-fold cross-validation.

W1	Spec.	Sens.	MCC	Acc
1	97.81%	29.75%	0.39	89.93%
2	94.86%	48.77%	0.46	89.52%
3	93.27%	55.54%	0.47	88.90%
4	92.29%	58.47%	0.47	88.37%
5	91.79%	58.87%	0.47	87.98%
6	91.59%	59.33%	0.47	87.86%
7	91.48%	59.49%	0.46	87.77%
8	91.44%	59.69%	0.47	87.76%
9	91.43%	59.69%	0.46	87.75%
10	91.42%	59.73%	0.46	87.75%

(B) Three-way data split.

W1	Spec.	Sens.	MCC	Acc
1	96.54%	36.75%	0.41	89.62%
2	94.28%	47.40%	0.43	88.85%
3	93.79%	49.39%	0.44	88.64%
4	93.59%	49.86%	0.44	88.53%
5	93.53%	50.06%	0.44	88.49%
6	93.51%	49.98%	0.44	88.46%
7	93.51%	49.98%	0.44	88.47%
8	93.51%	49.98%	0.44	88.47%
9	93.51%	49.98%	0.44	88.47%
10	93.51%	49.98%	0.44	88.47%

Table C 4. The RBP107 data set experiment results with –b option in SVM for (A) smoothed PSSM by five-fold cross-validation, (B) standard PSSM by five-fold cross-validation, (C) smoothed PSSM by three-way data split, and (D) standard PSSM by three-way data split.

(A) The experiment result of smoothed PSSM with –b option in SVM by five-fold cross-validation.

Threshold	Spec.	Sens.	MCC	Threshold	Spec.	Sens.	MCC
0	0.00%	100.00%	0.00	**0.51**	97.47%	32.72%	0.41
0.01	7.60%	99.61%	0.09	**0.52**	97.60%	31.86%	0.40
0.02	21.19%	98.24%	0.16	**0.53**	97.72%	30.22%	0.39
0.03	34.93%	96.09%	0.21	**0.54**	97.80%	28.65%	0.38
0.04	46.04%	93.31%	0.26	**0.55**	97.90%	27.75%	0.37
0.05	54.70%	90.53%	0.29	**0.56**	98.02%	26.81%	0.37
0.06	61.88%	88.49%	0.32	**0.57**	98.14%	25.60%	0.36
0.07	68.56%	85.68%	0.36	**0.58**	98.26%	24.31%	0.35
0.08	72.79%	83.37%	0.38	**0.59**	98.35%	23.68%	0.35
0.09	76.38%	81.10%	0.40	**0.6**	98.41%	22.54%	0.34
0.1	78.88%	79.14%	0.41	**0.61**	98.53%	21.33%	0.33
0.11	80.87%	77.14%	0.42	**0.62**	98.60%	20.27%	0.32
0.12	82.65%	74.95%	0.43	**0.63**	98.70%	19.37%	0.32
0.13	83.94%	73.58%	0.44	**0.64**	98.79%	18.24%	0.31
0.14	85.10%	72.56%	0.45	**0.65**	98.90%	17.38%	0.31
0.15	86.07%	71.23%	0.45	**0.66**	98.97%	16.40%	0.30
0.16	86.93%	69.43%	0.45	**0.67**	99.04%	15.50%	0.29
0.17	87.71%	67.83%	0.46	**0.68**	99.12%	14.40%	0.28
0.18	88.37%	66.54%	0.46	**0.69**	99.22%	13.50%	0.27
0.19	89.06%	65.44%	0.46	**0.7**	99.31%	12.76%	0.27
0.2	89.66%	64.66%	0.47	**0.71**	99.37%	12.09%	0.27
0.21	90.06%	63.52%	0.47	**0.72**	99.44%	10.80%	0.25
0.22	90.52%	62.50%	0.47	**0.73**	99.50%	9.90%	0.24
0.23	90.94%	61.33%	0.47	**0.74**	99.57%	8.88%	0.23
0.24	91.36%	60.31%	0.47	**0.75**	99.60%	8.14%	0.22
0.25	91.72%	59.84%	0.47	**0.76**	99.64%	7.08%	0.20
0.26	92.13%	58.83%	0.47	**0.77**	99.69%	6.26%	0.19
0.27	92.50%	57.69%	0.47	**0.78**	99.73%	5.28%	0.17
0.28	92.81%	56.44%	0.47	**0.79**	99.74%	4.85%	0.17
0.29	93.12%	55.23%	0.47	**0.8**	99.78%	4.27%	0.16
0.3	93.43%	53.93%	0.47	**0.81**	99.80%	3.84%	0.15
0.31	93.77%	52.96%	0.47	**0.82**	99.85%	3.21%	0.14
0.32	94.05%	51.86%	0.46	**0.83**	99.87%	2.97%	0.13
0.33	94.21%	51.15%	0.46	**0.84**	99.88%	2.58%	0.12
0.34	94.49%	50.22%	0.46	**0.85**	99.90%	2.19%	0.12
0.35	94.68%	49.28%	0.46	**0.86**	99.92%	1.76%	0.10
0.36	94.90%	48.45%	0.46	**0.87**	99.94%	1.45%	0.09
0.37	95.16%	47.16%	0.46	**0.88**	99.94%	1.33%	0.09
0.38	95.36%	45.95%	0.45	**0.89**	99.96%	0.98%	0.08

0.39	95.55%	44.97%	0.45	**0.9**	99.97%	0.74%	0.07
0.4	95.73%	43.68%	0.44	**0.91**	99.97%	0.43%	0.05
0.41	95.92%	42.86%	0.44	**0.92**	99.98%	0.31%	0.04
0.42	96.11%	42.00%	0.44	**0.93**	99.98%	0.27%	0.04
0.43	96.30%	40.94%	0.44	**0.94**	100.00%	0.20%	0.04
0.44	96.41%	40.00%	0.43	**0.95**	100.00%	0.04%	0.02
0.45	96.59%	38.71%	0.43	**0.96**	100.00%	0.00%	0.00
0.46	96.74%	37.57%	0.42	**0.97**	100.00%	0.00%	0.00
0.47	96.90%	36.83%	0.42	**0.98**	100.00%	0.00%	0.00
0.48	97.06%	35.73%	0.42	**0.99**	100.00%	0.00%	0.00
0.49	97.24%	34.32%	0.41	**1**	100.00%	0.00%	0.00
0.5	97.45%	33.15%	0.41				

(B) The experiment result of standard PSSM with –b option in SVM by five-fold cross-validation.

Threshold	**Spec.**	**Sens.**	**MCC**	**Threshold**	**Spec.**	**Sens.**	**MCC**
0	0.00%	100.00%	0.00	**0.51**	98.52%	21.84%	0.34
0.01	3.03%	99.61%	0.05	**0.52**	98.56%	21.17%	0.33
0.02	12.75%	98.63%	0.11	**0.53**	98.60%	20.67%	0.33
0.03	23.83%	96.48%	0.16	**0.54**	98.66%	20.00%	0.32
0.04	34.14%	93.66%	0.19	**0.55**	98.71%	19.49%	0.32
0.05	43.04%	90.57%	0.22	**0.56**	98.74%	18.83%	0.31
0.06	51.69%	87.63%	0.25	**0.57**	98.79%	18.20%	0.31
0.07	59.83%	83.87%	0.28	**0.58**	98.86%	17.57%	0.31
0.08	64.93%	80.82%	0.30	**0.59**	98.92%	16.83%	0.30
0.09	69.40%	77.65%	0.31	**0.6**	98.97%	16.01%	0.29
0.1	73.06%	75.19%	0.33	**0.61**	99.03%	15.69%	0.29
0.11	76.10%	73.03%	0.34	**0.62**	99.07%	14.95%	0.28
0.12	78.74%	70.61%	0.36	**0.63**	99.11%	14.44%	0.28
0.13	80.82%	68.53%	0.37	**0.64**	99.15%	13.86%	0.27
0.14	82.69%	66.18%	0.37	**0.65**	99.22%	13.11%	0.27
0.15	84.29%	63.84%	0.38	**0.66**	99.26%	12.29%	0.26
0.16	85.71%	61.80%	0.38	**0.67**	99.28%	11.59%	0.25
0.17	86.95%	59.69%	0.38	**0.68**	99.33%	11.08%	0.25
0.18	88.16%	57.85%	0.39	**0.69**	99.36%	10.88%	0.24
0.19	89.22%	55.93%	0.39	**0.7**	99.39%	10.06%	0.23
0.2	90.06%	54.21%	0.40	**0.71**	99.41%	9.43%	0.22
0.21	90.86%	52.80%	0.40	**0.72**	99.45%	8.73%	0.22
0.22	91.66%	51.74%	0.41	**0.73**	99.47%	8.34%	0.21
0.23	92.27%	49.67%	0.40	**0.74**	99.48%	7.98%	0.20
0.24	92.85%	48.10%	0.40	**0.75**	99.50%	7.36%	0.19
0.25	93.38%	46.54%	0.40	**0.76**	99.54%	7.08%	0.19
0.26	93.81%	45.05%	0.40	**0.77**	99.56%	6.65%	0.19
0.27	94.17%	43.87%	0.40	**0.78**	99.60%	6.34%	0.18
0.28	94.53%	42.54%	0.40	**0.79**	99.63%	5.99%	0.18
0.29	94.84%	41.68%	0.40	**0.8**	99.65%	5.48%	0.17

0.3	95.17%	40.39%	0.40	**0.81**	99.67%	4.85%	0.16
0.31	95.45%	38.75%	0.39	**0.82**	99.69%	4.62%	0.15
0.32	95.72%	37.73%	0.39	**0.83**	99.72%	4.15%	0.15
0.33	95.89%	37.14%	0.39	**0.84**	99.76%	3.76%	0.14
0.34	96.09%	35.73%	0.38	**0.85**	99.78%	3.37%	0.13
0.35	96.38%	34.76%	0.38	**0.86**	99.82%	3.05%	0.13
0.36	96.63%	33.86%	0.38	**0.87**	99.86%	2.47%	0.12
0.37	96.86%	32.84%	0.38	**0.88**	99.89%	2.11%	0.11
0.38	97.07%	32.05%	0.38	**0.89**	99.92%	1.80%	0.10
0.39	97.26%	30.61%	0.38	**0.9**	99.94%	1.29%	0.09
0.4	97.43%	29.98%	0.38	**0.91**	99.97%	0.90%	0.08
0.41	97.54%	29.47%	0.38	**0.92**	99.98%	0.74%	0.07
0.42	97.66%	28.57%	0.37	**0.93**	99.99%	0.63%	0.07
0.43	97.76%	27.87%	0.37	**0.94**	99.99%	0.51%	0.06
0.44	97.91%	26.97%	0.37	**0.95**	100.00%	0.27%	0.05
0.45	98.03%	26.18%	0.36	**0.96**	100.00%	0.16%	0.04
0.46	98.12%	25.44%	0.36	**0.97**	100.00%	0.04%	0.02
0.47	98.19%	24.62%	0.35	**0.98**	100.00%	0.00%	0.00
0.48	98.28%	24.03%	0.35	**0.99**	100.00%	0.00%	0.00
0.49	98.37%	23.17%	0.35	**1**	100.00%	0.00%	0.00
0.5	98.48%	22.07%	0.34				

(C) The experiment result of smoothed PSSM with –b option in SVM by three-way data split.

Threshold	**Spec.**	**Sens.**	**MCC**	**Threshold**	**Spec.**	**Sens.**	**MCC**
0	0.00%	100.00%	0.00	**0.51**	97.78%	27.87%	0.37
0.01	5.11%	99.80%	0.08	**0.52**	97.90%	27.24%	0.37
0.02	17.04%	98.55%	0.14	**0.53**	98.00%	26.42%	0.36
0.03	29.12%	96.48%	0.19	**0.54**	98.11%	25.60%	0.36
0.04	40.02%	94.64%	0.23	**0.55**	98.18%	24.58%	0.35
0.05	49.04%	91.90%	0.26	**0.56**	98.32%	23.25%	0.34
0.06	56.72%	89.24%	0.29	**0.57**	98.39%	22.15%	0.34
0.07	64.12%	85.87%	0.32	**0.58**	98.47%	21.14%	0.33
0.08	69.08%	83.56%	0.35	**0.59**	98.57%	20.16%	0.32
0.09	72.94%	80.86%	0.37	**0.6**	98.66%	19.10%	0.31
0.1	76.05%	77.57%	0.37	**0.61**	98.73%	18.08%	0.31
0.11	78.68%	75.58%	0.39	**0.62**	98.80%	17.26%	0.30
0.12	80.65%	73.62%	0.40	**0.63**	98.90%	16.75%	0.30
0.13	82.53%	71.51%	0.41	**0.64**	98.96%	15.85%	0.29
0.14	83.96%	69.32%	0.41	**0.65**	99.09%	14.60%	0.28
0.15	85.19%	67.71%	0.42	**0.66**	99.17%	13.74%	0.27
0.16	86.25%	66.22%	0.42	**0.67**	99.24%	12.92%	0.27
0.17	87.24%	64.85%	0.43	**0.68**	99.29%	12.09%	0.26
0.18	88.05%	62.97%	0.43	**0.69**	99.32%	11.78%	0.26
0.19	88.82%	61.33%	0.43	**0.7**	99.37%	11.08%	0.25
0.2	89.52%	60.27%	0.43	**0.71**	99.44%	10.06%	0.24

0.21	90.20%	59.14%	0.44	**0.72**	99.48%	9.35%	0.23
0.22	90.74%	57.65%	0.44	**0.73**	99.54%	8.85%	0.23
0.23	91.31%	56.28%	0.44	**0.74**	99.61%	8.49%	0.23
0.24	91.72%	55.11%	0.44	**0.75**	99.68%	7.55%	0.22
0.25	92.13%	54.05%	0.44	**0.76**	99.69%	7.05%	0.21
0.26	92.53%	52.96%	0.44	**0.77**	99.71%	6.22%	0.19
0.27	92.91%	51.90%	0.44	**0.78**	99.75%	5.83%	0.19
0.28	93.26%	50.88%	0.44	**0.79**	99.78%	5.01%	0.18
0.29	93.61%	50.02%	0.44	**0.8**	99.84%	4.23%	0.16
0.3	93.89%	48.81%	0.44	**0.81**	99.87%	3.84%	0.16
0.31	94.16%	47.83%	0.43	**0.82**	99.90%	3.44%	0.15
0.32	94.42%	46.89%	0.43	**0.83**	99.92%	2.90%	0.14
0.33	94.59%	46.18%	0.43	**0.84**	99.93%	2.39%	0.13
0.34	94.83%	44.78%	0.43	**0.85**	99.93%	1.80%	0.11
0.35	95.08%	43.84%	0.43	**0.86**	99.95%	1.49%	0.10
0.36	95.30%	42.54%	0.42	**0.87**	99.95%	1.25%	0.09
0.37	95.54%	41.49%	0.42	**0.88**	99.96%	1.21%	0.09
0.38	95.73%	40.43%	0.42	**0.89**	99.97%	0.82%	0.07
0.39	95.91%	39.22%	0.41	**0.9**	99.97%	0.59%	0.06
0.4	96.10%	37.89%	0.40	**0.91**	99.97%	0.35%	0.04
0.41	96.30%	37.18%	0.40	**0.92**	99.98%	0.31%	0.04
0.42	96.47%	36.20%	0.40	**0.93**	99.98%	0.23%	0.03
0.43	96.61%	35.23%	0.40	**0.94**	99.99%	0.20%	0.03
0.44	96.77%	34.21%	0.39	**0.95**	99.99%	0.08%	0.02
0.45	96.88%	33.19%	0.39	**0.96**	99.99%	0.04%	0.01
0.46	97.09%	32.52%	0.39	**0.97**	100.00%	0.04%	0.02
0.47	97.22%	31.62%	0.39	**0.98**	100.00%	0.00%	0.00
0.48	97.39%	30.57%	0.38	**0.99**	100.00%	0.00%	0.00
0.49	97.52%	29.47%	0.38	**1**	100.00%	0.00%	0.00
0.5	97.74%	28.30%	0.37				

(D) The experiment result of standard PSSM with –b option in SVM by three-way data split.

Threshold	Spec.	Sens.	MCC	Threshold	Spec.	Sens.	MCC
0	0.00%	100.00%	0.00	**0.51**	98.66%	20.08%	0.33
0.01	3.04%	99.73%	0.05	**0.52**	98.73%	19.33%	0.32
0.02	12.44%	98.79%	0.11	**0.53**	98.79%	18.55%	0.31
0.03	23.07%	96.67%	0.16	**0.54**	98.85%	17.77%	0.31
0.04	33.19%	94.52%	0.19	**0.55**	98.92%	17.26%	0.31
0.05	42.39%	91.66%	0.22	**0.56**	98.94%	16.71%	0.30
0.06	50.56%	88.22%	0.25	**0.57**	98.98%	16.20%	0.30
0.07	58.88%	84.74%	0.28	**0.58**	99.02%	15.62%	0.29
0.08	64.43%	81.53%	0.30	**0.59**	99.08%	15.19%	0.29
0.09	68.83%	78.75%	0.32	**0.6**	99.14%	14.68%	0.29
0.1	72.64%	76.16%	0.33	**0.61**	99.21%	14.09%	0.28
0.11	75.86%	73.11%	0.34	**0.62**	99.23%	13.54%	0.28

0.12	78.46%	70.57%	0.35	**0.63**	99.25%	13.27%	0.27
0.13	80.62%	67.83%	0.36	**0.64**	99.28%	12.84%	0.27
0.14	82.56%	65.83%	0.37	**0.65**	99.33%	12.25%	0.26
0.15	84.25%	63.68%	0.37	**0.66**	99.37%	11.74%	0.26
0.16	85.67%	61.25%	0.38	**0.67**	99.40%	11.55%	0.26
0.17	86.86%	58.98%	0.38	**0.68**	99.42%	11.23%	0.26
0.18	88.03%	57.30%	0.38	**0.69**	99.45%	10.88%	0.25
0.19	89.02%	55.07%	0.38	**0.7**	99.46%	10.45%	0.25
0.2	89.89%	53.31%	0.39	**0.71**	99.50%	9.82%	0.24
0.21	90.59%	51.90%	0.39	**0.72**	99.53%	9.32%	0.23
0.22	91.28%	50.72%	0.39	**0.73**	99.57%	8.85%	0.23
0.23	91.95%	49.24%	0.39	**0.74**	99.59%	8.26%	0.22
0.24	92.53%	47.48%	0.39	**0.75**	99.59%	8.06%	0.22
0.25	93.10%	45.91%	0.39	**0.76**	99.62%	7.67%	0.21
0.26	93.62%	44.58%	0.39	**0.77**	99.65%	6.97%	0.20
0.27	94.10%	43.17%	0.39	**0.78**	99.67%	6.54%	0.19
0.28	94.56%	41.76%	0.39	**0.79**	99.69%	6.14%	0.19
0.29	94.96%	40.20%	0.39	**0.8**	99.70%	5.64%	0.18
0.3	95.36%	38.83%	0.39	**0.81**	99.74%	5.24%	0.17
0.31	95.66%	37.50%	0.39	**0.82**	99.75%	4.85%	0.17
0.32	95.92%	36.59%	0.39	**0.83**	99.77%	4.54%	0.16
0.33	96.07%	35.89%	0.39	**0.84**	99.78%	3.99%	0.15
0.34	96.30%	34.56%	0.38	**0.85**	99.80%	3.64%	0.14
0.35	96.53%	33.58%	0.38	**0.86**	99.83%	3.41%	0.14
0.36	96.72%	32.52%	0.38	**0.87**	99.84%	3.09%	0.13
0.37	96.91%	31.43%	0.37	**0.88**	99.87%	2.62%	0.12
0.38	97.10%	30.53%	0.37	**0.89**	99.89%	2.00%	0.11
0.39	97.26%	29.55%	0.37	**0.9**	99.93%	1.53%	0.10
0.4	97.49%	28.57%	0.36	**0.91**	99.94%	0.98%	0.07
0.41	97.56%	27.91%	0.36	**0.92**	99.96%	0.70%	0.06
0.42	97.70%	26.97%	0.36	**0.93**	99.99%	0.47%	0.06
0.43	97.83%	26.26%	0.35	**0.94**	99.99%	0.35%	0.05
0.44	97.93%	25.28%	0.35	**0.95**	99.99%	0.12%	0.03
0.45	98.03%	24.27%	0.34	**0.96**	100.00%	0.08%	0.03
0.46	98.15%	23.56%	0.34	**0.97**	100.00%	0.00%	0.00
0.47	98.26%	22.66%	0.34	**0.98**	100.00%	0.00%	0.00
0.48	98.36%	21.76%	0.33	**0.99**	100.00%	0.00%	0.00
0.49	98.47%	21.21%	0.33	**1**	100.00%	0.00%	0.00
0.5	98.62%	20.39%	0.33				