

5-1-2009

# Prediction of Protein Function and Functional Sites From Protein Sequences

Jing Hu  
*Utah State University*

---

## Recommended Citation

Hu, Jing, "Prediction of Protein Function and Functional Sites From Protein Sequences" (2009). *All Graduate Theses and Dissertations*. Paper 292.  
<http://digitalcommons.usu.edu/etd/292>

This Dissertation is brought to you for free and open access by the Graduate Studies, School of at DigitalCommons@USU. It has been accepted for inclusion in All Graduate Theses and Dissertations by an authorized administrator of DigitalCommons@USU. For more information, please contact [becky.thoms@usu.edu](mailto:becky.thoms@usu.edu).



PREDICTION OF PROTEIN FUNCTION AND FUNCTIONAL SITES  
FROM PROTEIN SEQUENCES

by

Jing Hu

A dissertation submitted in partial fulfillment  
of the requirements for the degree

of

DOCTOR OF PHILOSOPHY

in

Computer Science

Approved:

---

Dr. Changhui Yan  
Major Professor

---

Dr. Donald H. Cooley  
Committee Member

---

Dr. Heng-Da Cheng  
Committee Member

---

Dr. Xiaojun Qi  
Committee Member

---

Dr. John R. Stevens  
Committee Member

---

Dr. Byron R. Burnham  
Dean of Graduate Studies

UTAH STATE UNIVERSITY  
Logan, Utah

2009

Copyright © Jing Hu 2009  
All Rights Reserved

## ABSTRACT

Prediction of Protein Function and Functional Sites

From Protein Sequences

by

Jing Hu, Doctor of Philosophy

Utah State University, 2009

Major Professor: Dr. Changhui Yan  
Department: Computer Science

High-throughput genomics projects have resulted in a rapid accumulation of protein sequences. Therefore, computational methods that can predict protein functions and functional sites efficiently and accurately are in high demand. In addition, prediction methods utilizing only sequence information are of particular interest because for most proteins, 3-dimensional structures are not available. However, there are several key challenges in developing methods for predicting protein function and functional sites. These challenges include the following: the construction of representative datasets to train and evaluate the method, the collection of features related to the protein functions, the selection of the most useful features, and the integration of selected features into suitable computational models. In this proposed study, we tackle these challenges by developing procedures for benchmark dataset construction and protein feature extraction, implementing efficient feature selection strategies, and developing effective machine learning algorithms for protein function and functional site predictions. We investigate

these challenges in three bioinformatics tasks: the discovery of transmembrane beta-barrel (TMB) proteins in gram-negative bacterial proteomes, the identification of deleterious non-synonymous single nucleotide polymorphisms (nsSNPs), and the identification of helix-turn-helix (HTH) motifs from protein sequence.

(148 pages)

## ACKNOWLEDGMENTS

I would like to thank Dr. Changhui Yan for his detailed direction and encouragement for the research in this dissertation. I would also like to thank my committee members, Dr. Don Cooley, Dr. Heng-Da Cheng, Dr. Xiaojun Qi, and Dr. John Stevens, for their inspiration, continuous supervision, and valuable advice throughout the entire process.

I cordially give thanks to my family, friends, and colleagues for their encouragement, moral support, and patience as I worked my way from writing the initial proposal to this final document. I could not have done it without all of you.

Finally, it should be noted that although I am not the principle author of the paper on which Chapter 4 is based, I conducted the majority of the research reported in Chapter 4. The research I did not participate in is not reported in said chapter.

Jing Hu

## CONTENTS

	Page
ABSTRACT.....	iii
ACKNOWLEDGMENTS .....	v
LIST OF TABLES .....	ix
LIST OF FIGURES .....	x
CHAPTER	
1 INTRODUCTION .....	1
1.1 Machine Learning and Bioinformatics Problems.....	1
1.2 Goals of This Dissertation .....	2
1.2.1 The Construction of Benchmark Datasets .....	2
1.2.2 The Compilation of Sequence-based Features.....	3
1.2.3 The Feature Selection Process .....	4
1.2.4 The Development of Appropriate Machine Learning Techniques ..	9
2 DISCOVERY OF TRANSMEMBRANE BETA-BARREL PROTEINS IN GRAM-NEGATIVE BACTERIAL PROTEOMES.....	13
2.1 Background.....	13
2.1.1 The Importance of TMB Proteins .....	13
2.1.2 The Need for Computational Methods to Identify TMB Protein ..	14
2.1.3 Current Methods for the Identification of TMB Proteins .....	15
2.1.4 Current Methods to Predict the Topology of TMB Proteins .....	20
2.1.5 Motivation of This Study .....	22
2.2 Materials and Methods.....	23
2.2.1 Datasets .....	23
2.2.2 Feature Set .....	24
2.2.3 Five-Fold Cross-Validation .....	26
2.2.4 K-NN Algorithm.....	26
2.2.5 Feature Selection.....	29
2.2.6 Performance Measurement .....	31
2.3 Results.....	32

2.3.1	The Proposed K-NN Method's Ability to Identify TMB Proteins	32
2.3.2	Including Homologous Sequence Information Improves the Performance .....	32
2.3.3	Further Improvement of the Prediction Performance by Feature Selection.....	33
2.3.4	Comparison with Predictions Solely Based on Similarity Search	33
2.3.5	Comparison with Other Prediction Methods .....	34
2.3.6	A Web Server for the Prediction of TMB Proteins.....	37
2.3.7	Genome Scan .....	37
2.4	Discussion .....	41
2.5	Conclusion .....	43
2.6	Future Work .....	43
3	IDENTIFICATION OF DELETERIOUS NON-SYNONYMOUS SINGLE NUCLEOTIDE POLYMORPHISMS .....	45
3.1	Background.....	45
3.1.1	Neutral or Deleterious nsSNPs .....	45
3.1.2	Current Methods for the Identification of Deleterious nsSNPs .....	46
3.1.3	Motivation of This Study .....	54
3.2	Materials and Methods.....	55
3.2.1	Datasets .....	55
3.2.2	Feature Set .....	56
3.2.3	Decision Tree Algorithm .....	61
3.2.4	Performance Measurement .....	63
3.2.5	Cross-Validation and Independent Test.....	64
3.2.6	Feature Selection.....	65
3.3	Results.....	66
3.3.1	The Developed Method Identifies Deleterious nsSNPs.....	66
3.3.2	Analysis of Selected Features .....	69
3.3.3	Comparisons with Previously Published Methods .....	73
3.3.4	A Web Server for the Identification of Deleterious Non-synonymous Single Nucleotide Polymorphisms .....	78
3.4	Discussion .....	78
3.5	Conclusion .....	80
3.6	Future Work .....	80
4	IDENTIFICATION OF HELIX-TURN-HELIX MOTIFS FROM PROTEIN SEQUENCES .....	83

4.1	Background.....	83
4.1.1	Helix-Turn-Helix: An Important Structure Through Which Proteins Bind with DNA.....	83
4.1.2	Current Prediction Methods to Identify Helix-Turn-Helix Motif..	84
4.1.3	Motivation of This Study.....	91
4.2	Materials and Methods.....	92
4.2.1	Datasets.....	92
4.2.2	Hidden Markov Model.....	94
4.2.3	Feature Set.....	98
4.2.4	Software Implementation.....	101
4.2.5	Performance Measurements.....	101
4.3	Results.....	101
4.3.1	Discretization of Solvent Accessibility.....	101
4.3.2	Constructing Profiles for Each HTH Protein Family Increases the Prediction Accuracy.....	107
4.4	Discussion.....	112
4.5	Conclusion.....	115
4.6	Future Work.....	116
5	CONCLUSION.....	117
	REFERENCES.....	120
	VITA.....	135

## LIST OF TABLES

Table		Page
1	Comparison of the Proposed <i>K-NN</i> Method with A Similarity Search. ....	34
2	Comparison of Different TMB Prediction Methods. ....	35
3	Prediction Results of 11 Gram-Bacteria Proteomes. ....	39
4	All Sequence-based Features of SAP Sites. ....	58
5	Prediction Performances of the Proposed Method. ....	68
6	List of Features in the Order They Are Selected. ....	68
7	Comparisons of Classification Methods of SAPs. ....	76
8	List of Reduced Alphabets. ....	100
9	HMM_AA_SA Achieves Better Performance Than HMM_AA by Dividing Solvent Accessibility into Two Discrete Categories. ....	104
10	HMM_AA_SA's Performance Can Be Improved by Dividing Solvent Accessibility into Three Discrete Categories. ....	106
11	Including Solvent Accessibility Information into the Model and Using Reduced Alphabet Increase Performance in Identifying HTH Motifs. ....	109
12	HMM_AA_SA Recognizes More HTH Motifs from Other Families. ....	110

## LIST OF FIGURES

Figure		Page
1	TMB proteins prediction web server .....	38
2	Classification performances as feature selection process progresses .....	67
3	Decision tree trained on 10 selected features as visualized using WEKA .....	71
4	ROC curves of the proposed decision tree method, SIFT and PANTHER on Ye's dataset.....	77
5	ROC curves of the proposed decision tree method, SIFT and PANTHER on Swiss-Prot dataset.....	77
6	The web server for the prediction of deleterious SAPs .....	81
7	Images of HTH motifs .....	84
8	Hidden Markov model (right) that emits only amino acid residues .....	96
9	Hidden Markov model that emits both amino acids and solvent accessibility ..	97
10	The performance of HMM_AA_SA with solvent accessibility being divided into two categories .....	103
11	The performance of HMM_AA_SA with solvent accessibility being divided into three categories ( $\alpha_1, \alpha_2$ ) with $\alpha_1 = 0.05$ .....	106

# CHAPTER 1

## INTRODUCTION

### **1.1 Machine Learning and Bioinformatics Problems**

With the development of high-throughput genome sequencing projects in recent years, we have witnessed an exponential accumulation of biological data stored in public databases, i.e., DNA, RNA, and protein sequences. Because of their limitations and speed, experimental approaches can hardly keep up with the accumulation of new biological data. On the other hand, machine learning methods, which rely heavily on Bayesian probabilistic frameworks, are widely applied to learning knowledge and extracting information automatically from huge amount of biological data [1, 2, 3, 4].

Bioinformatics is a field that merges biology, computer science, and statistics into a single discipline.

Machine learning methods have achieved significant success in many bioinformatics problems. For example, neural networks have been widely applied to predict protein secondary structure from amino acid sequences [5, 6, 7], to predict protein signal peptides and their cleavage sites [8, 9], to find genes in eukaryotic DNA, to identify intron splice sites [10, 11], etc. Hidden Markov models have been proven to be useful in protein pair-wise sequence alignment [12, 13], multiple sequence alignment [14], protein homology detection [13, 15], protein structure prediction [16, 17], topology annotation for alpha-helical transmembrane proteins [18, 19], beta barrel proteins [20], and genomic annotation [21, 22], etc. Other machine learning techniques, such as support

vector machine, decision tree, random forest, k-nearest neighbor are also used to solve many bioinformatics problems.

## **1.2 Goals of This Dissertation**

Numerous computational methods have been developed to predict protein function and functional sites by using information derived from protein sequences and structures. However, the 3-dimensional structural information of most proteins is not available, which limits the application of structure-based methods. Therefore, computational methods that only require sequence information are key because they have a broader range of applications than structure-based methods.

In this study, we develop efficient machine-learning approaches to discover the attribute-class relationship between sequence features and protein functions. There are several challenges in constructing computational methods with high performance. These challenges include the topics discussed in Sections 1.21 through 1.24 below.

### *1.2.1 The Construction of Benchmark Datasets*

In order to develop efficient and accurate computational methods, it is necessary to construct a highly representative dataset of sufficient size. An inappropriate dataset will significantly deteriorate the performance of the method and yield misleading results in the evaluation. For some bioinformatics problems, widely validated benchmark datasets have already been constructed. Conversely, for many other problems, it is necessary to compile nonredundant and representative datasets for the purpose of method development and evaluation. However, the construction of benchmark datasets is not a trivial task. Usually, experimentally validated data is distributed over multiple databases.

In order to construct benchmark datasets, several databases have to be queried, and collected data has to be processed and purified. The process is often very inefficient and lengthy. Whenever there are updates of databases, the whole process has to be re-executed. An automatic dataset updating strategy helps to solve the problem, allowing the re-construction of the dataset whenever a new release of a database is published. For example, DOCKGROUND [23] is a comprehensive database of cocrystallized (bound-bound) protein-protein binding complexes that can be regularly updated to reflect the growth of the protein data bank (PDB) [24]. In this study, we have constructed and selected representative nonredundant datasets for the target problems.

### *1.2.2 The Compilation of Sequence-based Features*

There is limited knowledge about which features are useful for the prediction of protein functions and functional sites. Therefore, it is necessary to collect various sequence-based features for future analysis. Some sequence-based features can be extracted directly from an amino acid sequence. For example, the composition of amino acid residues, di-peptides, and n-peptides of proteins can be easily computed from protein sequences. Certain physicochemical and biochemical properties of amino acids can be obtained from public databases such as AAindex [25]. Other features, such as relative solvent accessibility and conservation score of each residue position, are very useful in inferring protein functions, but are not directly available from protein sequences. Extracting these features on a large scale poses great challenges. We address these difficulties by generating multiple important features using various approaches. For example, residue frequency and a conservation score of each residue position can be

calculated from the multiple sequence alignment of the interested protein with its homologous sequences found by a PSI-BLAST [26] search. Solvent accessibility of each residue can be predicted from amino acid information by JPred server [27].

### *1.2.3 The Feature Selection Process*

Not all sequence-based features are useful for the prediction of protein functions and functional sites. Simply using all features without analysis and processing will only complicate the problem and generally deteriorate the prediction performance most of the time. However, because of the extreme complexity of biological systems, there are not many mature theories exploring the relationship between protein features and protein functions. Therefore, feature selection is commonly adopted to facilitate the understanding of a biological system, to reduce the noise in the biological data, and to improve prediction performance. In addition, the dimensionality of the feature space is reduced after feature selection process; therefore, the learning and prediction processes are sped up.

There are several hurdles complicating the feature selection problem. Due to the high dimensionality of the feature space, a brute force feature search method is not practical because the search space grows exponentially with the number of individual features. In general, there are two categories of feature selection methods.

*1.2.3.1 Manual Feature Selection.* The best feature selection method is to choose features manually based on a deep understanding of the problem and the biochemical/biophysical meanings of the features. As to some well studied bioinformatics problems, some features have already been proved to be useful based on

previous findings. For example, solvent accessibility of amino acid residues has been shown to be useful in the prediction of functional effects of single nucleotide polymorphisms [28] and protein secondary structures [29]. Hydrophobicity and charge distribution of amino acids were verified to be important attributes to predict integral topology of membrane proteins [30]. Amino acid compositions have been used to predict protein structural class [31] and protein subcellular location [32, 33] with high performance. In fact, many bioinformatics problems are solved by manually selecting important features based on detailed studies of biologists and integrating these features with appropriate computational methods.

*1.2.3.2 Automatic Feature Selection Algorithm.* In cases where there are too many candidate features or it is not clear which features are useful, systematic feature selection techniques can be applied to automatically choose a reduced feature subset that is most effective for prediction. Depending on how to combine a feature space search with the construction of a classification model, automatic feature selection methods can be classified into three subcategories, filter techniques, wrapper methods, and embedded methods [34, 35].

*1.2.3.2.1 Filter Techniques.* Attributes are ranked based on their relevance scores to produce the most relevant feature subset. The resulting features are then fed into certain classification methods to make predictions. There are no universally accepted relevance score measurements, though there are several good practices, such as information gain, gain ratio [36], and Fisher discriminant ratio (FDR). Another feature ranking strategy is also widely used. In this strategy, each feature is chosen individually

to build the predictor, and the prediction performance of the corresponding model is used as the relevance score. There are several advantages with filter methods. First, each feature is evaluated only once. Second, the method is scalable and independent of classifiers. Therefore, the filter method has been widely applied in solving many bioinformatics problems.

For example, in the study of [37], a filter method based on  $\chi^2$  test was applied to select relevant features from a large set of sequence based features. Then, the selected features were input to the interpolated Markov models (IMMs) to build the GLIMMER method, which is capable of finding genes in microbial genomes. The 1R algorithm and information gain were applied by [28] to rank 17 attributes for the identification of deleterious nonsynonymous single nucleotide polymorphisms (nsSNPs). Also, for the problem of identifying deleterious nsSNPs, the authors of [38] evaluated the entire set of attributes in terms of their association with disease and selected the final subset in terms of good predictive power. In their study, support vector machines were built from each single attribute, and cross-validation accuracies were used as relevance scores to rank all attributes. In another example, FDR has been proved to be an efficient feature selection method to predict outer member proteins by [39].

Despite their benefits, simple filter methods do have their inherit shortcomings. In using a filter method, each feature is considered separately; thus, feature dependencies are ignored, which may lead to worse classification performances when top ranked features are combined together. Furthermore, if there are pairs of redundant attributes, the filter method either selects or discards both attributes. These problems can be solved by

selecting a subset of attributes that have low inter-correlation scores or by introducing multivariate filter methods such as simple bivariate interactions [40].

*1.2.3.2 Wrapper Method.* In a wrapper method, a feature selection algorithm is “wrapped” around the classification model. The effectiveness of a feature subset is evaluated by training and testing the classifier built with the selected features. Different from filter techniques which are independent of the classifier, a wrapper method may select a different feature subset for different classifiers. Due to the exponential growth of the feature subset space with the number of individual features, usually heuristic search methods are used to search for the optimal feature subset. These search methods can be divided into two types, a deterministic search algorithm and a randomized search algorithm.

Greedy search (i.e., forward selection, backward elimination, and bidirectional search) [41], best first search [35, 42] and beam search [43] are all examples of deterministic searches. Forward selection starts with no feature selected. Each attribute is tentatively added to the current feature subset, and the resulting feature set is evaluated by its prediction performance. The feature with the highest improvement when included into the current subset is chosen. Then, the algorithm finds the next feature in the same way. The process continues until adding any remaining features only deteriorates the performance. Backward elimination works in the opposite direction by starting with all features and then gradually removing one feature a time to improve the performance. A bidirectional search is a combination of the forward selection and backward elimination. A greedy search method does not ensure a globally optimal feature subset. Different from

a greedy search method, a best first [35, 43] method can find the global optimal configuration of features instead of stagnating in the local optimum. This method does not terminate when the performance begins to drop down. Instead, it keeps a list of all feature subsets so far in sorting order, which can be revisited later. A beam search [43] keeps a fixed number of the most promising candidate features by truncating its list of features at each stage.

A randomized search algorithm includes simulated annealing, randomized hill climbing [44], and genetic algorithm search procedures [45]. Notice that in a genetic algorithm search, the selection of features is configured and encoded using a sequence of binary bits, with 1 representing a chosen feature and 0 denoting non-chosen. The best feature subset is then evolved by an evolutionary search after certain iterations until convergence.

Wrapper methods have been applied by some groups to select features in solving bioinformatics problems. For example, a bidirectional greedy search strategy was employed by [39] to find the best feature subset to build a support vector machine in the identification of outer membrane proteins. The algorithm can be divided into two stages, backward elimination and forward selection, to select useful features from 20 amino acid compositions and 400 di-peptide (amino acid pair) compositions. In the backward stage starts with 20 amino acids compositions, and then gradually reduces the size of the feature set, similar to backward elimination process. The process stops when decreasing the size of the current best subset leads to a lower prediction rate. The forward selection stage takes over to include di-peptide composition until there is no more improvement of

the prediction performance. Cross-validated classification accuracy has been used as performance measurement of this method.

Wrapper methods take into consideration feature dependencies and integrate the feature selection process with classification methods; thus the chosen features will best fit the computational model. The disadvantages of this method are also obvious. The selection process is not as computationally efficient as that of filter techniques, and it may introduce over-fitting problems.

*1.2.3.2.3 Embedded Techniques.* In embedded techniques, the feature selection process is built into the construction of the classifier. For example, the construction of a decision tree is such a process. It chooses the most promising attribute to split on at each node. The final set of features actually chosen to build the tree is the best feature set selected. Similar to wrapper methods, embedded techniques select features specific to the classifiers, meanwhile these techniques have the benefit of fast speed and fewer computations.

In this study, we implement efficient feature selection methods to choose the most useful features to solve certain important bioinformatics problems. The chosen features are analyzed to provide further insights into mechanisms of biological processes.

#### *1.2.4 The Development of Appropriate Machine Learning Techniques*

There are several expert practices for choosing the appropriate machine learning techniques for solving certain problems. For example, hidden Markov models are well suitable for the prediction of alpha-helical transmembrane proteins, since each state of the model can represent the cell position of each amino acid [18, 19]. Neural networks and

support vector machines are efficient classification algorithms. However, due to the special complexity of bioinformatics problems, most of the time choosing certain machine learning algorithms based on previous experiences without careful study of problems does not ensure satisfactory solutions.

The challenges for predicting protein function and function sites vary depending on the specific problems. We have tackled these challenges by solving three important bioinformatics problems. These include the following: the discovery of transmembrane beta-barrel (TMB) proteins in gram-negative bacterial proteomes (discussed in Chapter 2), the identification of deleterious non-synonymous single nucleotide polymorphisms (nsSNPs) (discussed in Chapter 3), and the identification of helix-turn-helix (HTH) motifs from protein sequences (discussed in Chapter 4).

For each project, we collected various features derived from protein sequences. Given the high number of sequence-derived features collected, the next step was to choose the most relevant and useful features for prediction. For the first two problems, we applied automatic feature selection procedures (i.e., wrapper methods) to select relevant features. For the third problem, we selected certain feature combinations based on a detailed study of the problem. Using selected features, we then applied suitable machine learning algorithms to build predictors of high performance. The predictors were trained and evaluated on benchmark datasets, which were either constructed in this study or derived from previous studies.

To develop the predictor of TMB proteins, we first extracted datasets of TMB proteins and non-TMB proteins from public databases. Next, a set of features, including

20 residue compositions and 400 di-peptide compositions, were compiled for each protein. We then applied a greedy feature selection approach to choose the most useful features. The feature selection process contains a reduce stage and a growth stage. In the end, compositions of 19 residues and 24 di-peptides were selected. Using the selected features to calculate weighted Euclidian distances, we developed a K-nearest neighbor method that can discriminate TMB proteins from non-TMB proteins with high performances. The developed method was used to scan 11 proteomes of gram-negative bacteria for possible TMB proteins. The details of the method and results have been published at:

Hu, J. and Yan, C. A method for discovering transmembrane beta-barrel proteins in gram-negative bacteria proteomes. *Computational Biology and Chemistry* 32, 4 (2008), 298-301.

To develop the method for identifying deleterious nsSNPs, we obtained datasets from previous studies. For each amino acid substitution, we compiled a set of 686 features from protein sequences. Next, a greedy feature selection approach was used to select features that were useful for the classification of nsSNPs. The feature selection strategy is similar to that used in problem 1, except there is only a growth stage. Using ten selected features, a decision tree method is capable of identifying deleterious nsSNPs on proteomic scale. The details of the method and result have been published in:

Hu, J. and Yan, C. Identification of deleterious non-synonymous single nucleotide polymorphisms using sequence-derived information. *BMC Bioinformatics* 9 (2008), 297.

To identify HTH motifs from protein sequences, we first constructed datasets of HTH and non-HTH proteins. Then, we expanded the traditional profile hidden Markov model by allowing both match and insertion states to emit both amino acid and solvent accessibility information. The solvent accessibility of each residue is predicted from the protein sequence and discretized into three states (Buried (B), Medium (M), and Exposed (E)), wherein discretization thresholds are chosen by a systematic analysis. To reduce the number of parameters and the complexity of the protein sequences, several reduced alphabets of amino acids were investigated. We tried different combinations of feature subsets (e.g., amino acid plus solvent accessibility, and reduced alphabets plus solvent accessibility), and found that using certain reduced alphabets and predicted solvent accessibility, the developed profile hidden Markov model can effectively identify HTH motifs. The details of the method and results have been published in:

Yan, C. and Hu, J. An exploration to the combining of solvent accessibility with amino acid sequence in the identification of helix-turn-helix motifs. *WSEAS Transaction on Biology and Biomedicine* 6, 3 (2006), 477-484.

Yan, C. and Hu, J. Identification of helix-turn-helix motifs from amino acid sequence. In *Proc. of IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, 2006, 1-7.

Yan, C. and Hu, J. A hidden Markov model for the identification of helix-turn-helix motifs. In *Proc. of WSEAS International Conference on Cellular and Molecular Biology-Biophysics and Bioengineering*, 2006, 14-19.

CHAPTER 2  
DISCOVERY OF TRANSMEMBRANE BETA-BARREL PROTEINS  
IN GRAM-NEGATIVE BACTERIAL PROTEOMES<sup>1</sup>

## 2.1 Background

### 2.1.1 *The Importance of Transmembrane Beta-barrel Proteins*

Transmembrane proteins can be divided into two classes based on the structure of the transmembrane regions: transmembrane  $\alpha$ -helical (TMA) proteins whose transmembrane regions are  $\alpha$ -helices, and transmembrane  $\beta$ -barrel (TMB) proteins whose transmembrane segments are anti-parallel  $\beta$ -strands in the form of beta-barrel. TMA proteins, which are typical membrane proteins, can be found in the plasma membrane (inner membrane) of both prokaryotic and eukaryotic organisms, and perform a variety of biologically important functions [46]. On the other hand, TMB proteins only reside in the outer membrane of gram-negative bacteria and the outer membranes of mitochondria and chloroplasts [47, 48]. These proteins perform diverse important functional roles, such as bacterial adhesion, structural integrity of the cell wall, material transport, and catalytic activity [47, 48, 49]. Once the structure of a TMB protein is obtained, the transmembrane beta-barrels can be identified using computational methods that model the transmembrane environment [50].

Due to the importance of transmembrane proteins, computational methods that can identify and predict transmembrane proteins are in high demand. Among them, TMA proteins are relatively easy to predict for several reasons. First, their transmembrane

---

<sup>1</sup> Co-authored by Hu, J. and Yan, C.

regions are generally formed by easily detectable long hydrophobic sequence stretches. Second, there is a strong bias in transmembrane regions toward positively charged residues, known as the “positive-inside rule” [51]. Third, experimentally determined alpha-helical proteins with high 3-dimensional resolution structures are relatively abundant. Thus, several computational methods, such as statistical methods, neural networks, and hidden Markov models, achieve high performance in the identification and topology prediction of TMA proteins [18, 19].

### *2.1.2 The Need for Computational Methods to Identify TMB Proteins*

Unlike TMA proteins, TMB proteins are much harder to predict due to their much shorter transmembrane stretches of amino acids and their lack of clear patterns in their membrane spanning regions [49]. In general, amino acids in the TMB strands alternate between being polar and nonpolar, with nonpolar residues facing the lipid bilayer and the protein interfaces, and the polar residues pointing into the interior of the barrel. Residues pointing inwards in the barrel can also be nonpolar, thus obstructing the regular alternation between polar and nonpolar residues [47, 52]. Furthermore, discrimination of transmembrane strands from other beta-strands forming beta-barrel structures in water-soluble proteins is even more difficult [53].

Since very few TMB proteins have 3-dimensional structural information which is experimentally resolved and it is very difficult to crystallize TMB proteins in labs, computational methods that can identify TMB proteins and predict the topology of TMB proteins are in high demand. A lot of methods have been proposed to solve this problem.

Some methods try to classify TMB proteins from non-TMB proteins. Other methods focus on the prediction of transmembrane topologies of TMB proteins.

### *2.1.3 Current Methods for the Identification of TMB Proteins*

Despite these many difficulties, numerous methods of identifying TMB proteins have been published. For example, profiles constructed from structurally conserved regions of porins have been used by [54] as a potential tool to identify beta-stranded integral membrane proteins. The basic structural motif of the porins (i.e., the beta-barrel that forms the transmembrane core) consists of 16 beta strands in general-diffusion porins and 18 beta strands in sugar-specific porins. The authors of [55] calculated the probability for the existence of beta-sheet, beta-barrel, and hairpin structures among all proteins of the *Arabidopsis thaliana* genome. Based on the existence of these structures, the authors generated a pool of candidate proteins. The pool was further trimmed by considering the signal peptide information. Another computational tool, named beta-barrel finder (BBF) program, was developed by [56] to classify the proteins within any completely sequenced prokaryotic genome. Using information such as secondary structure, hydrophathy, and amphipathicity, as well as the presence of an N-terminal targeting sequence, the BBF retrieved 118 proteins out of 4290 sequences within the *Escherichia coli* genome as TMB proteins. After analyzing the amino acid composition in membrane parts of 12  $\beta$ -barrel membrane proteins versus  $\beta$ -strands of 79 all- $\beta$  soluble proteins, the authors of [57] developed a linear classifier built with selected amino acids composition and predicted secondary structure. The method achieved 85.48% sensitivity and 92.53% specificity when applied to 241  $\beta$ -barrel membrane proteins and 3855 soluble proteins with various

structures. The hidden Markov model has also been successfully applied by [58] to identify TMB protein. In this method, a hidden Markov model (HMM) with an architecture obeying  $\beta$ -barrel membrane proteins' construction principles was trained. The authors used the log-odds score relative to a null model to discriminate TMB proteins from other proteins. Because of its speed, the method is capable of scanning proteomes for TMB proteins.

Recently, two state-of-the-art methods, i.e., TMB-Hunt [59, 60] and BOMP [52], have been developed for the quick identification of TMB proteins. Both TMB-Hunt and BOMP have free on-line prediction servers available for public usage, making it possible to compare them with our method. Other published TMB classification methods include [61, 62, 63, 64, 65].

*2.1.3.1 TMB-Hunt.* TMB-Hunt<sup>2</sup> [59, 60] uses a modified k-nearest neighbor (K-NN) algorithm to classify proteins as TMB or non-TMB on the basis of whole sequence amino acid composition. The K-NN algorithm is a simple instance-based learning algorithm in which the class (i.e., TMB, TMA, or nontransmembrane) of a query protein is predicted using the class of its k-nearest neighbors within the training set. Since the K-NN algorithm focuses on the neighborhood of the query instance, it needs a way to define distances between proteins. The distance or difference between two proteins  $d^2(x_i, x_j)$  is measured using the standard Euclidean metric

$$d^2(x_i, x_j) = \sum_{r=1}^n [a_r(x_i) - a_r(x_j)]^2, \quad (2.1)$$

---

<sup>2</sup> The TMB-Hunt web server is available online at the following website: [http://bmbpcu36.leeds.ac.uk/~andy/betaBarrel/AACompPred/aaTMB\\_Hunt.cgi](http://bmbpcu36.leeds.ac.uk/~andy/betaBarrel/AACompPred/aaTMB_Hunt.cgi). It takes FASTA format sequences as input.

where  $a_r(x)$  is the relative frequency occurrence of amino acid  $r$  in protein  $x$ . Then a score  $S(x_q, x)$  can be assigned to each possible class  $c$  using

$$S(x_q, x) = \sum_{i=1}^k \delta[c, c(x_i)] / d^2(x_q, x_i) \quad (2.2)$$

where  $\delta(c, c(x_i)) = 1$  if classes  $c$  and  $c(x_i)$  are equal and 0 otherwise. Thus, the score for each class is the sum of positive contributions from the nearest neighbors of that class, where contribution is weighed according to their reciprocal square distances. Since the problem is a binary classification problem, a discrimination score

$$D(x_q, c) = S(x_q, c) - \sum_{c' \neq c} S(x_q, c') \quad (2.3)$$

is used as score for the TMB class minus the scores for other classes.

Considering the fact that some dimensions provide more information to the classification, a genetic algorithm was used to calculate the optimal weighting of each dimension. Then, the distance was modified to

$$d^2(x_i, x_j) = \sum_{r=1}^n g_r [a_r(x_i) - a_r(x_j)]^2 \quad (2.4)$$

where  $g_r$  is the weight of  $r^{\text{th}}$  dimension.

In order to improve the classification capability of the method, TMB-Hunt includes evolutionary information by a BLAST [26] query against Swiss-Prot with an E-value threshold of 0.0001 and a maximum of 25 homologues to calculate the average amino acid composition. After calibrating the score, TMB-Hunt achieved a 92.5% discrimination accuracy for TMB and nontransmembrane proteins, with a 91% sensitivity

rate and a 93.8% positive predictive value (PPV) using leave homologues out cross-validation on their datasets.

TMB-Hunt also has an option to include evolutionary information for the highest prediction performance. Because TMB-Hunt uses K-NN methods and only takes amino acid composition as inputs, the method is very fast and can scan the whole proteome to find candidate TMB proteins.

*2.1.3.2 BOMP.* BOMP<sup>3</sup> [52] is another efficient method to identify TMB proteins encoded within genomes of gram-negative bacteria. BOMP is composed of two independent methods for identifying possible TMB proteins and a filtering mechanism to remove false positives.

The first method [52] uses C-terminal patterns to discover possible integral TMB proteins. After careful analysis of the last 10 amino acids in the C-terminal end of 12 TMB proteins with resolved crystal structure and less than 70% conserved residues, a C-terminal pattern was extracted to compare with the far C-terminal end of the query protein sequence with a minimal length of 110 amino acids.

The second method [52] is based on the compositions of each amino acid in the external and internal positions of the membrane spanning segments, and the relative abundance to the genomic residue compositions calculated by [42]. Using the normalized amino acid distribution, a score is calculated for each sliding window of ten residues by taking the maximum of two scores, i.e., the scores obtained by summarizing the amino acids in the window starting with either internal or external amino acids. Next, the

---

<sup>3</sup> An online server of BOMP is provided at <http://www.bioinfo.no/tools/bomp>.

integral  $\beta$ -barrel score of a protein is calculated by adding the average of the eight highest-scoring non-overlapping windows and the average of the 12 lowest-scoring non-overlapping windows. Proteins with an integral  $\beta$ -barrel score above the user-defined threshold are predicted to be TMB proteins. The higher the score, the more reliable the prediction is.

In order to reduce the number of false positives (i.e., non-TMB proteins predicted to be TMB proteins), a filtering procedure [52] is carried out after the query protein is reported to be a TMB candidate by the previous two methods. By selecting nonredundant proteins with subcellular localization annotations from Swiss-Prot Release 42, a final reference set containing 1231 sequences is created, of which 110 are outer membrane proteins. When a protein is run through the filter, it is compared with the final reference set by using a k-nearest-neighbor method ( $k=5$ ) to determine if the candidate is a true TMB protein or not.

As a supplement, there is an additional BLAST [26] function for finding amino acid sequences which are highly similar to proteins with experimentally annotated subcellular localization in Swiss-Port. This supplement, then, either supports or contradicts the prediction results of BOMP.

BOMP [52] has achieved an 80% sensitivity rate and a 88% positive predictive value (PPV) on the proteins with Swiss-Port annotated subcellular locations in *Escherichia coli* K 12 (788 sequences).

#### *2.1.4 Current Methods to Predict the Topology of TMB Proteins*

In addition to methods that predict whether a protein is a TMB protein, many other methods aim at predicting the topology of a given TMB protein. Some of these topology-predicting methods can also discriminate TMB proteins.

Early computational methods tried to utilize hydrophobicity patterns to identify beta-barrel transmembrane strands [66, 67], or constructed special empirical rules using amino acid propensities and prior knowledge of structural nature to predict beta-barrel transmembrane segments [68, 69]. There are several shortcomings with these methods since such patterns and rules were generated from insufficient training datasets and did not catch the structural features of TMB proteins. Improved prediction performances of TMB topologies were realized recently by applying much more complex machine learning techniques, such as neural networks [70, 71, 72, 73], support vector machines [74], and hidden Markov models [20, 53, 57, 75, 76, 77]. Recently, Diao et al. [78] introduced cellular automata and Lempel-Ziv complexity to predict the TM regions of integral protein including both  $\alpha$ -helical and  $\beta$ -barrel membrane proteins. Other methods such as transFold [79] can also predict the structure of TMB proteins.

In a recent study, the authors of [80] made a systematic comparison of the topology-predicting methods and found that the best topology predictors are those methods based on hidden Markov models (HMMs). After evaluating these methods on a nonredundant dataset of 20 TMB proteins from gram-negative bacteria, the authors found that PRED-TMBB [53, 76] achieved the best performance in predicting the topology of TMB proteins and PROFtmb [20] achieved the second highest score in the comparison.

PRED-TMBB<sup>4</sup> [53, 76] is based on a hidden Markov model whose architecture is fitted to the limitations imposed by known TMB structures. Conditional maximum likelihood (CML) training [81] was employed to train the HMM for the labeled data. CML training maximizes the probability of the correct prediction rather than the probability of protein sequences generated by the HMM. The HMM was further retrained on the nonredundant dataset of TMB proteins whose 3-dimensional structures were recently solved. PRED-TMBB provides three decoding schema for the query protein: the standard Viterbi algorithm [82], the N-best algorithm [83], and the posterior decoding method using a dynamic programming algorithm. The posterior decoding method using a dynamic programming algorithm was found to achieve the best performance in locating the beta-barrel transmembrane strands. PROFtmb<sup>5</sup> [20] is based a profile hidden Markov model. In their study, the authors introduced a new definition of beta-hairpin motifs of model beta-barrel strands. The method can predict if a protein is a TMB protein by using a log-odds whole-protein discrimination score, Z-value. It can also label residues of TMB proteins with four states, upward- and downward-facing strands, periplasmic hairpins, and extracellular loops. PROFtmb can discriminate TMB from non-TMB very quickly and has been evaluated in a proteomic scale.

Both PRED-TMBB and PROFtmb can be used to identify TMB proteins. In addition, among all the methods that were investigated in [80], PROFtmb is the only

---

<sup>4</sup> PRED-TMBB is available to the public at the following website: <http://biophysics.biol.uoa.gr/PRED-TMBB/input.jsp>.

<sup>5</sup> PROFtmb is available for academic use at the following website: <http://rostlab.org/services/proftmb/>.

method that has been evaluated in a proteomic scale. Therefore, PRED-TMBB and PROFtmb were used for comparison with our proposed method.

### *2.1.5 Motivation of This Study*

Many of the previous methods achieved significant progress in identifying TMB proteins and predicting TMB topologies, yet the problem has not been fully solved. First, the prediction accuracy is still not very high. Second, many methods were developed a relatively long time ago; therefore, their training datasets are not up-to-date, representative, nor complete. In addition, many training datasets do not have sufficient TMB proteins with experimentally resolved 3-dimensional structures, thus making it difficult to train and evaluate the methods.

The purpose of this study is to predict whether a protein is a TMB protein, using only information derived from protein sequences, such that the model can be applied to scan the whole dataset of gram-negative bacterial proteomes to find candidate TMB proteins. Such an endeavor is of great significance to disease research and drug discovery. To this end, we have developed a K-nearest neighbor (K-NN) method. The method was trained and evaluated on benchmark datasets of TMB and non-TMB proteins with experimentally determined structures. Different from the K-NN method used in TMB-Hunt and BOMP, the proposed method uses a weighted Euclidian distance (WED) as a distance measurement. Said measurement was calculated using compositions of certain amino acids and di-peptides chosen by a systematic feature selection process.

## 2.2 Materials and Methods

### 2.2.1 Datasets

We compiled a set of TMB proteins that have been experimentally confirmed. There are two data sources from which one can extract TMB proteins, and we used both sources.

Source 1: SCOP (Structural Classifications of Proteins) is a publicly accessible database, and it is frequently updated. It manually classifies protein structure domains based on the evolutionary and structural relationships of proteins [84]. It includes 118 proteins that are classified as “transmembrane beta-barrels,” which are TMB proteins.

Source 2: TCDB (Transport Proteins Database) is a web-accessible, curated, relational database containing sequence, classification, structural, functional, and evolutionary information about transport systems of a variety of living organisms [85]. It includes 188 proteins belonging to the “ $\beta$ -Barrel porins” subclass.

The transmembrane portions of these 306 (118+188) proteins consist exclusively of  $\beta$ -strands which form a  $\beta$ -barrel; thus they are TMB proteins. Some proteins may share high sequence similarity. In order to better train and evaluate any method, it is necessary to remove redundancy. One common practice is to ensure the identity between any two proteins is less than 25% using BLAST [26], i.e., with less than 25% identical residues on 90% length coverage of any sequence. In training our method, proteins with less than 50 amino acids were removed, and proteins that were not from gram-negative bacterial were also discarded since the prediction is for gram-negative proteins. The final dataset consisted of 119 TMB proteins.

Non-TMB proteins were obtained from the PSORTdb database [86], which categorizes bacterial proteins based on their subcellular localizations. Using PSORTdb, we extracted all proteins from gram-negative bacteria whose subcellular locations had been experimentally confirmed. Proteins associated with any subcellular localization other than outer membrane were considered non-TMB proteins. Next, we removed redundant proteins so that the mutual identity was less than 25%. We also removed proteins with less than 50 amino acids. The final non-TMB proteins were divided into 6 sub-groups based on their subcellular localizations: 245 proteins from “Cytoplasmic,” 195 proteins from “CytoplasmicMembrane,” 15 proteins from “Cytoplasmic, CytoplasmicMembrane,” 165 proteins from “Periplasmic,” 35 proteins from “Periplasmic, CytoplasmicMembrane,” and 87 proteins from “Extracellular.”

The dataset of TMB and non-TMB proteins are available for downloading.<sup>6</sup>

## 2.2.2 Feature Set

*2.2.1.1 Residue Composition.* There are 20 different residue (amino acid) types.

Composition of each residue is calculated using

$$x_i = n_i / \sum_{j=1}^{20} n_j, \quad (2.5)$$

where  $n_i$  and  $n_j$  are the numbers of residues of types  $i$  and  $j$ . The average residue composition of TMB proteins is calculated using

$$\bar{x}_{i\_tmb} = n_{i\_tmb} / \sum_{j=1}^{20} n_{j\_tmb}, \quad (2.6)$$

---

<sup>6</sup> The datasets of TMB and non-TMB proteins are available at <http://yanbioinformatics.cs.usu.edu:8080/TMBKNNsubmit>.

where  $n_{i\_tmb}$  and  $n_{j\_tmb}$  are the total numbers of residues of types  $i$  and  $j$  in all TMB proteins in the training set. The averaged residue compositions for every subgroup of non-TMB proteins are calculated in a similar way. They are denoted as:  $\bar{x}_{i\_Cyt}$  (“Cytoplasmic” subgroup),  $\bar{x}_{i\_CytM}$  (“CytoplasmicMembrane” subgroup),  $\bar{x}_{i\_CCM}$  (“Cytoplasmic,CytoplasmicMembrane” subgroup),  $\bar{x}_{i\_Ext}$  (“Extracellular” subgroup),  $\bar{x}_{i\_Per}$  (“Periplasmic” subgroup) and  $\bar{x}_{i\_PCM}$  (“Periplasmic, CytoplasmicMembrane” subgroup).

*2.2.1.2 Di-peptide Composition.* A di-peptide consists of two amino acids connected by a single peptide bond. For example, if a protein sequence part is “VADV<sub>G</sub>,” there are four di-peptides, which are “VA” AD, DV, and VG.” In total, there are 400 types of di-peptides. The composition of each di-peptide is calculated in a similar way to residue composition, using

$$y_i = m_i / \sum_{j=1}^{400} m_j, \quad (2.7)$$

where  $m_i$  and  $m_j$  are the numbers of different di-peptides. The average di-peptide composition of TMB proteins is calculated using

$$\bar{y}_{i\_tmb} = n_{i\_tmb} / \sum_{j=1}^{20} n_{j\_tmb}, \quad (2.8)$$

where  $m_{i\_tmb}$  and  $m_{j\_tmb}$  are the total numbers of di-peptides of types  $i$  and  $j$  in all TMB proteins in the training set. The averaged di-peptide compositions for every subgroup of non-TMB proteins are calculated in a similar way. Similarly, we can calculate the n-peptide composition of any proteins, where n can be 3, 4, etc. When n is 2, n-peptides

becomes di-peptides. Di-peptides and n-peptides not only model sequence residue composition, but also provide sequential order information. However, each time we increase the order  $n$  by 1, the number of dimensions is 20 times larger. A large dimension of features can introduce problems of insufficient training. Therefore, only di-peptide compositions were investigated in this study.

### 2.2.3 Five-Fold Cross-Validation

Five-fold cross-validations were used to evaluate the proposed method. The overall dataset was divided into five subsets. In each round of experiment, four subsets were used as the training set, and the remaining subset was used as the test set. This procedure was repeated five times, with each subset being used as test set once.

### 2.2.4 K-NN Algorithm

A K-nearest neighbor (K-NN) algorithm is a typical instance-based learning method. It assumes all instances correspond to points in an  $n$ -dimensional space. The nearest neighbors of an instance are usually defined in terms of the standard Euclidian distance. Given a query instance  $x_q$ , the algorithm first finds its  $k$  nearest training instances. The class of  $x_q$  is then assigned to the most common class value among its  $k$  nearest training examples. One refinement to the K-NN algorithm is called a *distance-weighted K-NN* algorithm. It weights the contribution of each of the  $k$  neighbors according to their distance to the query point  $x_q$ , i.e., weights the vote of each neighbor according to the inverse square of its distance from  $x_q$ , giving greater weight to closer neighbors, as can be seen from Eq. (2.2) [59, 60, 87].

There are several advantages of the K-NN algorithm. First, the training and prediction procedures are very fast and efficient. There is no need to construct the target function compared with other learning algorithms. Instead, K-NN just simply stores all training examples and finds the relationships of a new instance with stored instances. Second, for each query instance, K-NN can construct a different approximation to the target function, which is very important when the target function is too complex to approximate in advance. Third, a K-NN algorithm, including a distance-weighted K-NN algorithm, is robust to noisy data, especially when there are sufficient training examples.

Because of its classification effectiveness, the K-NN algorithm has been applied to identify TMB proteins by [52, 59, 60]. Despite its advantages, K-NN algorithm also has its own drawbacks. First, all attributes are used to calculate the distances between instances. However, not all attributes are equally useful. This problem can be solved by weighting each attribute differently when calculating the distance, as can be seen from Eq. (2.4). Second, finding  $k$  nearest neighbors is very expensive when there is a large number of training examples. Many methods have been proposed to solve this problem at some memory cost, such as KD-trees [88, 89] and ball trees [90]. Third, standard Euclidian distance is not always the best distance measurement. Other distance measurements such as Pearson sample correlation distance, Mahalanobis distance or Kullback-Leibler divergence (KLD) distance can be applied under certain circumstances.

In this study, we propose a different version of the K-NN algorithm. The unique qualities of our K-NN method include the following. 1) Our algorithm is based on weighted Euclidian distance instead of standard Euclidian distance; 2) Instead of

considering only  $k$  nearest neighbors of the query instance, our algorithm finds  $k$  nearest training examples from each class to the query instance. For example, if there are  $n$  classes, our algorithm finds  $k$  nearest examples from each class to the query instance. In total, there are  $n * k$  training examples selected.

*2.2.4.1 Weighted Euclidian Distance.* For each test protein, the distance to train protein  $x$  of protein type  $T$  is calculated using

$$D_{tx} = \sqrt{\sum_i \frac{(t_i - x_i)^2}{\bar{x}_i}}, \quad (2.9)$$

where  $t_i$  is the  $i^{th}$  composition of the test protein,  $x_i$  is the  $i^{th}$  composition of the training protein  $x$ , and  $\bar{x}_i$  is the  $i^{th}$  average composition for all proteins of type  $T$ .  $T$  could be TMB proteins or non-TMB proteins, i.e., “Cytoplasmic” subgroup, “CytoplasmicMembrane” subgroup, “Cytoplasmic,CytoplasmicMembrane” subgroup, “Extracellular” subgroup, “Periplasmic” subgroup, or “Periplasmic, CytoplasmicMembrane” subgroup. Notice that  $\sqrt{\sum_i (t_i - x_i)^2}$  is the Euclidian distance between two proteins. Here, in the calculation of  $D_{tx}$ , each item within the summation is weighted by a factor of  $1/\bar{x}_i$ . Therefore,  $D_{tx}$  is referred to as *weighted Euclidean distance (WED)*.

*2.2.4.2 K-NN Algorithm for the Prediction of TMB Proteins.* For a test protein, WEDs to every TMB protein in the training set are calculated.  $K$  smallest distances are chosen. Let them be  $D_{imb-1}, D_{imb-2}, \dots, D_{imb-k}$ . The distance between the test protein and the TMB group is given by

$$\bar{D}_{imb} = \frac{1}{k} (D_{imb-1} + D_{imb-2} + \dots + D_{imb-k}). \quad (2.10)$$

The distances between the test protein and each of the six non-TMB subgroups are computed in a similar way. These distances are denoted as

$\bar{D}_{Cytoplasmic}$  (“Cytoplasmic”),  $\bar{D}_{CytoplasmicM}$  (“CytoplasmicMembrane”),  $\bar{D}_{CytoplasmicCM}$  (“Cytoplasmic, CytoplasmicMembrane”),  $\bar{D}_{Periplasmic}$  (“Periplasmic”),  $\bar{D}_{PeriplasmicCM}$  (“Periplasmic, CytoplasmicMembrane”) and  $\bar{D}_{Extracellular}$  (“Extracellular”). If  $\bar{D}_{mb}$  is less than all the other distances ( $\bar{D}_{Cytoplasmic}$ ,  $\bar{D}_{CytoplasmicM}$ ,  $\bar{D}_{CytoplasmicCM}$ ,  $\bar{D}_{Periplasmic}$ ,  $\bar{D}_{PeriplasmicCM}$  and  $\bar{D}_{Extracellular}$ ), then the test protein is predicted to be a TMB protein. Otherwise, it is predicted to be a non-TMB protein.

#### 2.2.4.3 Including Evolutionary Information to Calculate the Composition.

Evolution and mutation bring noise to biological data, making some proteins statistically different from the majority proteins of their own type. Information from homologous sequences has been proved to be useful in removing noise, which is helpful in solving many bioinformatics problems. In our model, for each protein, the BLAST program [26] was used to search for homologous sequences in the NCBI nonredundant database using threshold  $E=0.0001$ . The fifty best hits were chosen from the returned result. If less than 50 hits were returned, all of the hits were chosen. These sequences plus the query protein were used to calculate the residue composition and di-peptide composition for the query protein.

#### 2.2.5 Feature Selection

An automatic feature selection process was used to select the most useful attribute set. Different from methods such as a decision tree which selects a subset of attributes to

build the model, a standard K-NN algorithm uses all attributes to calculate the distances. However, not all features are equally useful for the identification of TMB proteins. Some features might be totally irrelevant. TMB-Hunt solves this problem by weighting each attribute differently. Additionally, the weight of each attribute can be optimized by genetic algorithm [59, 60]. In this study, a bi-direction feature selection process was applied to choose the most relevant features, i.e., residue composition or di-peptide composition. Ours is a greedy feature selection method that wraps the K-NN classifier in the feature selection process. This method is a simplified version of the *Best first* method included in [35] and was also used in another study [39].

This project's greedy feature selection algorithm started with a feature set that included 20 amino acids. Let  $n$  be the size of the feature set. Then  $n=20$  at the beginning. The algorithm can be divided into two stages: reduction and growth. In the reduction stage, the size of the feature set was gradually reduced. First, one amino acid was removed, and the composition of the remaining  $n-1$  amino acids were used to calculate WEDs. Five-fold cross-validation was used to evaluate the performance of the method. This step was repeated  $n$  times, so that every combination of  $n-1$  amino acids was tried. The combination that improved the performance most was chosen. Thus, the size of the feature set was reduced from  $n$  to  $n-1$ . This reduction process was continued until removing any amino acid from the feature set would reduce the performance. At the end of the reduction stage, we reached a feature set that included the composition of  $N$  amino acids ( $N \leq 20$ ).

Next, we used a growth stage to increase the size of the feature set by adding di-peptides. One di-peptide was added at a time, and the resulting feature set was used to calculate WEDs. Five-fold cross-validation was used to evaluate the performance of the method. This step was repeated 400 times, so that every di-peptide was tried. The di-peptide that yielded the greatest improvement in performance was chosen and added to the feature set. Thus, the size of the feature set was increased to  $N+1$ . This growth process continued until adding any more di-peptides would decrease the performance level.

### 2.2.6 Performance Measurements

Performance was measured using sensitivity, specificity, overall accuracy, and the Matthews correlation coefficient (MCC)

$$\text{Sensitivity} = \frac{TP}{TP + FN} \quad (2.11)$$

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (2.12)$$

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + TN + FP} \quad (2.13)$$

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FN)(TP + FP)(TN + FP)(TN + FN)}} \quad (2.14)$$

where  $TP$  is the number of true positives (i.e., the number of TMB proteins predicted as TMB);  $TN$  is the number of true negatives (i.e., the number of non-TMB proteins predicted as non-TMB);  $FN$  is the number of false negatives (i.e., the number of TMB proteins incorrectly predicted as non-TMB); and  $FP$  is the number of false positives (i.e., the number of non-TMB proteins incorrectly predicted as

TMB). Sensitivity is the measure of the percentage of TMB proteins correctly classified. Specificity is the percentage of non-TMB proteins correctly classified. Accuracy is the overall percentage of proteins correctly predicted. The Matthews correlation coefficient (MCC) measures the correlation between predictions and actual class labels, which is in the range of  $[-1, 1]$ , with 1 denoting perfect predictions and -1 denoting completely incorrect predictions. In a two-class classification, if the numbers of examples of the two classes are not equal, MCC is a better measure than accuracy [91].

## 2.3 Results

### 2.3.1 *The Proposed K-NN Method's Ability to Identify TMB Proteins*

We have developed a weighted Euclidian distance as the distance measurement in our K-NN algorithm. Five-fold cross-validations were used to evaluate performance. For each protein, only the protein itself was used to calculate residue composition. Twenty amino acids were used to calculate the weighted Euclidian distances. As can be seen from Table 1 (row 2), the proposed method can distinguish TMB proteins and non-TMB proteins with a 91.5% overall accuracy, with 0.633 MCC. 64.5% (sensitivity) of the TMB proteins, and 95.8% (specificity) of the non-TMB proteins correctly identified.

### 2.3.2 *Including Homologous Sequence Information Improves the Performance*

For each of query and training proteins, the BLAST program [26] was used to search for homologous sequences in the NCBI nonredundant database using threshold  $E=0.0001$ . At most, 50 best hits plus the protein itself were used to calculate residue

composition. Compositions of all 20 amino acids were used to calculate the weighted Euclidian distances. As can be seen from Table 1 (row 3), the prediction performance can be improved remarkably by including homologous sequence information. The prediction performance was increased to 94.4% overall accuracy, with 0.757 MCC, 74.6% sensitivity and 97.6% specificity.

### *2.3.3 Further Improvement of the Prediction Performance by Feature Selection*

We tried to include both residue compositions and di-peptide compositions to calculate the WED. However, not all 20+400 features are useful for prediction. After the greedy feature selection process, we obtained a much smaller feature subset, which included 19 residues {A, C, D, E, F, G, H, I, K, L, M, P, Q, R, S, T, V, W, Y} and 24 di-peptides {AI, CC, DP, II, IT, KA, LF, LG, LI, LK, LT, MV, NE, NH, QY, RK, SP, SY, VR, WE, WH, WN, YK, YR }. Using selected features, the prediction performance was further improved to 97.1% accuracy, 0.876 MCC, 86.4% sensitivity and 98.8% specificity, which can be seen in Table 1 (row 4).

### *2.3.4 Comparison with Predictions Solely Based on Similarity Search*

Similarity searches have been widely used to infer protein functions. If two proteins are highly similar in a sequence, then they might share similar functions, structures or evolutionary origin. For each test protein, we conducted a homologous search on the training set using the BLAST program [26]. The test protein was then predicted to be the protein type (i.e., TMB or non-TMB) of the most homologous protein. Using the same dataset partition and five-fold cross-valuation, the similarity search only

Table 1. Comparison of the Proposed K-NN Method with A Similarity Search.

	Accuracy	MCC	Sensitivity	Specificity
<i>K-NN</i> (Single sequence + 20 residues)	91.5%	0.633	64.5%	95.8%
<i>K-NN</i> (Homologous sequences + 20 residues)	94.4%	0.757	74.6%	97.6%
<i>K-NN</i> (Homologous sequences + 19 residues + 24 di-peptides)	97.1%	0.876	86.4%	98.8%
Similarity search	75.4%	0.439	86.4%	73.6%

achieved 75.4% accuracy with 0.439 MCC, 86.4% sensitivity and 73.6% specificity, which was much lower than the proposed K-NN method.

### 2.3.5 Comparison with Other Prediction Methods

We also compared our method with other state-of-the-art prediction methods, such as TMH-Hunt, BOMP, PRED-TMBB and PROFtmb. All these methods provide web servers, which makes it easy to compare them with our method. Since the number of positive and negative samples is not balanced, MCC was used as the primary measurement of the prediction performance. Other measurements, such as accuracy, sensitivity and specificity, are also reported.

The datasets used in this study, i.e., both TMB and non-TMB proteins, were submitted to the servers of TMH-Hunt, BOMP, PRED-TMBB and PROFtmb. Table 2 shows the prediction results of all five methods.

Table 2 (row 2) shows the prediction performance of the proposed K-NN method. Homologous sequence information was included to calculate the composition of selected residues and di-peptides. A weighted Euclidian distance is used as distance measurement.

Table 2 (row 3) shows the prediction performance of BOMP [52]. A Blast search option was chosen to ensure highest performance, as mentioned in Section 2.1.3.2.

Table 2 (row 4) gives the prediction results of TMB-Hunt [59, 60]. Evolutionary information was used, which ensures the best performance of the method, as mentioned in Section 2.1.3.1.

Table 2 (row 5) gives the prediction performance of PRED-TMBB [53, 76] using posterior decoding. Three decoding methods are provided on the web server, while the posterior decoding was reported to achieve the best performance.

Table 2. Comparison of Different Methods.

Method	Accuracy	MCC	Sensitivity	Specificity
K-NN (Homologous sequences +19 residues + 24 di-peptides)	97.1%	0.876	86.4%	98.8%
BOMP (with BLAST search)	95.0%	0.787	79.8%	97.4%
TMB-Hunt (with evolutionary information)	93.7%	0.747	81.5%	95.7%
PRED-TMBB	64.3%	0.342	89.1%	60.4%
PROFtmb	92.6%	0.684	71.4%	96.0%

Table 2 (row 6) shows the prediction performance of PROFtmb [20]. The server provides two options:  $Z \geq 10$  and  $Z \geq 6$ . We tried both values. PROFtmb achieved better performance with  $Z \geq 6$  as the threshold. The results reported here were achieved with  $Z \geq 6$ .

As can be seen from Table 2, the proposed K-NN algorithm outperformed all other methods in both MCC and accuracy. It is worthwhile to point out that the datasets used in the current study are likely to have a big overlap with the datasets that were used to train BOMP, TMB-Hunt, PRED-TMBB, and PROFtmb servers. Thus, when we evaluated these methods by submitting our datasets to their web servers, the performance of these methods might have been overestimated. In contrast, our K-NN method was evaluated using a five-fold cross-validation such that any protein in the training set and any protein in the test set shared less than 25% identity. Remarkably, our method still outperformed the others under this condition.

Another virtue of the proposed K-NN method is its speed. No parameters need to be tuned. The training and prediction processes are simple and efficient. The calculation of residue and di-peptide composition is fast and straightforward. Thus, our method can be applied to scan the whole dataset of gram-negative proteomes for possible TMB proteins. Among the five methods compared, PRED-TMBB and PROFtmb achieved relatively lower performances. However, the major purpose of PRED-TMBB and PROFtmb is to predict the topologies of TMB proteins. The relatively low prediction performances of these methods in the classification of TMB protein can be explained by the following facts: 1) the training datasets only contain positive examples, and there

were no negative models; 2) the training sets are relatively small since only very few TMB proteins have topology information which is experimentally determined.

### 2.3.6 *A Web Server for the Prediction of TMB Proteins*

A web server (Figure 1) based on the proposed method was developed.<sup>7</sup> It allows users to submit their protein sequences to identify candidate TMB proteins. The server can run in two modes: not using homologous sequences or using homologous sequences. The method runs faster when homologous sequences are not used. But it achieves more accurate predictions when homologous sequences are used. When homologous sequences are used, users can either upload the homologous sequences for their input sequences or let the server do a BLAST search. Detailed instructions for users are also available on the server.

### 2.3.7 *Genome Scan*

The proteomes of 11 gram-negative bacteria<sup>8</sup> were scanned using our server. We chose homologous sequences as our model. The details of the predictions are available at.<sup>9</sup> The statistics of the predictions of all 11 gram-negative bacterial proteomes are in Table 3.

We analyzed the predictions on the proteome of *Escherichia coli* in details, since this proteome is relatively well studied compared with the others. The *Escherichia coli* proteome consists of 4319 proteins, among which 144 (3.33%) were predicted to be TMB

---

<sup>7</sup> The web server is available at <http://yanbioinformatics.cs.usu.edu:8080/TMBKNNsubmit>.

<sup>8</sup> Proteins of 11 gram-negative bacterial proteomes are available at <http://ca.expasy.org/sprot/hamap>.

<sup>9</sup> <http://yanbioinformatics.cs.usu.edu:8080/TMBKNNsubmit>.

**Running mode?**

- A. Not using homologous sequences ([example](#)). Proteins are input in the FASTA format.
- B. Homologous sequences are input by the user ([example](#)). Each protein is followed by at most 50 homologous sequences.
- C. Homologous sequences will be found using BLAST search ([example](#)). This option only allows one protein in an input file. Warning: It may take a long time (~3 mins) to complete. Please be patient!

Please set the E-value for the blast search:

**Paste your protein sequences here:**

Or upload your file from:

C:\Documents and Settings\ Browse...

*When you submit files, please restrict your file sizes to at most 10 MB.*

Clear All Data Run TMB prediction!

(A)

```
>O52158
-----TMB
>O87343
-----TMB
>gij114054
-----Non-TMB
>gij1170436
-----Non-TMB
>P06129
-----TMB
>P72121
-----TMB
```

---

There are 6 test proteins!  
4 are found to be **TMBs**!  
2 are **non-TMBs**

(B)

Figure 1. TMB proteins prediction web server. A: The Input form. B: The output form.

Table 3. Prediction Results of 11 Gram-Bacteria Proteomes.

Proteome	Total number of proteins	Number of predicted TMB proteins	Hits
<i>Bordetella pertussis</i>	3261	118	3.6%
<i>Caulobacter crescentus</i>	3718	186	5.0%
<i>Chlamydia pneumoniae</i>	1052	22	2.1%
<i>Escherichia coli</i> O157:H7	5271	206	3.9%
<i>Escherichia coli</i>	4319	144	3.3%
<i>Fusobacterium nucleatum</i>	2046	84	4.1%
<i>Haemophilus influenzae</i>	1710	53	3.1%
<i>Helicobacter pylori</i> (Campylobacter pylori)	1553	89	5.7%
<i>Pasteurella multocida</i>	2015	52	2.6%
<i>Pseudomonas aeruginosa</i>	5558	248	4.5%
<i>Salmonella typhimurium</i>	4531	147	3.2%

proteins by the proposed K-NN method. Of these 144 hits, 12 are found in the dataset which was used to train the server, 49 proteins are annotated as “outer membrane proteins” in Swiss-Prot, and 15 proteins share very high similarity with some TMB proteins in the training dataset ( $E < 0.0001$  in BLAST search). Thus, we have a high confidence level that these 76 proteins are true positives. Besides these true positives, 22 proteins are annotated with “membrane,” “cell membrane” and “multi-pass membrane protein” in Swiss-Prot. Only 1 of these 22 proteins was predicted to be transmembrane  $\alpha$ -helical proteins (inner membrane proteins) by TMHMM [19] and PSORTb [86]. Thus, most of these proteins are likely TMB proteins. For the remaining 46 proteins, 27 proteins were annotated with subcellular locations other than outer membranes. Thus,

these 27 proteins are false positives. The remaining 19 proteins may suggest new TMB proteins that have not been previously discovered. TMB proteins are secreted across the inner membrane by virtue of a signal peptide that is cleaved by signal peptidase I (SPaseI) [92]. Thus, the existence of a SPaseI-cleaved signal peptide is a characteristic of TMB proteins. Lipoproteins also reside on the outer membrane. One difference between lipoproteins and TMB proteins is that lipoproteins contain a signal peptide (referred to as *lipoprotein signal peptide*) that is cleaved by lipoprotein signal peptidase (Lsp) [93]. We submitted the 144 predicted TMB proteins to LipoP [93], a computational server that discriminates SPaseI-cleaved signal peptides and lipoprotein signal peptides. One-hundred-eight of them were predicted to contain a SPaseI-cleaved signal peptide. In the remaining proteins, one was predicted to contain lipoprotein signal peptide.

We also compared our method's predictions with the proteome scanning results obtained by BOMP [52]. We chose BOMP for comparison for two reasons: 1) Table 2 shows that BOMP achieves better performance than TMB-Hunt, PRED-TMBB, and PROFtmb; and 2) BOMP's predictions of *Escherichia coli* proteins are available on its server. In the *E. Coli* proteome, BOMP predicted 103 TMB proteins. Comparisons show that 73 proteins were predicted to be TMB by both our K-NN method and BOMP. Seventy-one proteins were predicted to be TMB by our K-NN method but not by BOMP. Among them, 20 proteins are true positives. Besides these, 18 proteins were annotated with "membrane," "Cell membrane" or "multi-pass membrane protein" in Swiss-Prot. When TMHMM [19] and PSORTb [86] were used to scan these proteins, only 1 was predicted to be transmembrane  $\alpha$ -helical proteins (inner membrane protein). Thus, most

of them are likely TMB proteins. Thirty proteins are predicted to be TMB proteins by BOMP but not by K-NN. Among them, only 12 are true positives. This comparison shows that there is a big overlap between the predictions of the K-NN method and BOMP. It also shows that each of the two methods can identify some TMB proteins missed by the other. This suggests the possibility of achieving better performance by combining these two methods.

## 2.4 Discussion

In this study, we propose a K-NN method that can identify TMB proteins with high accuracy. The originality of our method lies in the following aspects.

1. Instead of a standard Euclidian distance, a weighted Euclidian distance was used as the distance measurement in the proposed K-NN algorithm. Compared with the standard Euclidian distance, the weighted Euclidian distance is a better measurement to evaluate the relationship between proteins. For example, consider the same amount of standard Euclidian distance of 0.01. The difference between  $x_i$  and  $y_i$ , when  $x_i$  is 0.9 and  $y_i$  is 0.89, is less significant than the difference when  $x_i$  is 0.11 and  $y_i$  is 0.10. By assigning a weight of  $1/\bar{x}_i$  to the term, significant differences between  $x_i$  and  $y_i$  will return larger values in the WED.
2. Homologous sequences were included to calculate the residue and di-peptide compositions. By including evolutionary information, prediction performance was improved dramatically.

3. An automatic feature selection process was employed to choose the most relevant residues and di-peptides to calculate the WED, which further improved the prediction performance.
4. Negative proteins (non-TMB proteins) were divided into six subgroups based on their subcellular localizations.
5. In the standard K-NN algorithm, only  $k$ -nearest proteins close to the test protein were chosen. The test protein was then assigned to the most common class value among its  $k$ -nearest training examples. However, the K-NN algorithm proposed in this study finds  $k$ -nearest proteins from each subgroup. In total, there are  $k * n$  proteins considered, where  $n$  is the number of subgroups (i.e.,  $n$  is 7 in this example, which includes 1 TMB group and 6 non-TMB subgroups). For each test protein, the WEDs to every protein in each subgroup were calculated.  $K$ -smallest distances were chosen and averaged. The test protein was then predicted to be the TMB proteins if its average distance to TMB group was the smallest. Otherwise, it was predicted to be non-TMB. It was proved that better prediction performance was achieved via this modification (results not shown here).

We have applied the proposed method to scan the gram-negative proteomes for possible TMB proteins and a set of candidate proteins were prioritized for molecular biologists.

## 2.5 Conclusion

In summary, we have developed a K-NN method that can identify TMB proteins with high accuracy. The method uses a weighted Euclidian distance (WED) as distance measurement, which was calculated using compositions of certain residues and di-peptides chosen by a systematic feature selection process. Due to its speed and simplicity, the proposed can be applied to discover TMB proteins on a proteomic scale.

## 2.6 Future Work

Besides residues and di-peptides, other features such as pseudo-amino acids may provide more useful information in discovering TMB proteins. Unlike residue composition, the pseudo amino acid composition consists of  $20+\lambda+\mu$  discrete numbers, of which the first 20 are the same as the standard amino acid compositions, and the remainder represent  $\lambda+\mu$  ranks of sequence-order correlation factors [33, 94, 95]. Suppose a protein  $X$  with a sequence of  $L$  amino acid residues:  $R_1 R_2 R_3 R_4 \dots R_L$ , where  $R_l$  represents the residue at sequence position  $l$ ,  $R_2$  the residue at position 2, and so on. The  $\lambda$  sequence-order-correlated factors are given by

$$\delta_i = \frac{1}{L-i} \sum_{j=1}^{L-i} \Delta_{j,j+i}, \quad (2.15)$$

where  $i=1,2,3 \dots \lambda$ ,  $\lambda < L$ , and  $\Delta_{j,j+i} = \Delta(R_j, R_{j+i})=1$  if  $R_j = R_{j+i}$ , else 0. The second set of  $\mu$  factors are given by

$$h_i = \frac{1}{L-i} \sum_{j=1}^{L-i} H_{j,j+i}, \quad (2.16)$$

where  $i=1,2,3\dots\mu$ ,  $\mu < L$ ,  $H_{j,j+i} = H(R_j) * H(R_{j+i})$  and  $H(R_j)$  and  $H(R_{j+i})$  are the standard converted hydrophobicity values of  $R_j$  and  $R_{j+i}$ . As can be seen, pseudo amino acid compositions contain more sequence-order and biochemical information than an n-peptide does, while the dimension only increases by linear order. By including certain pseudo amino acid compositions to calculate WED, the prediction accuracy could be further improved.

## CHAPTER 3

IDENTIFICATION OF DELETERIOUS NONSYNONYMOUS SINGLE  
NUCLEOTIDE POLYMORPHISMS<sup>10</sup>**3.1 Background***3.1.1 Neutral or Deleterious Nonsynonymous  
Nucleotide Polymorphisms*

It is estimated that around 90% of human genetic variations are differences in single bases of DNA, known as single nucleotide polymorphisms (SNPs). Single nucleotide polymorphism (SNP) is a single nucleotide substitution in the genome due to evolution or mutation. Depending on where the variations occur and the variations themselves, SNPs may result in different biological effects. SNPs happening at coding regions of genes or in regulatory regions are more likely to lead to a different polypeptide sequence produced, causing functional differences than SNPs in intergenic regions [96]. These SNPs, called nonsynonymous single nucleotide polymorphisms (nsSNPs), also known as single amino acid polymorphisms (SAPs) that cause amino acid changes in proteins, have the potential to affect both protein structures and functions [97]. While most of the mutations in SAP sites are not associated with any changes in phenotype and are considered functional neutral, others may bring deleterious effects to protein functions and are responsible for many human genetic diseases, such as sickle cell anemia, diabetes, and various cancers [98, 99, 100, 101]. Such deleterious SAPs may result in producing completely malformed proteins that are unable to carry out their original functional roles. The large number of SAPs present in human genomes (i.e.,

---

<sup>10</sup> Co-authored by Hu, J. and Yan, C.

around 67,000~ 20,000) calls for reliable prediction methods for the automatic annotation of disease-related SAPs.

### *3.1.2 Current Methods for the Identification of Deleterious Nonsynonymous Single Nucleotide Polymorphisms*

Recent years have seen an explosion in the number of SAPs in public databases, such as dbSNP [102], HGVBASE [103], and Swiss-Prot [104]. The large size of these databases presents a challenging hurdle for annotating the effects of all SAPs experimentally. Therefore, prediction methods that can identify disease-related SAPs are in high demand. Computational methods may not be 100% accurate, but they can prioritize a much smaller number of candidate deleterious SAPs for future analysis.

Multiple prediction methods have been proposed to classify SAPs based on the attributes of SAP sites. Sequence and structural information around substitution sites have been proven to be useful in the prediction of SAP effects. In addition, disease-related SAPs tend to be buried and occur at highly structural- and sequence-conserved regions.

*3.1.2.1 Earliest Attempts.* Cargill et al. [105] tried to classify the effects of SAPs by using an amino acid substitution scoring matrix, BLOSUM62 [106]. The basic assumption behind this method is that bigger biochemical differences between wild and mutant allele indicate higher chances of an SAP being disease related. However, this method does not consider the information specific to the protein of interest and the physicochemical environment around the SAP positions. Moreover, the substitution

matrix was originally designed for sequence alignment, which alone is not suitable for the prediction of SAPs.

*3.1.2.2 Empirical Rules and Probabilistic Approaches.* Empirical rules have been employed to predict the effect of SAPs. Most of these methods are based on the assumption that important amino acids exist at conserved regions, and mutations at well-conserved regions tend to have damaging effects. Herrgard et al. [107] developed a method focusing on active sites to predict the effects of SAPs on enzyme catalytic activity. Three-dimensional sequence profiles surrounding active sites were computationally derived, and were then used to analyze the effects of SAPs by considering three key features, proximity of SAP position to the active site, degree of amino acid conservation at the position in related proteins, and compatibility of the SAP mutation with residues observed at that position in similar proteins. The authors found that changes at key active sites and highly conserved positions are more likely to have deleterious effects on the catalytic activity, and nonconservative SAP mutations at highly conserved residues are even more likely to be disease related. Probably one of the most well-known SAP effect prediction methods is SIFT<sup>11</sup> [98, 99, 100], which utilizes sequence homology to predict whether an amino acid substitution is deleterious. The method is based on sequence conservation and position-specific scores. Given a protein sequence, SIFT first finds its related proteins and obtains an alignment of homologous proteins with the protein of interest. Then, based on the position of SAPs, SIFT calculates the probability that an amino acid at a position is tolerated conditional on the most frequent amino acid being tolerated. The SAP is predicted to be deleterious if this

---

<sup>11</sup> The online web server of SIFT is available at <http://blocks.fhcrc.org/sift/SIFT.html>.

normalized value is above a user-defined threshold. It is worthwhile to point out that SIFT only uses sequence information to predict the effect of SAPs, making it applicable to the whole proteome. SIFT has been tested on the human variant databases and was proved to be able to find disease-related SAPs. Other methods have applied probabilistic approaches to predict whether the SAP is deleterious. For example, in the study of [108], a set of structure and sequence-based features were derived by using a structural model and phylogenetic information. Then, the feature set was integrated into a probabilistic assessment to indicate the effects of SAPs for the query proteins.

*3.1.2.3 Machine Learning Approaches.* Machine learning techniques, such as Bayesian network, decision tree, random forest, neural networks, support vector machines and hidden Markov models have been widely applied to identify deleterious SAPs.

Bayesian network has been applied by [109] to predict the effect of mutations on protein function. The strength of the method lies in its ability to handle incomplete data and to encode relationships between variables. First, it can handle situations where there is missing information, e.g., missing structural information, missing evolutionary information. Second, it is tolerant to incomplete training data.

A decision tree has the benefit of generating human interpretable rules; therefore, it is very useful for discovering the mechanism of deleterious SAP. Dobson et al. [28] applied a decision tree to find deleterious SAPs over the dataset of Ensembl human genome protein sequences. For each SAP position, the authors extracted a total of 17 features to build the classifier, of which 11 are nonstructurally dependent features and 6

are structurally dependant features. Nonstructural features include the following: residue types and physicochemical properties of both wild-type and mutant-type residue; conservation score of SAP position; changes of accepted point mutations (PAM) scores measured from PAM120 matrix [110]; change of side chain volume, mass, and hydrophobicity between wild-type and mutant-type residues, and other attributes extracted from the Swiss-Prot feature table; pathway information; and finally gene ontology classifications and interacting regions. Structural information of each SAP site includes the following: secondary structure conformation, relative solvent accessibility, normalized relative accessibility, exposure (i.e., relative accessibility in states of buried, exposure or intermediate), buried charge, and interacting regions mapped by the MMDBBIND database [111]. The authors of [28] first applied the 1R algorithm and information gain to rank single attributes, and identified the conservation score as the most discriminative feature, which matches with previous research findings. Other features, such as normalized relative accessibility, mass change, PAM score, and relative solvent accessibility, were also among the top useful features. After experimenting with different sets of attributes, they found that a decision tree using all attributes on a balanced dataset achieved best results. Krishnan and Westhead [112] also applied the decision tree method on mutagenesis dataset of the lac repressor [113, 114] and generalized a set of classification rules guiding the prediction of SAP effects.

Random forest (RF) is a classifier consisting of an ensemble of tree-structured classifiers [115]. It combines bagging (i.e., bootstrap aggregating) and random feature selection techniques. It then outputs the class value that is the mode of the classes

predicted by each individual decision tree. Recently, a random forest-based method nsSNPAnalyzer [116, 117] was developed. It was applied to discover deleterious SAPs from a human variant SAP dataset. The results indicate that nsSNPAnalyzer achieved better performances than SIFT [98, 99, 100] on that dataset.

Neural network approaches has also been applied to predict the effects of SAPs. Ferrer-Costa et al. [118] applied a three-layer, feed-forward neural network model using encoded parameters derived from three main categories: 1) structure-based descriptors, such as secondary structure and solvent accessibility; 2) residue/sequence properties, such as hydrophobicity, secondary structure propensity, volume, probability of each residue at SAP position, and changes of Blosum62 [106] and PAM40 [110] scores between wild-type and mutant-type; 3) properties derived from the multiple sequence alignment. Notice that all the structural features involved in this method were predicted from sequence information. The method was tested against human-mouse homolog datasets. Results indicate that prediction method developed for one organism can still be used to predict the SAP effects of other homologous organisms. SNAP [119, 120] was another recent method based on a neural network. SNAP only utilizes information derived from protein sequences; therefore, it can be applied to predict large mutation databases. SNAP considers the immediate local sequence environment of SAP sites by using symmetric windows around SAP sites. Based on previous studies and experiences, a set of protein features were included for evaluation. These features include biochemical properties (i.e., hydrophobicity, charge, and size of residues), sequence information, transition frequencies, PSI-BLAST [26] profiles, position-specific independent counts

(PSIC), predicted 1-dimension structures (i.e., relative solvent accessibility) of wild-type residues and their changes, predicted flexibility, protein family information, and Swiss-Prot annotations (i.e., active residues, bonding residues, posttranslational modification residues, variable residues and transmembrane region). These features were encoded to train the neural network. The method was evaluated on a dataset obtained from the Protein Mutant Database [121], which is based on experimental amino acid substitutions, and a 78% accuracy was reported. SNAP was claimed to outperform most other previous developed methods.

Due to their high generalization abilities, support vector machines (SVM) have been widely applied to predict deleterious SAP by many research groups [38, 112, 116, 122, 123, 124]. Among them, SAPRED<sup>12</sup> [38] is a recently published state-of-the-art method, which achieves the best performance. The authors of [38] investigated a large set of structural- and sequence-based features, which include both commonly used features and novel ones introduced in this study. Most features were calculated from both the wild-type and the variant proteins, and differences between wild-type and variant were also calculated. Commonly used features from previous studies include the following: residue frequency (wild-type residue frequency, variant-type residue frequency, and the difference between them) of the SAP site, conservation scores of the SAP site and its three neighboring positions on both sides, the secondary structure of the wild-type SAP, differences of hydrogen, and disulfide bond between wild-type and mutant-type, indication of whether the SAP is in disordered region. Some of the very important new features about SAP sites are structural neighbor profiles around the SAP (i.e., a 20-

---

<sup>12</sup> SAPRED is available at <http://sapred.cbi.pku.edu/cn/supp.do>.

dimensional vector of counts of each residues found in the 3-dimensional vicinity of SAP sites), sequence and spatial distances between SAP sites and their closest functional sites (i.e., active site, binding site, metal ion binding site, posttranslationally modified residue, disulfide bond, and transmembrane region), difference of structure model energy between wild-type and variant proteins, indication of whether a SAP is located in disordered regions, indication if a protein belongs to the histocompatibility leukocyte antigen (HLA) family, etc. After analyzing different types of features, the authors found that the most predictive features are residue frequencies, structural neighbor profiles, conservation scores, distance to nearby functional sites, and solvent accessibilities of each SAP site. Conforming to previous studies, residue frequency showed the highest predictive power. Conservation scores and solvent accessibilities were also ranked as the top predictive features. However, the authors found that two new features (i.e., structural neighbor profiles and distance to nearby functional sites) showed better predictive power than solvent accessibility. In fact, the structural neighbor profile alone is nearly as powerful as residue frequencies, which confirms the previous findings that microenvironments around SAP sites are very important. By applying all the structural and sequence-based features to build the SVM, SAPRED achieved an 82.6% overall accuracy and 0.604 MCC on a dataset of observed human alleles collected from variant pages of the Swiss-Prot knowledgebase. SAPRED\_SEQ, another version of the method, was also provided to predict proteins which have no experimentally determined 3-dimensional structures. SAPRED\_SEQ only requires sequence-derived attributes as input. It achieved an 81.5% overall accuracy with 0.577 MCC on the same dataset. Recently, Ju et al. [125] applied a

multi-scale RBF kernel fuzzy support vector machine to discriminate disease-related nsSNPs. The results show that it outperforms the traditional SVM method.

PANTHER<sup>13</sup> [126, 127] is a method based on the hidden Markov model (HMM). It relates protein sequence and function relationships in a robust and accurate way. It contains two parts: PANTHER library (PANTHER/LIB) and PANTHER index (PANTHER/X). PANTHER/LIB is a collection of protein families and subfamilies. Each family/subfamily is represented as a multiple sequence alignment, an HMM, and a family tree. PANTHER can be applied to identify deleterious SAPs on a database-wide scale. For a SAP allele on a protein sequence, PANTHER first maps the protein to the largest sub-tree on the family tree. Suppose there is a substitution of amino  $a$  by  $b$  at position  $i$ , PANTHER first calculates the possibilities of wild-type and mutant-type residues at that position, using HMM scores for that particular subfamily. The likelihood of a single amino acid at a particular position is calculated as

$$aaPSEC(a, i, j) = \ln[P_{aij} / \max(P_{ij})] \quad (3.1)$$

where  $P_{aij}$  is the probability of amino acid  $a$  at position  $i$  in HMM  $j$ , and  $\max(P_{ij})$  is the maximum probabilities of all amino acids at position  $i$  of HMM  $j$ . Suppose wild-type residue  $a$  is substituted by mutant-type residue  $b$  at the SAP site, the substitution score is calculated using

$$subPSEC(a, b, i, j) = - |aaPSEC(a, i, j) - aaPSEC(b, i, j)| = - | \ln(P_{aij} / P_{bij}) | . \quad (3.2)$$

The more negative the score, the higher the chance the SAP is deleterious.

Recently, PANTHER was further improved by introducing another parameter  $n_{ic}$ , an

---

<sup>13</sup> PANTHER is available at <http://www.pantherdb.org/tools/csnpscoreForm.jsp>.

independent count measuring the (global) diversity of sequences over which a position has been conserved. Now, the probability of a SAP being deleterious ( $P_{deleterious}$ ) as a function of  $subPSEC$  is given by

$$P_{deleterious} = 1 - \frac{\exp(subPSEC + 3.00)}{1 + \exp(subPSEC + 3.00)}, \quad (3.3)$$

in which  $subPSEC = -0.88 \ln P_{aj} + 0.89 \ln P_{bj} - 0.94 \ln n_{ic}$ , where  $P_{aj}$  is the larger and  $P_{bj}$  is the smaller of the wild-type and mutant-type residues.  $P_{deleterious}$  gives the possibility of SAP being deleterious. PANTHER only needs protein sequence information, and it has been applied to find deleterious SAPs from two databases: 1) Human Gene Mutation Database (HGMD) (i.e., a curated database of mutations in human genes, most of which are related to disease) [97, 128]; and 2) dsSNP database (i.e., a database of human gene variations, most of which are collected randomly) [102]. PANTHER can recognize 40% of the deleterious SAPs from dsSNPs and 76% from HGMD.

### 3.1.3 Motivation of This Study

There are several limitations to most current methods. First, most methods make prediction based on both structural- and sequence-related information around SAP sites. However, many proteins do not have 3-dimensional structural information, which restricts the application range of such methods. Second, all previous methods only consider a small set of arbitrarily chosen features. Third, no feature dependencies and redundancies are analyzed. Some research groups have evaluated the prediction power of each single attribute individually. For example, 1R algorithm and information gain were

used to identify the best single attributes by [28]. Ye et al. [38] built SVMs using each individual feature and ranked the prediction power of different features based on their prediction accuracy. All these methods have one common disadvantage. Each feature is considered separately, and feature dependency and redundancy are ignored, which may lead to worse classification performances when top ranked features are combined together. A more systematic analysis is needed to identify features that play vital roles in determining the effects of SAPs. Some features might not provide top discrimination power, but significant classification capabilities can be achieved by combining with other features. In addition, the same features can display different prediction power in different computational models.

In this study, we explored the feasibility of classifying SAPs into disease-causing and neutral mutations using only information derived from a protein sequence. From a protein sequence, we extracted a set of 686 features describing the difference between the wild-type residue and mutant-type residue. Then a greedy search process was employed to select the features that were useful for the classification of SAPs. Using ten selected features, a decision tree-based method was capable of detecting disease-related SAPs.

## **3.2 Materials and Methods**

### *3.2.1 Datasets*

There are two datasets used in this study. The first dataset was obtained from a recent study of Ye et al. [38]. It was collected from the variant pages of the Swiss-Prot knowledgebase. It has 3438 SAPs found in 522 proteins, including 2249 “Disease” and

1189 “Polymorphism” (“neutral”) SAPs. This dataset is named as Ye’s dataset, and it was used primarily for the training and development of the method. The second dataset<sup>14</sup> is the Humvar dataset from the PhD\_SNP server [122], which contains all the SAPs from the Swiss-Prot variant database. It has 12944 “Disease” and 8241 “Polymorphism” SAPs. This dataset is named as Swiss-Prot dataset, and it was used primarily for the evaluation of the method.

### 3.2.2 Feature Set

Our purpose is to identify deleterious SAPs from sequence information. Therefore, only sequence based features were extracted. Some features describe the biochemical or biophysical environment of SAP sites. For most features, however, the difference between mutant and wild type residues were calculated.

*3.2.2.1 Sequence Features Used in Previous Studies.* For each SAP site, Ye et al. [38] extracted 60 different features, among which 19 were derived from sequence information, as can be seen from Table 4 (feature No. 1-19). Residue frequency has been shown to be very useful in the analysis of SAP effects. Attributes *wt\_seq* and *mt\_seq* are the observed frequencies of wild-type residue and mutant-type residue of the SAP site. The residue frequency difference (*diff\_freq*) was then calculated using

$$diff\_freq = mt\_freq - wt\_freq \quad (3.4).$$

Residue frequencies are calculated based on the multiple alignment of homologous sequences [98, 99, 100]. For each query protein, PSI-BLAST [26] with parameter  $-e$  0.0001 and  $-h$  0.002 is run for four iterations to collect a pool of

---

<sup>14</sup> This dataset is available online at <http://gpcr2.biocomp.unibo.it/~emidio/PhD-SNP/HumVar.txt>.

homologous sequences from the NCBI nonredundant database. From the resulting position-specific scoring matrix (PSSM), residue frequencies of each SAP position can be extracted, including both wild-type residue and mutant-type residue. In this study, we introduced another feature (*nor\_diff\_freq*) by normalizing the frequency difference using

$$nor\_diff\_freq = \frac{mt\_freq - wt\_freq}{wt\_freq} \quad (3.5).$$

The benefit of introducing the normalized frequency difference is: for the same amount of absolute frequency difference between wild-type residue and mutant-type residue, *nor\_diff\_freq* can better model the relative change than *diff\_freq*. Often SAP is more likely to be deleterious if the absolute value of *nor\_diff\_freq* is larger.

Conserved regions (i.e., both structure- and sequence-conserved regions) are usually functionally important, and mutations at well-conserved regions tend to have damaging effects on protein functions and structures. Since the purpose is to identify deleterious SAPs from sequence information, the structure conservation score is discarded. Sequence conservation scores of the SAP position (i.e., *conserve*) as well as three positions to its left (i.e., *neibor3L*, *neibor2L*, and *neibor1L*) and 3 positions to its right (i.e., *neibor1R*, *neibor2R*, and *neibor3R*) are calculated to measure the level of conservation. As in [38, 129], the conservation score of a position is defined as the information content of the amino acid frequency distribution at this position in a multiple sequence alignment after PSI-BLAST [26] search, and it is calculated using

$$Conservation = -\sum_{i=1}^{20} p_i \log_2 p_i \quad (3.6).$$

Table 4. All Sequence-Based Features of SAP Sites.

No.	Feature name	Description
1	wt_seq	The wild type residue frequency.
2	mt_seq	The mutant residue frequency.
3	diff_freq	The difference between variant residue frequency and wild type residue frequency.
4	blosum	The BLOSUM62 score of the SAP substitution.
5	grantham	The GRANTHAM score of the SAP substitution.
6	neibor3L	The conservation score of the 3rd left residue from the SAP site.
7	Neibor2L	The conservation score of the 2nd left residue from the SAP site.
8	Neibor1L	The conservation score of the 1st left residue from the SAP site
9	conserv	The conservation score of the SAP site
10	neibor1R	The conservation score of the 1st right residue from the SAP site.
11	neibor2R	The conservation score of the 2nd right residue from the SAP site.
12	neibor3R	The conservation score of the 3rd right residue from the SAP site.
13	act_seq_neibor	The sequence distance between the SAP site and its nearest residue holding the functional site with Feature Key of ACT_SITE.
14	binding_seq_neibor	The sequence distance between the SAP site and its nearest residue holding the functional site with Feature Key of BINDING.
15	Metal_seq_neibor	The sequence distance between the SAP site and its nearest residue holding the functional site with Feature Key of METAL.
16	modres_seq_neibor	The sequence distance between the SAP site and its nearest residue holding the functional site with Feature Key of MOD_RES.
17	transmem	Whether the SAP site is in the transmembrane region.
18	In_disorder	Whether the SAP site is in the disordered region.
19	is_HLA	Whether the protein containing the SAP belongs to HLA family.
20	Nor_diff_freq	Normalized difference between mutant-type residue frequency and wild-type residue frequency.
21-551	$f_i$	For AAindex type 1, $f_i = \frac{index_i(mut) - index_i(wildtype)}{index_i(wildtype)}$ .
552-686	$f_i$	For AAindex type 2, $f_i = index_i$ .

where  $p_i$  is the frequency of residue type  $i$  at the interested position. The conservation score ranges from 0 to 4.32. The smaller the score, the more conservative the position is.

The substitution matrix describes the rate at which one amino acid changes to another amino acid through evolution. Though originally used for sequence alignment, a substitution matrix such as Blosum62 [106] is also widely used to measure the difference between wild-type residue and mutant residue (i.e., *blosum*) [38, 118]. The Grantham matrix [130] predicts the effect of substitutions between amino acids based on chemical properties, including polarity and molecular volume. This feature (i.e., *grantham*) was also used in previous studies [38, 131] in the prediction of SAP effect.

Empirically, mutations happening around functional sites are more likely to be disease associated. The authors of [38] calculated a group of features to measure the sequence distance between a SAP position and its closest functional sites, i.e., active site (i.e., *act\_seq\_neibor*), binding site (i.e., *binding\_seq\_neibor*), metal ion binding site (i.e., *metal\_seq\_neibor*) and posttranslationally modified residue (i.e., *modres\_seq\_neibo*). Other features, such as *transmem* and *in\_disorder*, were used to indicate if the SAP position is located in the transmembrane region or the disordered region. These features were included in the feature set of our study.

Another important feature (i.e., *is\_HLA*) was also used in this study. It is used to indicate whether the protein containing SAP belongs to histocompatibility leukocyte antigen (HLA), a large family of proteins whose variations are used by the immune system to distinguish non-self from self-molecules [132]. The protein that contains SAPs is searched against the IMGT/HLA database [133] using Blast [26], and it is considered

as HLA if it hits a sequence satisfying both the e-value less than 1.0 and the sequence identity over 80% [38].

*3.2.2.2 Amino Acid Features Obtained from AAindex.* AAindex [25] is a database of numerical indices representing various physicochemical and biochemical properties of amino acids. There are two types of entries available in AAindex. The first type of entries has 20 values, with each value indicating the property of one amino acid. The second type of entries consists of a 20x20 matrix, giving the property between each pair of amino acids, e.g., a substitution matrix. We downloaded the current version of AAindex (as of Sept 13, 2007) and removed entries with missing values. Remaining were 666 entries, with 531 from the first type and 135 from the second. For each entry  $i$ , we defined a feature for the SAP site that measured the distance between the wild-type residue and the mutant-type residue

1) If entry  $i$  was a first type entry, then the feature was given by

$$f_i = \frac{\text{index}_i(\text{mut}) - \text{index}_i(\text{wildtype})}{\text{index}_i(\text{wildtype})} \quad (3.7)$$

where,  $\text{index}_i(\text{mut})$  and  $\text{index}_i(\text{wildtype})$  were the property values of wild-type and mutant-type residues given by entry  $i$ , as can be see from Table 4 (feature No. 21-551). Because some of the values in entry  $i$  could be 0, to avoid zero values in the denominator, the 20 values in entry  $i$  were normalized to the range of [0.1, 1.1].

2) If entry  $i$  was a second type entry, the feature was given by the value in the matrix corresponding to the pair of mutant-type and wild-type residues, as can be seen from Table 4 (feature No. 552-686). .

Finally, we obtained a set of  $666+19+1=686$  features for each SAP site. Note that in the calculation of these features for an SAP site, the structure of the SAP site was not required.

### 3.2.3 Decision Tree Algorithm

A decision tree is decision support tool, whose leaves represent classifications and branches represent conjunctions of feature conditions that lead to these classifications. At each node, a certain attribute is tested. Depending on the value of the attribute, it moves down to the related sub-tree following the corresponding branch. The process starts from the root node, and repeats iteratively until it reach the leaf node, where a classification is made and the confidence of such classification is reported. A decision tree can approximate discrete-valued target functions, in which the function is represented by a decision tree [35, 87].

ID3 [134] is a widely used learning algorithm of the decision tree. The algorithm employs a top-down, greedy search through the space of possible decision trees. For each node in the tree, it selects the attribute that is most useful in classifying the instances upon that stage. Generally, the attribute that has the highest information gain is chosen at each node for the available training examples to that node. The information gain of an attribute measures the expected reduction in entropy caused by partitioning the examples using the attribute, and it is calculated using

$$Gain(S, A) \equiv Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v), \quad (3.8)$$

where  $Values(A)$  is the set of all possible values for attribute  $A$ , and  $S_v$  is the subset of  $S$  for which attribute  $A$  has value  $v$ . Notice that the entropy of a collection of examples  $S_v$  is defined as

$$Entropy(S_v) \equiv \sum_{i=1}^c -p_i \log_2 p_i \quad (3.9)$$

where  $p_i$  is the proportion of  $S_v$  belonging to class  $i$  and there are  $c$  classes in total. In ID3, shorter trees are preferred over longer trees, and attributes that have higher information gain are placed closer to the root [87].

While information gain is widely used in selecting attributes to test, it is not perfect all the time. One disadvantage of information gain is it tends to prefer attributes with a large number of possible values. To solve this problem, C4.5 [135] uses information gain ratio, which is calculated by dividing the original information gain by the entropy of the attribute. Another drawback of ID3 is it will grow trees deeply enough just to perfectly classify the training examples. However, when there is noisy data or when the number of training data is not large enough, although the learned decision tree perfectly fits the training examples, it may fail to classify the new data. C4.5 tries to avoid this overfitting problem by employing a technique called rule post-pruning.

Decision trees have been widely applied in many classification problems, including the classification of SAPs [28, 112]. One benefit of using a decision tree is that it generates classification rules that can be easily interpreted, which aids in the study of the mechanisms of disease-related SAPs. In this study, we used the J48 decision tree package of WEKA [35], which is an implementation of the C4.5 algorithm.

### 3.2.4 Performance Measurement

Overall accuracy, Matthews correlation coefficient (MCC), sensitivity, specificity, true positive rate (TPR), and false positive rate (FPR) were used to measure the performances. They are defined as

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP} \quad (3.10)$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FN)(TP + FP)(TN + FP)(TN + FN)}} \quad (3.11)$$

$$Sensitivity = \frac{TP}{TP + FN} \quad (3.12)$$

$$Specificity = \frac{TN}{TN + FP} \quad (3.13)$$

$$TPR = \frac{TP}{TP + FP} \quad (3.14)$$

$$FPR = \frac{FP}{TN + FP} \quad (3.15)$$

where  $TP$  is the number of true positives (i.e., the number of “Disease” SAPs predicted as “Disease”);  $TN$  is the number of true negatives (i.e., the number of “Polymorphism” SAPs predicted as “Polymorphism”);  $FN$  is the number of false negatives (i.e., the number of “Disease” SAPs incorrectly predicted as “Polymorphism” SAPs) and  $FP$  is the number of false positives (i.e., the number of “Polymorphism” incorrectly predicted as “Disease” SAPs).

Accuracy is the overall percentage of SAPs correctly predicted. MCC (Matthews correlation coefficient) measures the correlation between predictions and actual class labels. In a two-class classification, if the numbers of the two classes are not equal, MCC

is a better measure for evaluating the performance than accuracy [91]. In this study, the numbers of two classes (“Disease” and “Polymorphism”) are not equal. Thus, MCC is used as the primary measure for evaluating the performance in this study. Sensitivity is a measure of the percentage of “Disease” SAPs correctly classified. The true positive rate is the percentage of correctly classified “Disease” SAPs among all SAPs predicted to be “Disease.”

### *3.2.5 Cross-Validation and Independent Test*

Ye’s dataset was used to train and evaluate the proposed method. In the study of Ye et al. [38], the dataset was divided into five subsets at the protein level, such that SAPs from the same protein would be put into the same subset, thus ensuring much more stringent criteria than in other studies. In this study, we used the same dataset partitions as in [38]. The proposed method was evaluated using both cross-validations and an independent test.

Four subsets were used to perform feature selection using four-fold cross-valuation. In each round of experiments, three subsets were used as a training set, and the remaining subset was used as test set. This procedure was repeated four times with each subset being used as a test set once. The average performance is calculated. During the feature selection process, the average MCC was used as the primary performance measurement to evaluate the effectiveness of the selected feature set.

In the independent test stage, the fifth subset (independent set) was used to test the classifier. The decision tree was trained based the four subsets, using the selected features and then tested against the independent test dataset. Since the independent subset

was not seen in the training and feature selection processes, it provides a more reliable evaluation of the method.

### *3.2.6 Feature Selection Process*

In the learning process of a decision tree, a feature selection process is already embedded in the construction of tree nodes. For each node, the decision tree chooses the most useful attribute (i.e., the attribute with the highest information gain or gain ratio for the available training examples). The final feature set chosen to build the decision tree is usually a subset of all available features. However, this embedded feature selection process does have its problems. It ignores the correlations, redundancies, and errors among features. Therefore, an external feature selection process is required. In this study, a greedy selection process was applied to choose the most relevant features to build the decision tree. It wraps the decision tree classifier in the feature selection process.

As mentioned in the previous section, four subsets were used to select the most useful features to build the classification method. The feature selection process in this study is similar to the one proposed in Section 2.2.5 except that it has only one growth stage. The detailed feature selection algorithm is as below.

Let  $S$  be the set of the selected features,  $A$  be the set of available features, and  $N$  be the size of  $A$ . At the beginning,  $S$  is empty, and  $N=686$ . Features are added into  $S$  using the following procedure:

- (1) Pick one feature from  $A$ ;

- (2) Build classifiers using the newly picked feature and the features in  $S$ , and evaluate the classifiers using a four-fold cross-validation. Notice that MCC is used as the primary performance measurement;
- (3) Repeat steps (1) and (2)  $N$  times, so that every feature in  $A$  is tried once. The feature that brings the largest improvement in performance is removed from  $A$  and added into  $S$ . The value of  $N$  is deducted by 1.

This procedure continued until including more features into  $S$  does not increase the performance. In the end, ten features were selected.

### 3.3 Results

#### 3.3.1 *The Developed Method Identifies Deleterious Nonsynonymous Single Nucleotide Polymorphisms*

Four subsets were used to select features, using a greedy search approach. As can be seen from Figure 2, the prediction performance, measured using MCC, improves as the number of selected features increases and reaches its maximum when ten features are selected. The MCC remains unchanged when 11 and 12 features are selected. When more than 12 features are selected, the MCC slightly decreases. Therefore, 10 features were finally selected to build the decision tree. Table 6 lists all ten selected features in the order that they were chosen.

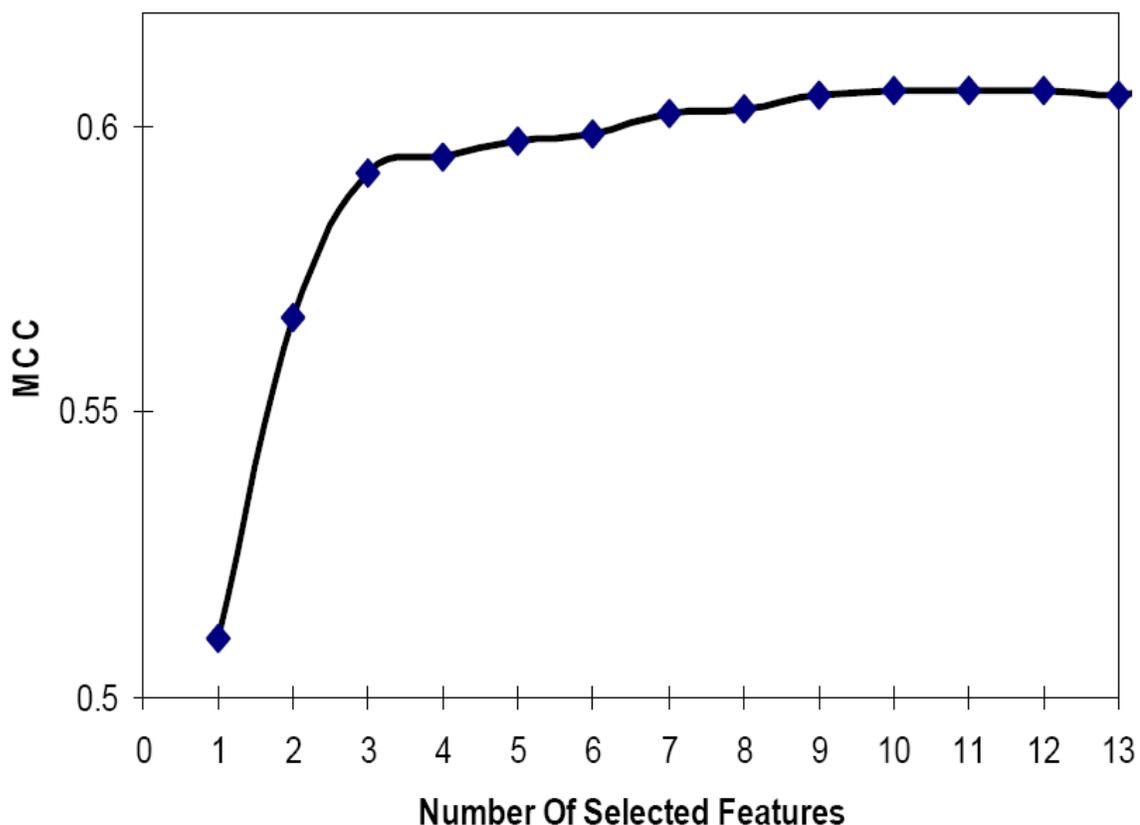


Figure 2. Classification performances as feature selection process progresses.

First, the proposed method was evaluated by four-fold cross-validation. As can be seen from Table 5 (column 2), the method achieves an 82.6% accuracy with 0.607 MCC.

The proposed method was further evaluated using an independent test, in which the classifier was trained using the four subsets and tested on an independent set. Note that the independent set was not seen by the algorithm during the feature-selection stage and the training of the classifier. The results (Table 5, column 3) show that the method achieves an 82.6% accuracy with 0.604 MCC in the independent test. Table 5 also indicates that the proposed method achieves consistent results in cross-validation and independent test.

Table 5. Prediction Performances of the Proposed Method.

	Cross-validation	Independent test	Swiss-Prot
MCC	0.607	0.604	0.42
Accuracy (%)	82.6	82.6	73.2
Sensitivity (%)	94.9	94.7	84.0
True Positive Rate (%)	81.6	81.6	75.0

Table 6. List of Features in the Order They Are Selected.

Feature	Annotation
is_HLA	Whether the protein containing the SAP belongs to HLA family [38].
nor_diff_freq	Normalized difference between mutant-type residue frequency and wild-type residue frequency.
DAYM780301	Log odds matrix for 250 PAMs [110]. The value between two amino acids shows how often one amino acid replaces another one in evolution.
FEND850101	Structure-Genetic matrix [136]. This matrix takes into account of the structural similarities of amino acids and the genetic code.
ZHAC000105	Environment-dependent residue contact energies [137]. The residue contact energies in different structural environment.
HENS920103	BLOSUM80 substitution matrix [106]. The value between two amino acids is defined based on the log likelihood of one amino acid substitutes the other by chance in sequence alignment.
NAKH900106	Normalized composition from animal [138]. Normalized residue composition calculated from animal mitochondrial proteins.
metal_seq_neibor	The sequence distance between the SAP site and its nearest residue holding the functional site with Feature Key of METAL [38].
MIYS850103	Quasichemical energy of interactions in an average buried environment [139].
modres_seq_neibor	The sequence distance between the SAP site and its nearest residue holding the functional site with Feature Key of MOD_RES [38].

In comparison, when all 686 features were used, the decision tree only achieved 0.503 MCC with a 77.7% accuracy evaluated using cross-validation.

### 3.3.2 Analysis of Selected Features.

At each step of the feature selection process, the feature that brought the largest improvement in performance was chosen. Table 6 lists all ten features in the order they were selected. The earlier a feature is selected, the more useful information it provides in the classification of SAPs. Among the ten selected features, three (i.e., *is\_HLA*, *metal\_seq\_neibor*, and *modres\_seq\_neibor*) were derived from a previous study [38], *nor\_diff\_seq* and the rest 6 features are novel to this study.

Two features are related to residue frequency (*nor\_diff\_seq* and *NAKH900106*), and two features related to substitution (*DAYM780301* and *HENS920103*). Feature *nor\_diff\_seq* is the normalized frequency difference between mutant-type residue and wild-type residue calculated from the position-specific scoring matrix, which is generated by PSI-BLAST [26]. Feature *NAKH900106* is the normalized composition difference between wild-type and mutant-type residues from animal proteomes, but it does not consider the position information of SAPs. Features *DAYM780301* and *HENS920103* give the possibility of one amino acid being replaced by another one in evolution. Three features (*FEND850101*, *ZHAC000105* and *MIYS850103*) are related to the differences of structure and contact energies between wild-type residue and mutant-type residue. Two features (*metal\_seq\_neibor* and *modres\_seq\_neibor*) measure the sequence distances of SAP positions to the nearby functional sites. One feature (*is\_HLA*) tells about the family of the protein. As can be seen from Table 6 and Figure 3, *is\_HLA* is the first feature

selected and is tested at the root node, which suggests that different classification rules apply to different protein families. If *is\_HLA* is positive (i.e., the protein belongs to the HLA family), it is very likely that the SAP is a polymorphism with confidence = 336/337. This is because that there are not many proteins in the training dataset belonging to the HLA family, and most of their SAPs are disease related.

Figure 3 shows the complete decision tree trained on four subsets using ten selected features. The two numbers inside each leaf node give the number of correction predictions versus the number of wrong predictions, which provide the confidence scores of the classification. For example, “Disease (1380.0/157.0)” on the leaf node indicates that when the leaf node is reached, the SAP will be predicted as “Disease” with a confidence score of 0.898 ( $1380/(1380+157)$ ). One benefit of using decision tree is that it can generate a set of human interpretable if-then rules (disjunctions of conjunctions) that provide insights into the mechanisms of deleterious SAPs. Following are some of rules derived from the decision tree for proteins that do not belong to the HLA family:

Rule 1: “*If (nor\_diff\_seq ≤ -0.96), then Disease*”

Rule 2: “*If (DAYM780301 ≤ 3.4) and (nor\_diff\_freq ≤ -0.37), then Disease*”

Rule 3: “*If (3.4 < DAYM780301 ≤ 3.84) and (nor\_diff\_freq ≤ -0.90), then Disease*”

Rule 4: “*If (DAYM780301 > 3.84) then*

*If (9 < Metal\_seq\_neighbor ≤ 29) then Disease*

*Else then Polymorphism*”

As mentioned in Section 3.2.2.1, if the normalized frequency difference (*nor\_seq\_diff*) is negative, the wild-type has a higher frequency than the mutant-type at

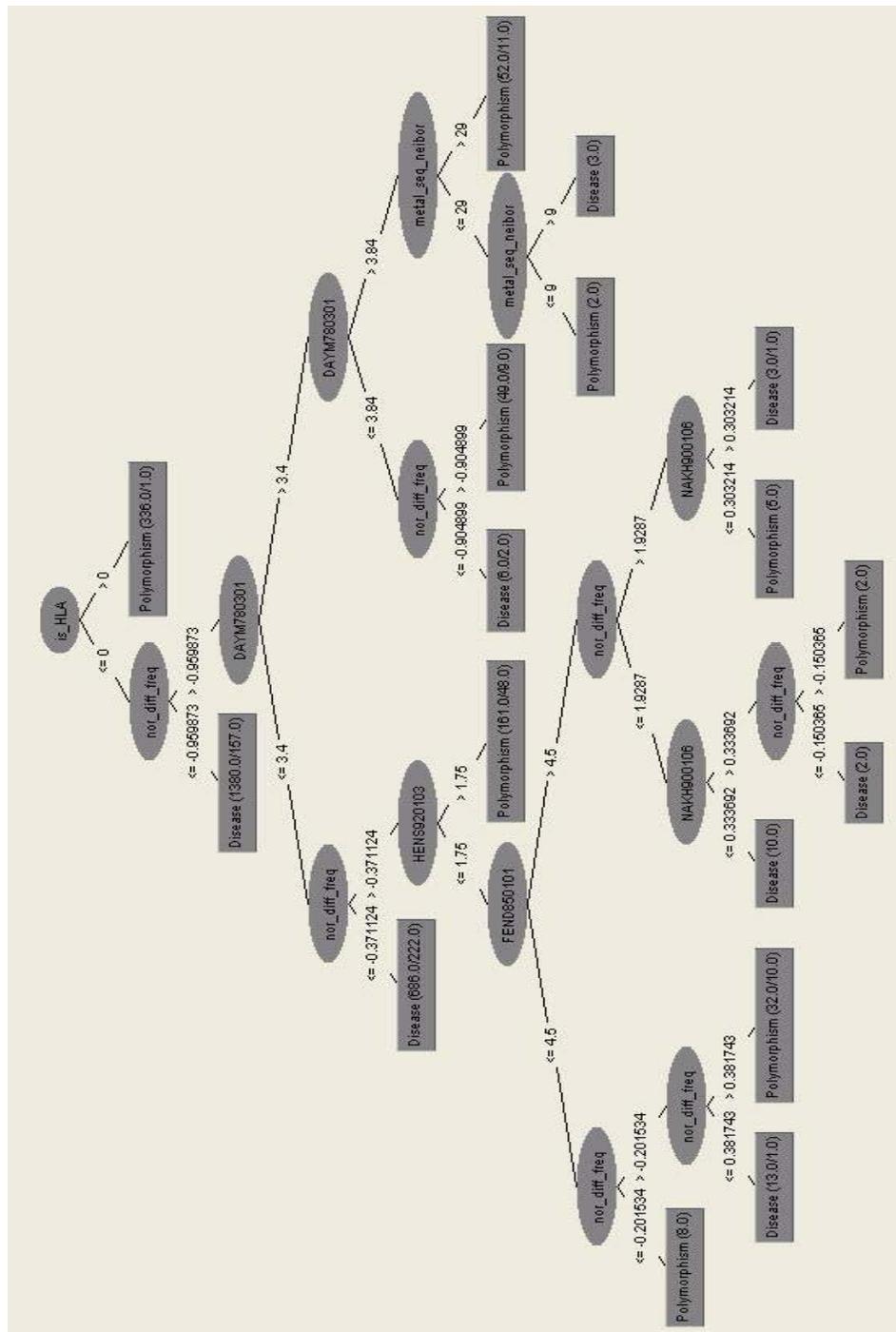


Figure 3. Decision tree trained on ten selected features as visualized using WEKA [35].

the SAP position. The lower the negative value, the higher the absolute frequency difference between wild-type residue and mutant-type residue. *DAYM780301* is the log odds matrix of PAM 250 [110]. The value between two amino acids shows how often one amino acid replaces the other in evolution. The higher the value, the more frequently one amino acid replaces the other. In other words, the matrix can be viewed as a measure of evolutionary similarity between amino acids. Higher values correspond to higher evolutionary similarities between residues. The four rules can be interpreted as below.

Rule 1: *If the residue frequency of mutant-type is lower than that of wild-type at the SAP site, and if the difference is significantly large enough ( $\leq -0.96$ ), the mutation is deleterious with a confidence of 0.898.* This suggests that if wild-type residue is much more common than mutant-type residue at the SAP position, in other words, if the mutation happens at conserved regions, the mutation is very possibly disease-related.

Rule 2: *If the evolutionary similarity between the mutant-type and wild-type is very low ( $\leq 3.4$ ), then although the difference between their frequencies is not very high (only  $\leq -0.37$ ), the mutation is still disease-related with a confidence of 0.7555.*

Rule 3: *If the evolutionary similarity between the mutant-type residue and wild-type residue is at median levels ( $3.4 < \text{DAYM780301} \leq 3.84$ ) and the frequency difference between the mutant and wild types is high (has to be  $\leq -0.90$ ), the SAP is disease related with a 0.755 confidence score.*

Rule 4: *If the similarity between the mutant-type and wild-type is very high ( $\text{DAYM780301} > 3.84$ ), the difference between their residue frequencies is no longer a crucial factor in determining the effect of the mutation.*

The rules generated from the decision tree suggest that if a mutation causes little changes (in chemical and physical properties, structural stabilities, or other properties) to the protein, then the mutation is likely neutral. Otherwise, it is very likely deleterious. Conforming to previous studies, the frequency difference between wild-type and mutant-type has proved to be very useful in the classification of SAPs. Other than residue frequency, similarities (geometrical or other related properties) between mutant-type and wild-type also needed to be considered.

### *3.3.3 Comparisons with Previously Published Methods*

Our method was compared with other published methods on two datasets: Ye's dataset, which has 2249 "Disease" and 1189 "Polymorphism" SAPs, and Swiss-Prot dataset, which has 12944 "Disease" and 8241 "Polymorphism" SAPs. Both datasets have a lot more positive ("Disease") instances than negative ("Polymorphism") instances. As mentioned in [91], in -two-class classification problems, if the numbers of the two classes are not balanced, MCC is a better measurement than overall accuracy in evaluating classification performance. Thus, MCC was used as the main measure in the comparison of different methods.

A receiver operating characteristic (ROC) curve is a graphical plot of a true positive rate (TPR) versus a false positive rate (FPR), or sensitivity versus (1-specificity) as the discrimination threshold changes [35], where (1-specificity) corresponds to the horizontal axis and sensitivity corresponds to vertical axis. In the plot, the diagonal line is the separation of good classification from poor classification. If a prediction generates a point above the diagonal line, it is called a good classification, otherwise a bad

classification. To the extreme, (0, 1) point is called the perfect classification, and it represents 100% sensitivity and 100% specificity, and the horizontal axis is called a no-discrimination line since it does not predict any positive examples. Therefore, the higher the ROC curve is above the diagonal line, the better the performance of the classification method. Another important statistic to summarize the ROC curve is the area under the curve, or AUC, which represents the probability that a classifier ranks a randomly chosen positive instance higher than a randomly chosen negative one. Thus, the larger the AUC, the better the classification model is [35].

As mentioned, one merit of the decision tree method is that it can generate predictions with confidence scores. One way to adjust the predictions is by setting up the discrimination threshold of confidence score  $T_s$ . A SAP is reported as “Disease” only if it is predicted by the decision tree to be “Disease” and the confidence score of the prediction is higher than the threshold  $T_s$ . By changing the threshold value, we have the flexibility of performing benefit (TPR) versus cost (FPR) analysis. In this study, competitive methods, such as SIFT [98, 99, 100] and PANTHER [126, 127, 140], also provide parameters so that prediction results can be adjusted, therefore making it possible to compare the prediction performances using ROC (receive operating characteristic) curves.

Bromberg and Rost [119] developed a neural network method (SNAP) for classifying SAPs. They evaluated the method on a dataset obtained from Protein Mutant Database [121], and a 78% accuracy was reported. The dataset from Protein Mutant Database is based on experimental amino acid substitutions, while the dataset used in this

study is based on observed human alleles. Due to the difference in the datasets used in the two studies, a direct comparison between SNAP and the method proposed in the current study is not possible.

*3.3.3.1 Comparisons on Ye's Dataset.* As mentioned, SAPRED is a support vector machine- (SVM) based method [38]. It used 60 structural and sequence-based features to make predictions about the effect of SAPs. In this study, we used the same dataset and partitions as SAPRED used. On the same datasets, SAPRED achieved an 82.6% accuracy and 0.604 MCC, and our method achieve an 82.6% accuracy and 0.607 MCC (see Table 7). While the performances of the two methods are comparable, the virtues of our method are two-fold. First, our method requires only sequence-derived information as input; thus it is applicable to SAPs whose structures are not available. Second, our method is based on a decision tree algorithm that is simpler than the SVM used by SAPRED. During the training of a decision tree-based classifier, no parameters need to be tuned. In contrast, the training of an SVM requires enormous efforts to search for optimal parameters (e.g., C and gamma) and takes a much longer time. Compared with SVM, an additional benefit of the decision tree is that the decision tree produces rules that can be easily interpreted. In this study, we also tried SVM instead of a decision tree, but no improvement was observed by switching to SVM. Note that, in their study, Ye et al. also presented a sequence-version of SAPRED that only required sequence-derived features as input. But the sequence-version of SAPRED achieved only 0.577 MCC (see Table 7), which is lower than that of our method.

Ye et al. compared SAPRED with other methods such as SIFT [98, 99, 100] using the same dataset. Based on the results they report, SIFT achieved 0.480 MCC (see Table 7). For a comparison with PANTHER [126, 127, 140], we submitted the dataset used in this study to their web server. PANTHER only achieved 0.318 MCC using the default threshold (see Table 7).

Figure 4 shows the ROC curves of the proposed decision tree method, as well as SIFT and PANTHER on Ye's dataset. It demonstrates that our method outperforms the other two methods.

*3.3.3.2 Comparisons on Swiss-Prot Dataset.* Since our method only requires sequence-based information, it can be applied to cases where the 3D structures of the proteins are not available. We also evaluated our method on a much larger dataset, Swiss-Prot dataset, which contains all the human SAPs from the Swiss-Prot variant database.

Table 7. Comparisons of Classification Methods of SAPs.

Method	MCC	
	Ye's dataset	Swiss-Prot dataset
Decision Tree	0.607	0.420
SAPRED	0.604	
SAPRED (sequence-version)	0.577	
SIFT	0.480	0.330
PANTHER	0.318	0.325

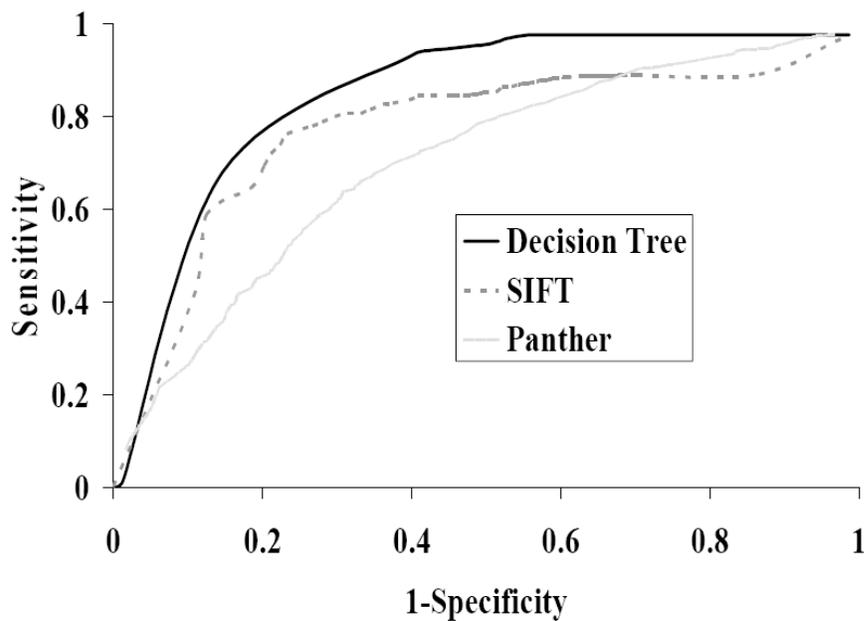


Figure 4. ROC curves of the proposed decision tree method, SIFT, and PANTHER on Ye's dataset. Area under ROC curve is 0.85 for decision tree, 0.77 for SIFT and 0.74 for PANTHER.

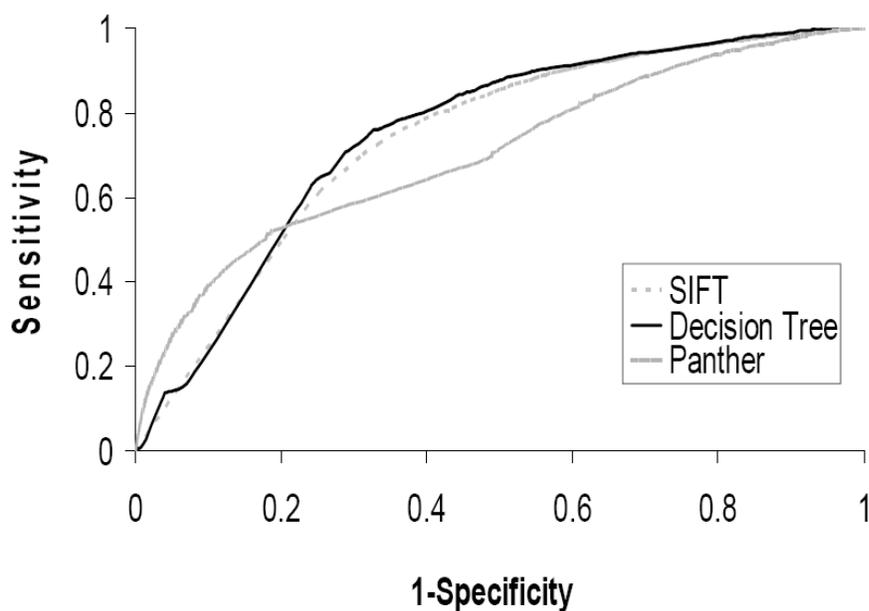


Figure 5. ROC curves of the proposed decision tree method, SIFT and PANTHER on Swiss-Prot dataset. Area under ROC curve is 0.75 for Decision Tree, 0.73 for SIFT and 0.70 for PANTHER.

Both SIFT and PANTHER only require sequence information, therefore they are comparable with our method on Swiss-Prot dataset. We were unable to evaluate SAPRED since it takes input of both structural and sequence features.

As can be seen from Table 7, when tested on Swiss-Prot dataset, the proposed method achieved 0.42 MCC, which is higher than that of SIFT (0.33 MCC) and PANTHER (0.325 MCC). The ROC curves (see Figure 5) proves our method still outperforms SIFT and PANTHER.

#### *3.3.4 A Web Server for the Identification of Deleterious Nonsynonymous Single Nucleotide Polymorphisms*

A web server<sup>15</sup> (see Figure 6) based on the proposed method was developed and released online. It allows users to submit SAPs with their protein sequences for the prediction service. For each query protein, it collects ten features to make predictions. The average prediction time per SAP normally takes around ten minutes. After the user submits the SAP example, the server generates a link for the results page. The user can choose to retrieve the results later based on the link or by providing an email address so that prediction results can be automatically sent back.

### **3.4 Discussion**

In this study, we explored the possibility of identifying disease-related SAPs using only information derived from protein sequences. The performance of the proposed method is much higher than that of SIFT, a classic method for classifying SAPs, and it is

---

<sup>15</sup> The server is available at <http://yanbioinformatics.cs.usu.edu:8080/SAPsubmit>.

comparable to that of SAPRED, a state-of-the-art method. The novelties of our method include the following:

1. In this study, we compiled a set of 686 features that were derived from proteins sequences. The number of features extracted in this study is more than ten times larger than those considered in previous studies.
2. In contrast to previous published methods which only consider a small set of arbitrary chosen features, we used an automatic feature selection method to discover useful features in classifying SAPs. Using selected features, a decision tree method was developed to identify deleterious SAPs.
3. The developed method only requires sequence-derived features as input; therefore, it can be applied to proteins with no structure information.
4. In contrast to the work of Dobson et al. [28] and Ye et al. [38] which evaluated the prediction power of each feature individually, in this study a more systematic analysis was employed to identify features that play vital roles in determining the effects of SAPs. From the trained decision tree, we derived a set of human-interpretable rules, which provides insights in understanding the mechanisms of disease causing SAPs.
5. Depending on the dataset and the partition, different features might be chosen after feature selection, which may result in different sets of classification rules after the decision tree is trained. This can be explained by the fact that there are correlations among some features.

For example, features about conservation scores and residue frequencies are correlated, i.e., wild-type residue tends to have a higher frequency in conserved regions, etc. Therefore, various rule sets can be regarded as different representations of same knowledge base.

### **3.5 Conclusion**

In conclusion, we have developed a useful tool for identifying deleterious SAPs; said tool is especially when the structure of the protein is not available. For each SAP mutation, a set of 686 features were derived from protein sequences to describe differences between wild-type and mutant-type residues. After an automatic greedy feature selection, ten features were chosen and analyzed. Using selected features, the decision tree identifies deleterious SAPs with high performances. From the decision tree, we also extracted a set of useful rules that provide biological insights into the mechanisms of disease related SAPs.

### **3.6 Future Work**

Random forest (RF) is a classifier consisting of an ensemble of tree-structured classifier [115]. For each tree, RF uses a bootstrap sampling to choose the training dataset (i.e., sampling training data with replacement), and uses the rest to estimate the error. For each node of the tree, it selects the best feature among a subset of

## Single Amino Acid Polymorphism Effect Prediction

Paste your protein sequences here:

Or upload your file from:

*When you submit the file, it must be in FASTA format!*

Mutation Name  ([help](#))

Email:

Notice: If email address is provided, prediction results will be automatically sent back via email

(A)

**Mutation:** F20N  
**Protein:** P35825  
**Predicted effect:** Polymorphism  
**Confidence score:** 0.661 (0-1, with 1 is the highest and 0 is the lowest)

(B)

Figure 6. The web server for the prediction of deleterious SAPs. A: The Input form; B: The output form.

features randomly chosen at that node to split. Different from decision tree's training algorithm which has a pruning process (i.e., C4.5), each tree in the random forest is fully grown until "purity" is achieved at each node. RF makes predictions by aggregating all the trees across the forest, either by majority vote or by averaging. The whole bootstrap and aggregating processes combined are called *bagging*. RF works well for classification problems where there are too many features or variables, which is the case for this study (686 candidate features). RF is tolerant to noisy data and does not suffer from the over fitting problems. It usually has better classification capabilities than a decision tree. Therefore, the possibility of applying RF for better prediction accuracy and reliability of deleterious SAPs should be investigated in future work.

CHAPTER 4  
IDENTIFICATION OF HELIX-TURN-HELIX MOTIFS FROM  
PROTEIN SEQUENCE <sup>16</sup>

## 4.1 Background

### *4.1.1 Helix-Turn-Helix: An Important Structure Through Which Proteins Bind with DNA*

DNA-binding proteins play pivotal roles in many genetic activities within organisms, such as transcription, packaging, rearrangement, replication, and repair. They are responsible for the transfer of biological information from genes to proteins. It is estimated that around 2-3% of prokaryotic genome and 6-7% of eukaryotic genomes encode DNA-binding proteins [141, 142]. DNA-binding proteins and protein-DNA complexes with experimentally determined 3-dimensional structures can be found in public databases such as Protein Data Bank (PDB) [24] and Nucleic Acid Database (NDB) [143].

A lot of known DNA-binding proteins have been found to bind DNA by a number of structural motifs, such as the helix-turn-helix (HTH) motif, the helix-loop-helix motif (HLH), the helix-hairpin-helix motif (HHH) and the zinc finger motif [144], of which the DNA-binding HTH motif is one of most important and well studied. HTH motifs can be classified into several classes based on their structures and sequences [145]. A typical HTH motif is composed of an alpha helix, a recognition helix that forms base-specific interactions with DNA, and a turn or linking region connecting two helices. The HTH

---

<sup>16</sup> Co-authored by Yan, C. and Hu, J.

motifs extend from the domain surface and constitute a convex unit capable of fitting into the major groove of DNA [146]. Most HTH motifs are about 20 to 22 residues in length. Usually for the purposes of analysis, three helices are considered. These are the recognition helix and the two helices preceding it [147, 148]. Proteins with low similarity in sequence can bind with DNA through similar HTH motifs [145, 147]. Figure 7 shows an example of HTH motifs binding with DNA in the lambda repressor-operator complex from PDB structure of 1LMB.

#### *4.1.2 Current Prediction Methods to Identify Helix-Turn-Helix Motif*

Due to the high-throughput genome sequencing project, an increasing number of protein sequences with little function annotation have been accumulated in public databases. Many newly sequenced proteins have no or low similarity to the current PDB

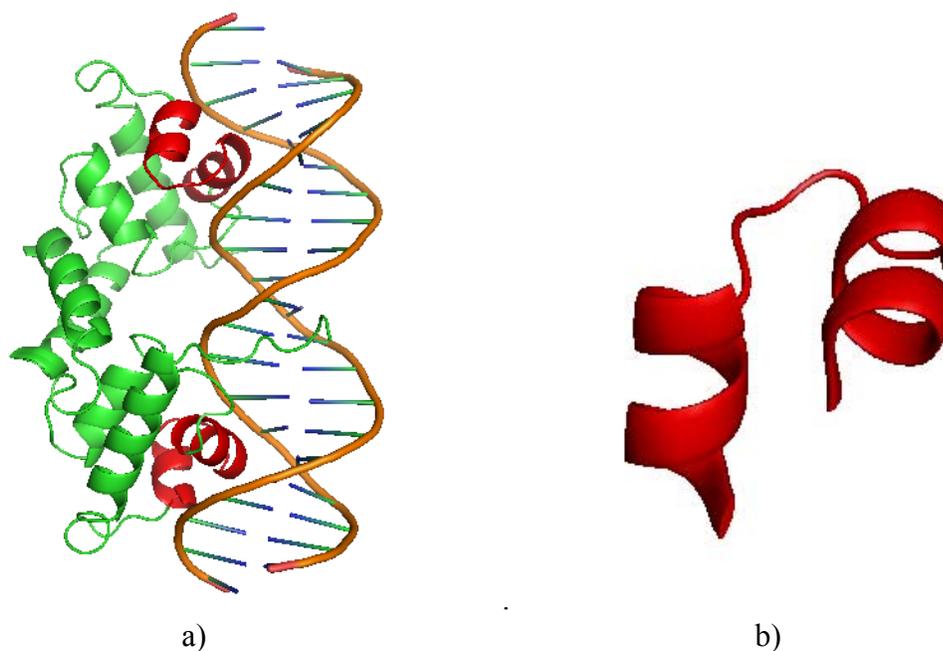


Figure 7. Images of HTH motifs. a) Proteins binding with DNA with red parts denoting helix-turn-helix motif; b) Motif of red part in a) is isolated.

[24] entries, which have experimentally resolved structures or functions, thus making it difficult to infer protein functions from their homologous proteins. Motifs, small conserved regions showing similar 3-dimensional folds and functional properties, are very important functional sites. Different proteins with the same motifs can have similar functions. Therefore, it is helpful to find motifs in protein sequences to investigate protein functions. Because of the importance of helix-turn-helix (HTH), it is necessary to identify HTH motifs for better understanding of the gene-regulation mechanism.

Identification of HTH motifs is not a trivial task. Different proteins with similar HTH motifs may only share very limited sequence similarity in the motif parts. In addition, the sequence length of motifs varies, and proteins may contain different number of HTH motifs.

Recently, several computational methods have been proposed to identify HTH motifs. Based on the type of information utilized, most methods can be classified into three categories, which are sequence-based methods, structure-based methods, and methods that explore both sequence and structural information.

*4.1.2.1 Sequence-based Methods.* Early sequence-based methods tried to identify HTH motifs by looking for occurrences of consensus sequences. Consensus sequences were constructed for the HTH motifs based on the multiple alignment of known motif sequences. For the query protein, a HTH motif is reported when such consensus sequences occur [146]. A more generalized consensus borrowed the concepts of regular expressions to allow amino acids substitutions for certain positions [149]. The profile method [150] has also been widely used to identify motifs. Based on the multiple

alignment of all known motifs, it computes a position-specific probability matrix or score matrix, which assigns a score for every amino acid at each position in the motif. For a sliding window in the input proteins, it calculates the weighted score based on the position specific score matrix. The subsequence covered by the sliding window is reported as motif if the score is above a user-defined threshold.

Recently, a “Pattern Dictionary” method was proposed to identify HTH motifs in protein sequences. The method, called the GYM algorithm, identifies HTH motifs by detecting the occurrence of sequence patterns from the pattern dictionary [151, 152]. The algorithm contains two parts: preprocessing and detection. The preprocessing part, also called “Pattern Mining” part, is the training stage. It takes as input a master set of aligned motifs without spaces. The motif is represented by a sequence of pairs. Within each pair,  $\langle \text{aa}, \text{pos} \rangle$  denotes a certain amino acid in a specific position (i.e., aa: amino acids; pos: position in the protein sequence). For example,  $\{G1, S4, M6\}$  is such a pattern, meaning amino acid G is in position 1, S is in position 4, and M is in position 6. The length of this pattern is 3. The support of a pattern is the number of training proteins in which the pattern appears. If the support of a pattern is greater or equal to a user-defined threshold, the pattern is called a significant pattern. A maximal pattern is a significant pattern not contained by any other significant proteins. The preprocessing stage outputs a pattern dictionary consisting of maximal patterns or frequent occurring patterns, which are used to find HTH motifs in the new protein sequence during the detection stage. The detection stage takes as input a motif length  $m$ , the dictionary of maximal patterns  $L$  output from the pattern mining stage, an integer  $k$  representing the number of best matches required as

output, and the new protein sequence  $P$ . A sliding window of length  $m$  is moved along the protein sequence  $P$ , and a matching process is performed for each subsequence covered by the window. The matching process returns a match-score measuring how well the window matched against the patterns in the pattern dictionary  $L$ . The  $k$  best match-scores with the corresponding locations are outputted as possible HTH motifs if the according scores are above a predefined threshold.

*4.1.2.2 Structure-based Methods.* Structure-based methods identify HTH motifs based on the structural templates; therefore, it only works for proteins whose 3-dimensional structures have been experimentally determined.

Early attempts [153] classified HTH structures according to the configuration of the helices (i.e., two alpha-helices in the helix-turn-helix combination, associated with one or two other helices before or after the motif). These methods relied on the angles between alpha helical portions in the alpha helix bundle containing the motif to make predictions. A measurement that sums the torsion angles between all pairs of alpha helices in the bundle part is calculated. For the interested region of the query protein, the difference of its summed torsion angles from that of HTH motifs is calculated as the similarity score, which is then used as a standard to predict if the region is an HTH motif or not.

Recent structure-based methods utilize more complex structural information to discover HTH motifs based on a statistical model. McLaughlin and Berman [154] developed a method to predict HTH motifs relying on the geometrical measurement of the motif. The structural measurements were based on eight possible secondary structure

elements for each motif, including the recognition helix (RH), the two alpha helices preceding the RH (RH-1, RH-2), the first alpha helix following the RH (RH+1), the two beta strands preceding the RH (S-1, S-2), and the two beta strands following the RH (S+1, S+2). There are four measurements in total. 1) The hydrophobic contact area for each secondary structural element pair was measured, providing information about which pair of secondary structure element contacts each other and the extent of the contact. There are 28 pairs in total. Many previous studies have confirmed this attribute to be a useful feature for the identification of HTH motifs. 2) The average relative solvent exposure of each secondary element was calculated from the residue's relative solvent accessibility, which is predicted using an NACCESS program [155]. Since the two alpha helices of HTH motifs are at the protein surface, the relative solvent accessibility is useful information because it measures the exposure level of a secondary structure element. 3) The average number of lysine residues and arginine residues per secondary structure element was counted, because such residues display positive charges and tend to bind with DNA which has a negatively charged backbone. 4) The torsion angles between neighboring alpha helices were calculated by PROMOTIF [156]. This measurement has also been used in previous studies [153]. A positive dataset that contains HTH motifs and an equal size of negative dataset that contains similar structures to the HTH motifs were constructed. Next, a J48 decision tree combined with the adaboost algorithm from the WEKA software package [35] was built to identify HTH motifs. The method was applied to the Protein Data Bank to find HTH motifs not previously reported.

Other structure-based methods try to find structures that can fit into the structure templates developed from known HTH motifs. From structurally non-homologous DNA-binding proteins in the Protein Data Bank, Jones et al. [157] built a library of 3-dimensional structural templates of HTH motifs. There are two types of templates: original templates and extended templates. An original template is a set of  $C_{\alpha}$  backbones from the first residue in the first alpha-helix to the last residue of the recognition helix. An extended template was created in a similar way, but includes a certain number of residues preceding the first alpha-helix and succeeding the recognition helix. The final template library includes seven original templates and seven extended (plus two residues in both directions) templates. These templates were scanned against the whole protein structure. A scan-rmsd algorithm based on the Kabsch method [158] was used to calculate the root mean squared deviation (rmsd) for the optimal superposition of each template on each structure. The rmsd of the input protein was taken as the minimum rmsd obtained from all the superpositions. The structures with rmsd values below  $1.6\text{\AA}$  and solvent accessible surface area (ASA) greater than  $990\text{\AA}^2$  were predicted to have HTH motifs involved in DNA binding, where ASA is the accessible surface area of all residues included in the +2 extended templates. Shanahan et al. [148] improved the method by adding an ASA threshold and electrostatic motif score (EMS) threshold to reduce the false positives and to increase the accuracy. HTHquery, a recently developed web server based on the method, allows structural biologists to submit protein structures to identify HTH motif [159].

#### *4.1.2.3 Methods That Explore Both Sequence and Structural Information.*

Because large variations exist in the sequence and structure of HTH motifs, neither sequence-based methods nor structure-based methods alone can identify HTH motifs with perfect performance. Methods that explore both sequence and structural data have shown promising results. Pellegrini-Calace and Thornton [160] analyzed the potential of combining both sequence and structural knowledge for the identification of HTH motifs. From a set of nonhomologous DNA-binding proteins containing HTH motifs, two different libraries of hidden Markov models (HMMs) were built. One library of HMMs was built from whole DNA-binding domains, which include the HTH motifs. The other library of HMMs was built from a much shorter domain corresponding to the functionally relevant HTH motif itself. Pellegrini-Calace and Thornton used the two libraries to scan against a dataset of protein sequences. The authors expected that HMMs based on HTH-only sequence should perform better than HMMs based on full sequence, but no such result was discovered. In fact, HTH identification performance can be significantly enhanced by combining the HMMs with structure-based 3-dimensional template method developed by [142]. The results prove that single-feature methods (either sequence-based method or structure-based method) are less powerful than those methods that utilize both sequence and structure information.

*4.1.2.4 Other Methods Capable of Predicting HTH Motifs.* Pfam [161] is a collection of proteins classified into several families based on sequence similarity. A profile HMM is constructed for each protein family. Proteins from different families share low sequence similarity. Some families contain HTH motifs. Pfam profiles built for

families of HTH proteins can be used to search HTH motifs. But the limitation of a Pfam profile is also obvious. A Pfam profile built from one HTH family can rarely identify HTH motifs from other families. If a protein does not fall into any currently available HTH family, Pfam profiles often fail to find HTH motifs. Moreover, some families do not have enough sequences to develop an effective Pfam profile.

#### *4.1.3 Motivations of This Study*

There are several limitations to most current methods.

1. Most sequence-based methods used HTH consensus, profiles, or pattern dictionary, which were built from the multiple sequence alignment of HTH motif sequence parts to scan the input proteins for the HTH motifs. A matching process is conducted for each sliding window moving along the protein sequence. For these methods, only conservation parts of HTH motifs are considered; however, the relationships between motif and other regions are ignored. Moreover, subsequences with similar sequence patterns can display different structure conformations depending on the 3-dimensional environment. Therefore, it is not always efficient to find HTH motifs using sequence templates.
2. Structure-based methods try to find functional sites from structure templates and require structure related features; therefore, they cannot be applied to cases wherein the 3-dimensional structures of proteins are not available.

3. Pfam profiles can rarely identify HTH motifs on proteins from remote families, especially when a protein does not fall into any currently available families.

In this study, we developed a profile HMM method to identify HTH motifs.

Different from the approach of [160], the proposed method only takes input of information derived from protein sequences. As mentioned in previous sections, methods utilizing both sequence and structural information achieve the best performance. Thus, structural information such as solvent accessibility is also modeled. The novelty here is that solvent accessibility of each residue is predicted from the protein sequence; therefore, it is still a sequence-derived feature. In order to catch the common properties of HTH motifs, reduced alphabets were investigated to group amino acids based on similar physicochemical properties. Because proteins sharing low sequence similarity can bind with DNA through similar HTH structures, we also investigated the ability of the proposed method to identify HTH motifs from remote proteins that have a limited sequence similarity to the current known DNA-binding proteins.

## **4.2 Materials and Methods**

### *4.2.1 Dataset*

A typical HTH motif consists of a bundle of three helices: the recognition helix that forms base-specific interactions with DNA, and the two helices preceding it [147]. The three helices are connected by two irregular regions. We focused on the strictest type of HTH motif in which there are no sheets between the helices. From Pfam [161], we extracted all families belonging to the HTH clan. The structures of these families were

visually inspected to ensure that these proteins contain an HTH motif of standard shape.

There are two datasets constructed for different purposes.

Dataset 1: After discarding families with no structure information, 19 families containing the helix-turn-helix (HTH) motif were extracted from the Pfam database. The full alignment of the whole Pfam database was obtained. A sequence was put into the negative set if it did not belong to any of the 19 families, and into the positive set otherwise. The negative set has 4,687 sequences. The positive set consists of the sequences from 19 families. Among them, one family has many more sequences than others. To avoid bias, we only chose nine sequences from this family for the positive dataset. In total, the positive dataset contains 70 sequences. This dataset was used for finding discretization thresholds of solvent accessibility, which is introduced in detail later.

Dataset 2: For all the 19 families, we discarded families having less than 30 sequences in their seed alignments. There were 12 families left. The structures of the HTH motifs in these families were visually inspected to ensure that they had a standard shape. Positive examples (i.e., proteins that belong to the family) were extracted separately for each family from the Pfam seed alignment. The Pfam IDs of these families and the numbers of positive examples (shown in parenthesis) are: PF08279 (109), PF04545 (164), PF01381 (194), PF01022 (42), PF00440 (112), PF00196 (30), PF00165 (90), PF01978 (53), PF02954 (96), PF03965 (48), PF08281 (142) and PF00126 (1635). One set of negative examples were used in this study. First, all the sequences in the Pfam seed alignment were extracted. The sequences belonging to the HTH clan were removed.

The sequences without structural information were also removed (in order to ensure that the sequences in the negative set do not contain HTH motifs). The sequences with less than 20 amino acids were also removed. The negative set consists of 2,497 examples.

#### *4.2.2 Hidden Markov Model*

After achieving its success in speech recognition, the hidden Markov model (HMM) has recently been widely applied to solve bioinformatics problems. As a statistical model, the system being modeled in HMM is assumed to be a Markov process. An HMM consists of a set of states. It can be viewed as a generation model that generates sequences of letters by going through paths of states. At each state, it emits observable letters based on the emission probabilities. The transitions among the states are controlled by the transition probabilities. A sequence of letters can be generated by the HMM with different possibilities by following different state paths; therefore, it is called hidden because there is no one-to-one correspondence between the states and the symbols [83, 162]. To calculate the probability of generating a sequence of letters following a particular path, one just needs to multiply all the transition probabilities on that path with the possibilities of each letter emitted by the corresponding state.

From an HMM, there are multiple ways of generating the same sequence of letters. Some state paths have higher probabilities than others do in generating the sequence. The one with the highest possibility is called the most probable state path, which can be found by the Viterbi algorithm [163]. By summing up all the possible paths that can generate the sequence, we obtained the possibility of a sequence of letters being generated by an HMM model, which can be calculated using a forward or backward algorithm [83, 162].

Another approach is to use the most probable state path as an approximation, which works amazingly well in practice.

HMM is trained by finding the parameters that can maximize the total possibility of generating all the training examples using such parameters. When the state sequences of training data are known, the HMM can be trained by a simple maximum likelihood estimation method. When the state paths are unknown, Baum-Welch [164] or Viterbi training algorithms [163] are good choices. The details about HMM can be found in Rabiner's tutorial [162].

The nature of the biological sequences (i.e., long chains of nucleotides or amino acids) is highly suitable for modeling by hidden Markov models, and impressive successes have been achieved [165G]. Smith et al. [12] applied HMM in pairwise sequence alignment. Krogh et al. [166] developed a hidden Markov architecture to represent profiles of multiple sequence alignments. Eddy [167] extended the architecture to develop profile hidden Markov models for protein families. The resulting database (known as Pfam) has been widely used in protein function annotations [161].

*4.2.2.1 Hidden Markov Model That Emits Only the Identity of Amino Acids (HMM\_AA).* In Pfam, protein families are represented by multiple sequence alignments and profile hidden Markov models (HMM), of which Profile HMMs are constructed from multiple sequence alignments (see Figure 8). The profile HMM has a linear left-to-right structure. The heart of the profile HMM is a set of match (M), insertion (I), and deletion (D) states. An M state corresponds to a consensus column in the multiple alignment. I and D states correspond to the insertions and deletions in the alignment, respectively. D

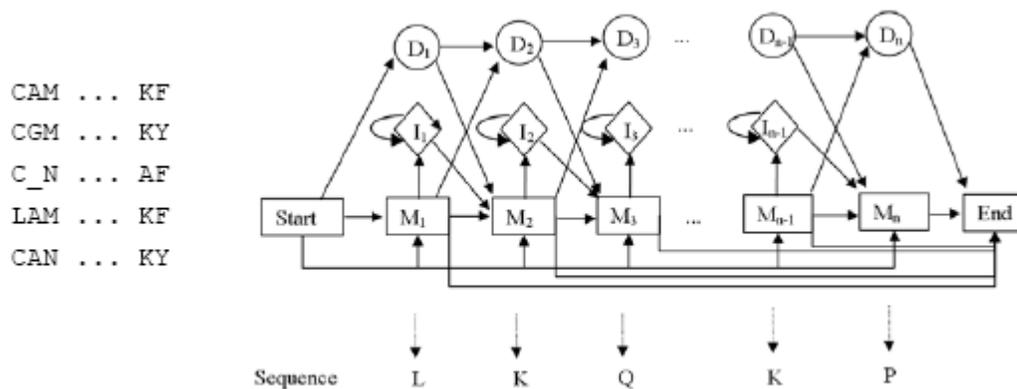


Figure 8. Hidden Markov model (right) that emits only amino acid residues (referred as HMM\_AA). It represents a multiple alignment of several sequences (left). M: Match state, I: Insertion state, D: Deletion state. Arrows show the state transitions. At each state, the model emits one amino acid residue.

state only produces a gap. Each M or I state emits one amino acid residue. Therefore, each M or I state is associated with 20 emission probabilities corresponding to the probabilities of emitting the 20 amino acid residues. The emission probabilities are determined by the frequency that residues are observed in the corresponding column of the multiple alignment. Transition probabilities between states are determined by the observed frequency of the corresponding transitions in the alignment. Profile HMMs are implemented by several software packages, among which HMMER is developed by Eddy's group and used in Pfam [168].

*4.2.2.2 Hidden Markov Model That Emits Both Solvent Accessibility and Identity of Amino Acids (HMM\_AA\_SA).* Although originally used to deal with sequence data, hidden Markov models have been applied to protein structure predictions in many studies [17, 53, 169]. Some studies encode the structure using one-dimensional symbols [170], while others explicitly model 3D coordinates [171]. Hargbo and Elofsson [172]

developed a hidden Markov model for fold recognition using amino acid sequence and predicted secondary structure. In their model, each state emits a letter of secondary structure in addition to a letter of amino acid residue. We adapted the Hargbo and Elofsson [172] approach to develop an HMM (referred to as HMM\_AA\_SA) method to model both amino acid sequence and predicted solvent accessibility (see Figure 9).

We modified Krogh's HMM to combine amino acid sequence and solvent accessibility. Figure 9 shows the core structure of the hidden Markov model (referred to as HMM\_AA\_SA) used in this study. The difference between the models in Figure 9 and Figure 8 is that the emission in Figure 9 includes both the identity of amino acid and their solvent accessibility.

Solvent accessibility of an amino acid residue measures the surface area of the residue that is accessible by solvent molecules. Relative solvent accessibility (RSA) is the fraction of its total surface that is accessible by solvent molecules. Relative solvent

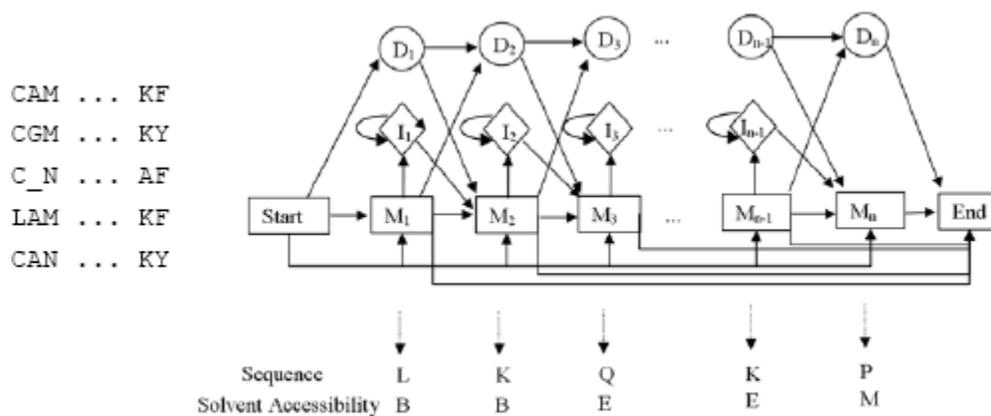


Figure 9. Hidden Markov model that emits both amino acids and solvent accessibility (Referred as HMM\_AA\_SA). M: Match states, I: Insertion states, D: Deletion states. Arrows show the state transitions. At each state the model emits one amino acid residues and its solvent accessibility.

accessibility falls in the range of [0, 1]. Solvent accessibility can be discretized based on the relative solvent accessibility. When the solvent accessibility is divided into two categories, i.e., Buried (B) and Exposed (E), each M or I state in Figure 9 is associated with 40 emission probabilities corresponding to the 40 combinations of 20 amino acid letters and two solvent accessibility letters. When the solvent accessibility is divided into three categories, i.e., Buried (B), Medium (M) and Exposed (E), each M or I state in Figure 9 is associated with 60 emission probabilities corresponding to the 60 combinations of 20 amino acid letters and three solvent accessibility letters.

When hidden Markov models were used to scan a protein sequence, a NULL model, which states the background occurrence of the query sequence, was used to calculate the significance of the hit [173]. E-value was used as the measure. The E-value shows the expected number of false positives that can fit the model at least as well as the hit. Thus, the lower the E-value, the more significant the hit. In this study, we chose  $E = 0.01$  as the cutoff to identify significant hits, meaning that the expected number of false positives is 0.01. We note that this cutoff is more stringent than the one ( $E=0.1$ ) suggested in the HMMER package [168]. We used a lower cutoff value to reduce the chance of recruiting insignificant hits.

#### *4.2.3 Feature Set*

Each M (match) or I (insertion) state of profile HMMs corresponds to a position in the protein sequence, and it emits certain number of letters or symbols. The HMM\_AA\_SA model proposed in this study can emit both identities of amino acid and

solvent accessibility. In particular, types of symbols can be emitted by each M or I state include:

*4.2.3.1 Amino Acids.* There are 20 different amino acids.

*4.2.3.2 Solvent Accessibility.* As mentioned in previous sections, solvent accessibility is very important structural information of each protein residue and can be discretized into two categories, i.e., Buried (B) and Exposed (E), or discretized into three categories, i.e., Buried (B), Medium (M) and Exposed (E), based on different RSA thresholds. The solvent accessibility of protein sequences was predicted by submitting the sequences to the Jpred server<sup>17</sup> [27].

*4.2.3.3 Reduced Alphabets.* There are 20 basic amino acids. Some of them share similar physicochemical properties. Many studies have been conducted to reduce the alphabet size of amino acids by clustering amino acids based on different properties (see Table 8). We named these reduced alphabets Chem\_6, Func\_8, Mur\_15, Mur\_10, Mur\_8, and Li\_10, with the numbers denoting the sizes of the alphabets. Using reduced alphabets can reduce the complexity of a protein sequence. In this study, using a reduced alphabet has the additional benefit of reducing the number of parameters (emission probabilities) in the models. For HMM\_AA\_SA, when reduced alphabets and solvent accessibility letters were used, the number of emission probabilities associated with each M or I state is  $3*n$  (3 categorizes: B, M and E) or  $2*n$  (2 categorizes: B and E) depending on the discretization of solvent accessibility, where  $n$  was the size the reduced alphabet.

---

<sup>17</sup> Jpred server is available at <http://www.compbio.dundee.ac.uk/~wwwjpred/submit.html>.

Table 8. List of Reduced Alphabets.

Alphabet Name	Amino Acids	Alphabet	Property
Chem_6 ( <a href="http://bio.math-inf.uni-greifswald.de/viscose/html/alphabets.html">http://bio.math-inf.uni-greifswald.de/viscose/html/alphabets.html</a> )	IVL	A	Aliphatic
	FYWH	R	Aromatic
	KR	P	Positive charged
	DE	N	Negative charged
	GACS	T	Tiny
	TMQNP	D	Diverse
Func_8 (Clustered based on functional groups)	DE	A	Acidic
	AGIL	L	Aliphatic
	NQ	M	Amide
	FWY	R	Aromatic
	P	P	Imines
	CM	S	Sulfur
	ST	H	Hydroxyl
Mur_15 [174]	RHK	B	Basic
	LVIM	L	Large hydrophobic
	C	C	
	A	A	
	G	G	
	S	S	
	T	T	
	P	P	
	FY	F	Hydrophobic, aromatic
	W	W	
	E	E	
	D	D	
	N	N	
	Q	Q	
KR	K	Long chain positive charged	
H	H		
Mur_10 [174]	LVIM	L	Large hydrophobic
	C	C	
	A	A	
	G	G	
	ST	S	Polar
	P	P	
	FYW	F	Hydrophobic, aromatic
	EDNQ	E	Charged, polar
	KR	K	Long chain positive charged
	H	H	
Mur_8 [174]	LVIMC	L	Hydrophobic
	AG	A	
	ST	S	Polar
	P	P	
	FYW	F	Hydrophobic, aromatic
	EDNQ	E	
	KR	K	Long chain, positive charged
	H	H	
Li_10 [175]	C	C	
	FYW	Y	
	ML	L	
	IV	V	
	G	G	
	P	P	
	ATS	S	
	NH	N	
	QED	E	
	RK	K	

#### 4.2.4 Software Implementation

The software for constructing and searching profile HMMs used in this study was implemented by modifying the HMMER [168] package to allow multiple emissions in a state.

#### 4.2.5 Performance Measurements

Sensitivity and false positive rate (FPR) were used to measure the performances.

They are defined as

$$\text{Sensitivity} = \frac{TP}{P} \quad (4.1)$$

$$\text{FPR} = \frac{FP}{N} \quad (4.2)$$

where  $TP$  is the number of true positives (i.e., the examples that are positive and are predicted as such);  $P$  is the total number of positive examples;  $FP$  is the number of false positives (i.e., the examples that are negative but are predicted as positive);  $N$  is the total number of negative examples. Here, proteins containing HTH motifs are positive examples, otherwise negative examples.

### 4.3 Results

#### 4.3.1 Discretization of Solvent Accessibility

In HMM\_AA\_SA, each M or I state emits one letter of solvent accessibility in addition to the amino acid, and there are multiple ways of dividing solvent accessibility. In order to model solvent accessibility information in profiling an HMM, it is necessary to find the optimal discretization, i.e., the number of categories and the thresholds of relative solvent accessibility. Dividing solvent accessibility into too many categories

introduces the problems of insufficient training, since the number of parameters increases dramatically while there is only a limited number of training examples. Previous studies have usually divided solvent accessibility into two or three categories.

To find the optimal discretization, we experimented with HMM\_AA\_SA on Dataset 1, with each M or I state emitting a basic amino acid plus one letter of solvent accessibility. A greedy search for thresholds was used to find the best categorization. We first searched for the optimal threshold  $\alpha_1$  of dividing solvent accessibility into two categories. Next, we fixed the threshold  $\alpha_1$ , and tried various value of threshold  $\alpha_2$  to discretize into three categorizes.

*4.3.1.1 Discretization in Two Categories.* We first discretized solvent accessibility into two categories, i.e., buried (B) and exposed (E), using one threshold  $\alpha_1$ . A residue is in the buried (B) category if its relative solvent accessibility is less than  $\alpha_1$ , and exposed (E) otherwise. Using this discretization, every state of the hidden Markov model (referred to as HMM\_AA\_SA) emits one solvent accessibility letter (B or E) in addition to one letter of the amino acid. Thus, each state is associated with 40 emission probabilities corresponding to the 40 combinations of 20 amino acid letters and two solvent accessibility letters. We tried various values of  $\alpha_1$ , ranging from 0.05 to 0.9 in increments of 0.05. For each threshold, we evaluated HMM\_AA\_SA on the positive data set using three-fold cross-validation. The sensitivity from the three-fold cross-validation is shown in Figure 10. We also examined HMM\_AA\_SA's false positive rate by building an HMM\_AA\_SA using the positive set and then testing it against the negative data set. The results are also shown in Figure 10. Figure 10 shows that when the threshold increases

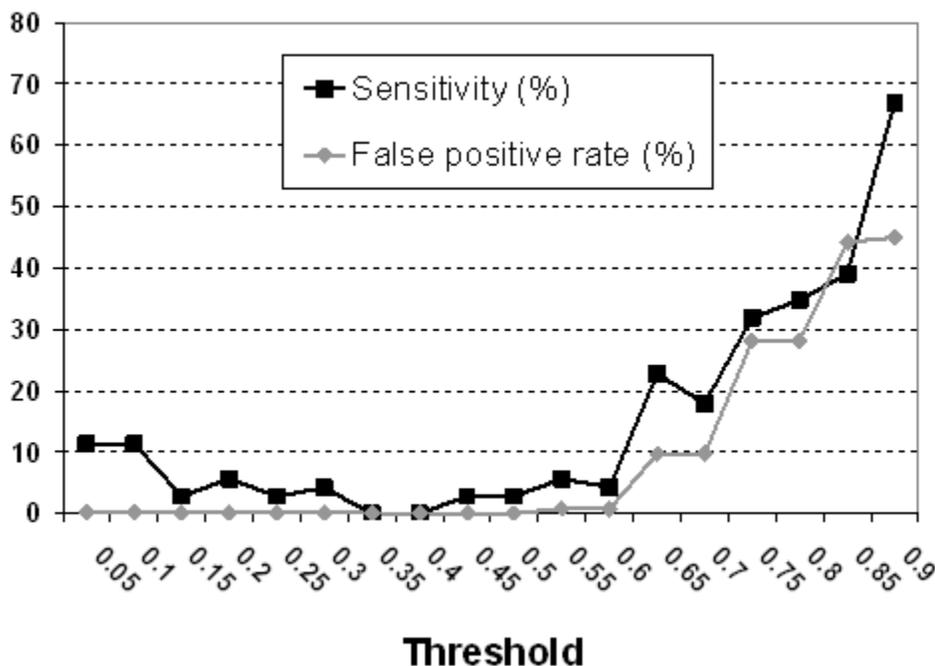


Figure 10. The performance of HMM\_AA\_SA with solvent accessibility being divided into two categories. Solvent accessibility is discretized into two categories (buried (B) and exposed (E)) using one threshold  $\alpha_1$ .

from 0.05 to 0.6, the sensitivity first decreases and then slightly increases, while the false positive rate remains at a very low level. When the threshold continues to increase, the sensitivity increases quickly, and the false positive rate also increases dramatically.

An ideal threshold should give a high sensitivity rate and low false positive rate.

Therefore, thresholds greater than 0.6 are not good choices because they introduce very high false positive rates. Focus is given to range from 0.05 to 0.6 wherein the false positive rate remains at a low level. As can be seen from the range, HMM\_AA\_SA achieves the highest sensitivity (11.4%) when the threshold takes 0.05 or 0.1. Thus, the threshold of 0.05 or 0.1 is the best choice for discretizing solvent accessibility into two categories for the HMM\_AA\_SA method.

Table 9. HMM\_AA\_SA Achieves Better Performance Than HMM\_AA by Dividing Solvent Accessibility into Two Discrete Categories.

Method	Sensitivity (%)	False positive rate (%)
HMM_AA	2.8	0
HMM_AA_SA ( $\alpha_1=0.05$ )	11.4	0.2
HMM_AA_SA ( $\alpha_1=0.1$ )	11.4	0.2

We also tested the performance of HMM\_AA on the same dataset and partition using three-fold cross-validation. Table 9 compares the performance of HMM\_AA with that of the HMM\_AA\_SA. The results show that HMM\_AA\_SA (row 3 and 4) achieves much higher sensitivity than HMM\_AA (row 2), while the false positive rate is remained at a low level.

*4.3.1.2 Discretization in Three Categories.* In this section, we explore the discretization that divides residue solvent accessibility into three categories: buried (B), medium (M) and exposed (E). To divide solvent accessibility into three discrete categories, two thresholds  $\alpha_1$  and  $\alpha_2$  are needed, with  $\alpha_1 < \alpha_2$ . A residue is in the buried (B) category if its relative solvent accessibility (RSA) is less than  $\alpha_1$ , medium (M) if  $\alpha_1 \leq \text{RSA} < \alpha_2$ , and exposed (E) if  $\text{RSA} \geq \alpha_2$ . Using this discretization, every state of HMM\_AA\_SA emits one solvent accessibility letter (B, M, or E) in addition to one amino acid letter. Thus, each state is associated with 60 emission probabilities corresponding to the 60 combinations of 20 amino acid letters and three solvent accessibility letters.

Since 0.1 and 0.05 were the best thresholds for dividing the solvent accessibility into two categories and 0.05 has been used by previous studies, we fixed  $\alpha_1=0.05$  and tried various values of  $\alpha_2$ , ranging from 0.1 to 0.9 with increments of 0.05 (see Figure 11). Since the number of negative examples is much larger than positive, we wanted to have a method that achieves very low specificity with comparatively high sensitivity, which narrowed down the  $\alpha_2$  in the range of [0.25, 0.55]. Among them, when  $\alpha_2$  is 0.25, 0.3 and 0.55, the sensitivity is comparatively high and specificity is low enough. We chose  $\alpha_2=0.25$  since it has been used in previous studies [176].

Table 10 shows the comparison among the HMM\_AA (row 2), the HMM\_AA\_SA that uses 0.05 as the threshold to divide solvent accessibility into two categories (row 3), and the HMM\_AA\_SA that uses (0.05, 0.25) as the thresholds to divide solvent accessibility into three categories (row 4). When comparing rows 3 and 4 with row 2, it is obvious that adding solvent accessibility into the HMM can improve its performance by greatly increasing sensitivity. At the same time, there is only a small increase in the false positive rate. In comparing rows 3 with 4, we can see that dividing solvent accessibility into three categories can further improve performance than can dividing it into just two categories.

We stopped further discretizing solvent accessibility since too many categories introduces the problem of insufficient training. In summary, HMM\_AA\_SA achieved best performances when solvent accessibility was discretized into three categories using threshold (0.05, 0.25).

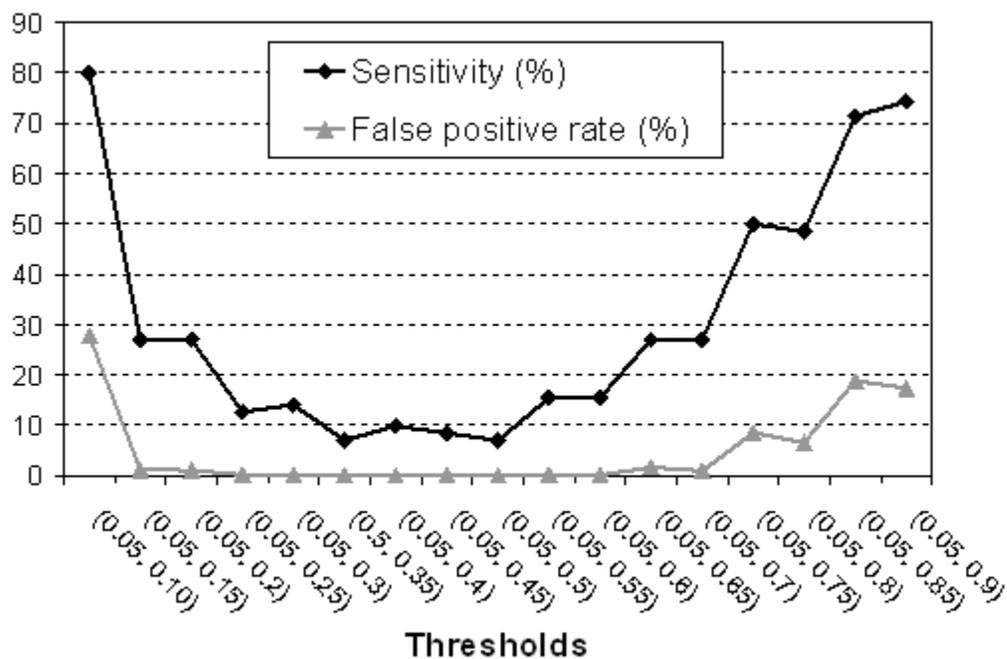


Figure 11. The performance of HMM\_AA\_SA with solvent accessibility being divided into three categories ( $\alpha_1, \alpha_2$ ) with  $\alpha_1 = 0.05$ .

Table 10. HMM\_AA\_SA's Performance Is Improved by Dividing Solvent Accessibility into Three Discrete Categories.

Method	Sensitivity (%)	False positive rate (%)
HMM_AA	2.8	0
HMM_AA_SA ( $\alpha_1=0.05$ )	11.4	0.2
HMM_AA_SA ( $\alpha_1=0.05, \alpha_1=0.25$ )	12.7	0.2

### *4.3.2 Constructing Profiles For Each HTH Protein Family Increases the Prediction Accuracy*

Although HMM\_AA\_SA achieved significant improvements over HMM\_AA, the prediction accuracies of both methods are still very low. This can be explained by the fact that both methods were built from the full alignment of HTH proteins from different families, and enormous variations in sequences have been observed among HTH motifs belonging to different families. Different profiles are needed for different families that contain HTH motifs. In Pfam, a profile HMM (HMM\_AA) is constructed for each protein family. HTH profiles from Pfam can be used to identify HTH proteins from their own families, but they can rarely recognize HTH protein from different families. Therefore, the goal in this study is to construct profiles using HMM\_AA\_SA for each HTH protein family so that they not only can identify HTH protein from their own families, but also are capable of finding HTH proteins from remote families and possibly new HTH proteins that do not fall into any current available HTH protein families.

*4.3.2.1 HMM\_AA\_SA Recognizes HTH Motifs from Same Families with Higher Sensitivity.* We evaluated the ability of HMM\_AA\_SA to identify HTH motifs on Dataset 2 since HTH proteins in Dataset 2 are grouped into 12 families. We divided the solvent accessibility of residues into three discrete states: Buried (B), Medium (M), and Exposed (E) using threshold (0.05, 0.25). The sensitivity of HMM\_AA\_SA was evaluated for each family individually using a three-fold cross-validation. The sensitivity averaged over the 12 families is reported in Table 11. For direct comparison, HMM\_AA was also evaluated using the same data sets. The results (see Table 11, row 3) show that HMM\_AA\_SA can identify HTH motifs with 94.9% sensitivity, while HMM\_AA only achieves a sensitivity

of 89.2% (see Table 11, row 2). The false positive number was also examined. For each family, one HMM\_AA\_SA was built using all the positive sequences in that family. Next, the HMM\_AA\_SA was used to scan the sequences in the negative set. The total number of false positives made by the 12 HMM\_AA\_SAs is reported (see Table 11). The false positive number of HMM\_AA was evaluated using the same approach. The 12 HMM\_AA\_SAs make two false positive predictions in total (see Table 11, row 3). The HMM\_AA makes no false positive prediction (see Table 11, row 2). Consider that there are 2,497 negative examples; the false negative rates of the two methods are comparable.

The results show that adding solvent accessibility into the hidden Markov model can increase sensitivity by 5.7 %, while there is only a slight increase in false positive number.

*4.3.2.2 Using Reduced Alphabets Further Improves Performance.* It is well known that there are similarities among the naturally occurring 20 amino acids. Clustering amino acids into similar groups and using reduced alphabets to represent the amino acids can reduce the complexity of protein sequence. In this study, an additional benefit of using reduced alphabets is that it can reduce the number of parameters (emission probabilities) in the models. We tried all sets of reduced alphabets developed by previous studies (see Table 8). The results (see Table 11, row 4-9) show that all the HMM\_AA\_SA methods using reduced alphabets except Func\_8 achieve higher sensitivity than the HMM\_AA\_SA using the standard 20-letter alphabet, while there is only a small increase in the false positive number.

Table 11. Including Solvent Accessibility Information into the Model and Using Reduced Alphabet Increase Performance in Identifying HTH motifs.

		Average sensitivity (%)	False positive number
HMM_AA (using the standard 20-letter alphabet)		89.2	0
HMM_AA_SA (using the standard 20-letter alphabet)		94.9	2
HMM_AA_SA using reduced alphabets	Mur_15	95.3	3
	Mur_10	95.7	5
	Mur_8	95.1	6
	Chem_6	95.4	2
	Func_8	94.2	2
	Li_10	95.8	7

4.3.2.3 *HMM\_AA\_SA Recognizes HTH Motifs from Remote Families.* Large sequence variations have been observed among HTH motifs belonging to different families. However, these HTH motifs often share highly similar structure and function. In the previous section, we tested HMM\_AA\_SA's ability to identify HTH motifs by building a HMM for each family of HTH motifs. However, to do so, for each family we needed to assemble a data set big enough to build the model. Furthermore, given the enormous variation in sequence among the observed HTH motifs, it is likely that there are some HTH motifs that do not fit into any currently known HTH families. To efficiently identify these novel HTH motifs, generic HTH models that can be used to identify all types of HTH motif are needed. Some structure-based methods [148, 157] address this need by building structural templates that capture the structural features essential to the function of HTH motifs. Addressing this problem using sequence-based

Table 12. HMM\_AA\_SA Recognize More HTH Motifs from Other Families.

	Use the whole conserved fragment to build HMMs	Only the HTH fragments are used to build HMMs.	
	HMM_AA (using the standard 20-letter alphabet)	HMM_AA_SA (using reduced alphabet Mur_15)	HMM_AA_SA (using reduced alphabet Mur_15)
True Positive Number	145	448	699
False Positive Number	0	3	5

methods is intimidated by the low similarity among different families of HTH motifs. We have proposed an HMM method for the identification of HTH motifs by combining amino acid sequence and predicted solvent accessibility. Since the solvent accessibility is predicted from protein sequence, the method is a sequence-based method that requires only protein sequence as input. Our method distinguishes itself from other sequence-based methods by incorporating predicted solvent accessibility.

For each family, we build an HMM\_AA\_SA using the sequences belonging to that family and use it to search for HTH motifs in the sequences from the 11 remaining families. In this way, each sequence in 12 families is scanned using 11 HMM\_AA\_SAs. If a sequence is predicted to be an HTH motif by at least one of the 11 HMM\_AA\_SAs from other families, it is said to be successfully recognized. In this experiment (referred as a cross-family experiment), we consider the HMM\_AA\_SA using reduced alphabet Mur\_15, because the results from the previous section show that it achieves high sensitivity, with relative low false positive number. For direct comparison, HMM\_AA

was built and used to search HTH motifs in the same setting. The results (see Table 12) show that HMM\_AA (column 2) can only recognize 145 HTH motifs, while HMM\_AA\_SA (column 3) can successfully recognize 448 HTH motifs in the cross-family examinations. At the same time, there is only a slight increase (from 0 to 3) in the false positive number. The results show that adding solvent accessibility into the HMM and using reduced alphabet Mur\_15 enhance the method's ability to recognize HTH motifs from other family. This indicates that HMM\_AA\_SA using reduced alphabets can capture the common features shared by different family of HTH motifs.

#### *4.3.2.4 Using an HTH Fragment to Build the HMM Further Improves*

*Performance.* In the results presented above, the HMMs were built using the conserved sequence fragments of each family obtained from the Pfam seed alignment. Further examination shows that the conserved fragment of each family contains a longer sequence than the fragment directly involved in the HTH motif. A typical HTH motif consists of a bundle of three helices: the recognition helix that forms base-specific interactions with the DNA and the two helices preceding it [147]. The three helices are connected by two irregular regions. In this section, we removed the conserved sequence fragments that are not part of the HTH motif and used the remaining HTH fragments to build HMMs. We evaluated the HMM's ability to recognize HTH sequences from other families. One HMM\_AA\_SA was built for each of the 12 family using only the HTH fragments. For direct comparison, we used reduced alphabet Mur\_15 to reduce the complexity of the sequence. Each model was used to scan the protein sequences from other families. A HTH sequence is considered being successfully recognized if it is

identified by at least one HMM\_AA\_SA from other families. The results (see Table 12, column 4) show that the HMM\_AA\_SA built from HTH fragments can successfully identify 699 HTH motifs, while the HMM\_AA\_SA built from the whole conserved sequence (see Table 12, column 3) can only identify 448. Note that the false positive numbers of the two methods, 5 and 3, respectively (see Table 12), are still comparable. These results show that using only HTH fragments to build a hidden Markov model can significantly increase HMM\_AA\_SA's ability to identify HTH motifs from other families.

#### 4.4 Discussion

In this study, we investigated the possibility of identifying HTH motifs by modifying the traditional profile hidden Markov model. The novelties of our method include the following:

1. For each M or I state, instead of emitting only amino acid information, HMM\_AA\_SA emits one amino acid and its solvent accessibility. Since the solvent accessibility is predicted from the amino acid sequence, the method requires only protein sequences as input. Thus, our method has much broader applications than structure-based methods.
2. Solvent accessibility is discretized into three categories (Buried (B), Medium (M), and Exposed (E)), wherein discretization thresholds are chosen by a systematic analysis.
3. When solvent accessibility is incorporated in the HMM, each state of HMM\_AA\_SA is associated with 60 emission probabilities corresponding to the 60 combinations of 20 amino acid letters and three solvent accessibility

letters. In a standard hidden Markov model (referred as HMM\_AA) that emits only amino acids, each state is associated with only 20 emission probabilities corresponding to the 20 amino acids. Compared with HMM\_AA, one disadvantage of HMM\_AA\_SA is that it has a larger set of parameters (emission probabilities), which usually requires a larger set of training data to estimate. In this study, a reduced alphabet was used to reduce the alphabet size of amino acids and number of parameters (emission probabilities) in the models.

The proposed method has been applied to predict HTH motifs from protein sequences. The results show that adding solvent accessibility into the model can increase the sensitivity, while the number of false positives is still small. When a reduced alphabet was used instead of standard 20-letter alphabet, the prediction performance was further improved. We also evaluated the proposed method's ability to identify HTH motifs across families. HMM\_AA\_SA was built for each HTH family and used to scan the sequences from other HTH families. The results show that 448 out of 2,715 HTH motifs can be recognized by the HMM\_AA\_SAs built from other families. This number is improved to 699 when only HTH fragments are used to build HMM\_AA\_SA. In comparison, HMM\_AA can only identify 145 HTH motifs from other families.

Identification of HTH motifs is a challenging problem since protein sequences sharing low similarity have been found to form the same or similar HTH structures and perform the same functions. HMM has been used to develop sequence pattern for each family of HTH motifs [168]. However, to do so, for each family one must assemble a

data set big enough to build the model. Furthermore, it is likely that there are some HTH motifs that do not fit into any currently known HTH families. To efficiently identify HTH motifs, generic HTH models that can capture the essential characters shared by different HTH motifs are needed. The low similarity among HTH families makes it seemingly impossible for sequence-based methods to identify HTH motifs across families. However, by including solvent accessibility information that is predicted from sequence information, an HMM\_AA\_SA built from one HTH family can successfully identify some HTH motifs from other families, which suggests that our method can capture some common characters shared by different families of HTH motifs. We can use HMM\_AA\_SA to build a model for each known family of HTH motifs. The resulting models would not only have higher sensitivity in identifying HTH motifs from the same family but also have higher chance of recognizing new HTH motifs that do not fall into any currently known family. If a query protein is predicted to be HTH by any of current models, it is predicted to have HTH motifs.

Using reduced alphabets not only reduces the number of parameters in the model but also reduces the complexity of the protein sequences, which will help to identify the features essential to the function. Different amino acids can perform a similar function because they have similar physiochemical properties or they are close in the evolution. Clustering the amino acids based on these properties can produce reduced alphabets without losing information for function or structure identification. Murphy et al. [174] clustered amino acids into groups based on physiochemical properties and obtained a series of reduced alphabets with size ranging from 2-15. Their results show that the

reduced alphabets with a size around 10 can be used to detect structural homolog with little loss in necessary information. In this study, we tried three reduced alphabets from their study: Mur\_15, Mur\_10, and Mur\_8. The results show that these reduced alphabets, indeed, increase sensitivity in identifying HTH motifs with no increase or little increase in false positive number. The reason for the improvement may reside in the fact that these reduced alphabets were developed for detecting structural homolog. Although different families of HTH motifs have low similarity in sequence, they share the same structure. A reduced alphabet that can detect structural homolog should be helpful in identifying HTH motifs that have the same structure.

#### **4.4 Conclusion**

In summary, we present a hidden Markov model method (referred as HMM\_AA\_SA) for identification of HTH motifs. The method models both amino acid sequence and solvent accessibility. In both match (M) and insertion (I) states, the model emits one letter of an amino acid and one letter of solvent accessibility. The results show that adding solvent accessibility into the model can dramatically increase its sensitivity in identifying HTH motifs, with just a slight increase in the number of false positives. The developed method can be further improved by using a reduce alphabet such as Mur\_15 [174]. The resulting model not only has a higher sensitivity in identifying HTH motifs from the same family but also a higher chance of recognizing new HTH motifs from proteins that do not fall into any of currently known families.

#### **4.5 Future Work**

Besides predicted solvent accessibility, features, such as predicted secondary structure, hydrophobicity and charges, can also be useful in identifying HTH motifs. By appropriate modifications, we also investigate the possibility of using the proposed model to find other types of motifs, such as helix-loop-helix (HLH) motifs and helix-hairpin-helix motifs (HHH).

## CHAPTER 5

## CONCLUSION

Proteins are large molecules made of amino acids connected by peptide bonds in a linear pattern. They are important elements of all organisms and participate in almost every biological process in cells. Driven by hydrogen bonds, ionic interactions, hydrophobic packing and other forces, proteins fold into certain conformations to perform their respective functions. Thus, structural information of proteins is very helpful in studying and predicting protein functions. However, for most proteins, their 3-dimensional structures are not available due to experimental difficulties. Therefore, it is important to predict protein functions and functional sites using only protein sequence information.

There are several challenges in constructing classification methods for predicting protein function and functional sites. To investigate these challenges, this dissertation addresses three important problems with biological, clinical, and pharmaceutical significance. These problems are: the discovery of transmembrane beta-barrel proteins in gram-negative bacterial proteomes, the identification of deleterious non-synonymous single nucleotide polymorphisms, and the identification of helix-turn-helix motifs from protein sequences. For each problem, we compiled a set of candidate features derived from protein sequences. Next, an appropriate feature selection approach was used to choose the most relevant subset of features, which were then input to the suitable machine learning algorithm to build the classifier. The predictors were trained and evaluated on benchmark datasets that were either compiled in this study or derived from

previous studies. Specifically, the contributions for solving each problem fall into the following:

1. We developed a K-NN method that can efficiently and effectively identify TMB proteins from gram-negative proteomes. We first constructed datasets of TMB proteins and non-TMB proteins. Non-TMB proteins were divided into six groups based on their subcellular locations. For each protein, compositions of 20 amino acids and 400 di-peptides were calculated. After a feature selection process, 19 amino acids and 24 di-peptides were left. By including homologous sequences and using compositions of selected amino acids and di-peptides to calculate weighted Euclidian distances, the K-nearest neighbor method can discriminate TMB proteins from non-TMB proteins with high performances and fast speed.
2. We developed a decision-tree based method that is capable of identifying deleterious nsSNPs with high performances. For each mutation site, a set of 686 features were derived from protein sequences to describe various differences between wild-type and mutant-type residues and their surrounding environment. After a systematic greedy feature selection, ten features were chosen and then fed into the decision tree to make predictions. The developed method only requires information from protein sequences; therefore, it is applicable to find disease related nsSNPs in a proteomic scale.
3. We developed a profile HMM (named HMM\_AA\_SA) that can successfully identify HTH motifs from protein sequences. The proposed model differs

from a traditional profile HMM (HMM\_AA) by allowing match (M) and insertion (I) states to emit both a letter of amino acid and a letter of solvent accessibility. Solvent accessibility of each residue is predicted from amino acid sequence and discretized into three categories: buried (B), medium (M) and exposed (E). The method achieved significant improvement over HMM\_AA in identifying HTH motifs. We also investigated various reduced alphabets instead of the standard 20-letter amino acid alphabet. When a reduced alphabet such as Mur\_15 [174] was used, the prediction performance was further improved. The final developed method not only can find HTH motifs from protein sequences, but also is capable of identifying HTH motifs from remote proteins that have limited similarity to the current known DNA-binding proteins.

In summary, this study successfully solved three important bioinformatics problems by systematically developing machine learning methods that uncover attribute-class relationships between sequence-based features and protein functions.

## REFERENCES

- 1 Aaldi, P. and Bunnak, S. *Bioinformatics: The Machine Learning Approach*, 2nd ed. MIT Press, 2001.
- 2 NCBI. 2001. <http://www.ncbi.nlm.nih.gov/>.
- 3 Alpaydin, E. *Introduction to Machine Learning (Adaptive Computation and Machine Learning)*. MIT Press, 2004.
- 4 Bishop, C. M. *Pattern Recognition and Machine Learning*. Springer, 2007.
- 5 Bohr, H., Bohr, J., Brunak, S., Cotterill, R.M.J., Lautrup, B., Nørskov, L., Olsen, O.H., and Petersen, S.B. Protein secondary structures and homology by neural networks: The alpha-helices in rhodopsin. *FEBS Letters* 241, 1-2 (1988), 223–228.
- 6 Rost, B. and Sander, C. Prediction of protein secondary structure at better than 70% accuracy. *J. Molecular Biology* 232, 2 (1993), 584–599.
- 7 Holley, H. and Karplus, M. Protein secondary structure prediction with a neural network, In *Proc. 86<sup>th</sup> National Academy of Sciences*, 1989.
- 8 Claros, M.G., Brunak, S., and von Heijne, G. Prediction of n-terminal protein sorting signals. *Current Opinion in Structural Biology* 7, 3 (1997), 394-398.
- 9 Nielsen, H., Engelbrecht, J., Brunak, S., and von Heijne, G. Identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites. *Protein Engineering* 10, 1 (1997), 1–6.
- 10 Brunak, S., Engelbrecht, J., and Knudsen, S. Prediction of human mRNA donor and acceptor sites from the DNA sequence. *J. Molecular Biology* 220, 1 (1991), 49–65.
- 11 Reese, M.G., Eeckman, F.H., Kulp, D., and Haussler, D. Improved splice site detection in Genie. *J. of Computational Biology* 4, 3 (1997), 311-323.
- 12 Smith, L., Yeganova, L., and Wilbur, W. J. Hidden Markov models and optimized sequence alignments. *Computational Biology and Chemistry* 27, 1 (2003), 77-84.
- 13 Park, J., Karplus, K., Barrett, C., Hughey, R., Haussler, D., Hubbard, T., and Chothia, C. Sequence comparisons using multiple sequences detect three times as many remote homologues as pairwise method. *J. Molecular Biology* 284, 4 (1998), 1201-1210.

- 14 Alexandersson, M., Cawley, S., and Pachter, L. SLAM: Cross-species gene finding and alignment with a generalized pair hidden Markov model. *Genome Research* 13, 3 (2003), 496-502.
- 15 Truong, K. and Ikura, M. Identification and characterization of subfamily-specific signatures in a large protein superfamily by a hidden Markov model approach. *BMC Bioinformatics* 3 (2002), 1.
- 16 Karplus, K., Barrett, C., Cline, M., Diekhans, M., Grate, L., and Hughey, R. Predicting protein structure using only sequence information. *Proteins: Structure, Function and Bioinformatic*, 37, suppl 3 (1999), 121-125.
- 17 Karplus, K., Sjölander, K., Barrett, C., Cline, M., Haussler, D., Hughey, R., Holm, L., and Sander, C. Predicting protein structure using hidden Markov model. *Proteins* 29, Suppl 1 (1997), 134-139.
- 18 Tusnady, G.E. and Simon, I. Principles governing amino acid composition of integral membrane proteins: Application to topology prediction. *J. of Molecular Biology* 283, 2 (1998), 489–506.
- 19 Krogh, A., Larsson, B., von Heijne, G., and Sonnhammer, E. L. Predicting transmembrane protein topology with a hidden Markov model: Application to complete genomes. *J. Molecular Biology* 305, 3 (2001), 567–80.
- 20 Bigelow, H. and Rost, B. PROFtmb: A web server for predicting bacterial transmembrane beta barrel proteins. *Nucleic Acids Research* 34, Web Server Issue (2006), W186-188.
- 21 Burge, C. and Karlin, S. Prediction of complete gene structure in human genomic DNA. *J. of Molecular Biology* 268, 1 (1997), 78-94.
- 22 Meyer, I. M. And Durbin, R. Comparative ab initio prediction of gene structure using pair HMMs. *Bioinformatics* 18, 10 (2002), 1309-1318.
- 23 Douguet, D., Chen, H.C., Tovchigrechko, A., and Vakser, I. A. DOCKGROUND resource for studying protein-protein interfaces. *Bioinformatics* 22, 21 (2006), 2612-2618.
- 24 Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., and Bourne, P. E. The protein data bank. *Nucleic Acids Research* 28, 1 (2000), 235–242.
- 25 Kawashima, S., Ogata, H., and Kanehisa, M. AAindex: Amino acid index database. *Nucleic Acids Research* 27, 1 (1999), 368-369.

- 26 Altschul, S. F., Madden, T. L., Schäffer, A. A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D. J. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Research* 25, 17 (1997), 3389-3402.
- 27 Cuff, J. A., Clamp, M. E., Siddiqui, A. S., Finlay, M., and Barton, G. J. JPred: A consensus secondary structure prediction server. *Bioinformatics* 14, 10 (1998), 892-893.
- 28 Dobson, R. J., Munroe, P. B., Caulfield, M. J., and Saqi, M.A.S. Predicting deleterious nsSNPs: An analysis of sequence and structural attributes. *BMC Bioinformatics* 7 (2006), 217.
- 29 Momen-Roknabadi, A., Sadeghi, M., Pezeshk, H., and Marashi, S. A. Impact of residue accessible surface area on the prediction of protein secondary structures. *BMC Bioinformatics* 9 (2008), 357.
- 30 von Heijne, G. The distribution of positively charged residues in bacterial inner membrane proteins correlates with the trans-membrane topology. *The EMBO J.* 5, 11 (1986): 3021-3027.
- 31 Nakashima, H., Nishikawa, K., and Ooi, T. The folding type of a protein is relevant to the amino acid composition. *J. Biochemistry* 99, 1 (1986),152-162.
- 32 Nakashima, H., and Nishikawa, K. Discrimination of intracellular and extracellular proteins using amino acid composition and residue-pair frequencies. *J. Molecular Biology* 238, 1 (1994), 54-61.
- 33 Chou, K.C. and Cai, Y. D. Prediction and classification of protein subcellular location-sequence-order effect and Pseudo amino acid composition. *J. Cellular Biochemistry* 90, 6 (2003), 1250-1260.
- 34 Saeys, Y., Inza, I., and Larrañaga, P. A review of feature selection techniques in bioinformatics. *Bioinformatics* 23, 19 (2007), 2507-2517.
- 35 Witten, I. H. and Frank, E. *Data Mining: Practical Machine Learning Tools and Techniques*, 2<sup>nd</sup> ed. Morgan Kaufmann, 2005.
- 36 Ben-Bassat, M. Classification pattern recognition and reduction of dimensionality. In *Handbook of Statistics II*, P. Krishnaiah and L. Kanal, Eds. Elsevier, 1983, 773–791.
- 37 Salzberg, S. L., Delcher, A. L., Kasif, S. and White, O. (1998) Microbial gene identification using interpolated markov models. *Nucleic Acids Research* 26, 2 (1988), 544–548.

- 38 Ye, Z-Q., Zhao, S-Q., Gao, G., Liu, X-Q., Langlois, R.E., Lu, H., and Wei, L. Finding new structural and sequence attributes to predict possible disease association of single amino acid polymorphism (SAP). *Bioinformatics* 23, 12 (2007), 1444-1450.
- 39 Park, K. J., Gromiha, M.M., Horton, P., and Suwa, M. Discrimination of outer membrane proteins using support vector machines. *Bioinformatics* 21, 23 (2005), 4223-4229.
- 40 Bø, T. H. and Jonassen, I. New feature subset selection procedures for classification of expression profiles. *Genome Biology* 3, 4 (2002), research 0017.1-research0017.11.
- 41 Kittler, J. Feature Set Search Algorithms. In *Pattern Recognition and Signal Processing*, C. H. Chen, Ed. Springer, 1978, 41-60.
- 42 Xu, L., Yan, P., and Chang, T. Best first strategy for feature selection. In *Proc. 9<sup>th</sup> Int'l Conference on Pattern Recognition*, 1988.
- 43 Siedlecky, W. and Sklansky, J.(1998) On automatic feature selection. *Int'l J. Pattern Recognition and AI* 2, 2 (1988), 197-220.
- 44 Skalak, D. B. Prototype and feature selection by sampling and random mutation hill climbing algorithms. In *Proc. 11<sup>th</sup> Int'l Conference on Machine Learning*, 1994.
- 45 Holland, J. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.
- 46 von Heijne, G. Recent advances in the understanding of membrane protein assembly and function. *Quarterly Reviews of Biophysics* 32, 4 (1999), 285-307.
- 47 Schulz, G.E. Beta-barrel membrane proteins. *Current Opinion in Structural Biology* 10, 4 (2000), 443-447.
- 48 Wimley, W. C. Towards genomic identification of beta-barrel membrane proteins: Composition and architecture of known structures. *Protein Science* 11, 2 (2002), 301-312.
- 49 Koebnik, R., Locher, K. P., and van Gelder, P. Structure and function of bacterial outer membrane proteins: Barrels in a nutshell. *Molecular Microbiology* 37, 2 (2000), 239-253.
- 50 Valavanis, I. K., Bagos, P.G., and Emiris, I. Z. Beta-Barrel transmembrane proteins: Geometric modeling, detection of transmembrane region, and structural properties. *Computational Biology and Chemistry* 30, 6 (2006), 416-424.

- 51 von Heijne, G. Membrane protein structure prediction. Hydrophobicity analysis and the positive-inside rule. *J. Molecular Biology* 225, 2 (1992), 487-494.
- 52 Berven, F. S., Flikka, K., Jensen, H. B., and Eidhammer, I. BOMP: A program to predict integral beta-barrel outer membrane proteins encoded within genomes of Gram-negative bacteria. *Nucleic Acids Research* 32, Web Server Version (2004), W394-399.
- 53 Bagos, P.G, Liakopoulos, T. D, Spyropoulos, I. C., and Hamodrakas, S. J. A hidden Markov model method, capable of predicting and discriminating beta-barrel outer membrane proteins. *BMC Bioinformatics* 5 (2004), 29.
- 54 Gnanasekaran, T. V., Peri, S., Arockiasamy, A., and Krishnaswamy, S. Profiles from structure based sequence alignment of porins can identify beta stranded integral membrane proteins. *Bioinformatics* 16, 9 (2000), 839-842.
- 55 Schleiff, E., Eichacker, L. A., Eckart, K., Becker, T., Mirus, O., Stahl, T., and Soll, J. Prediction of the plant beta-barrel proteome: A case study of the chloroplast outer envelope. *Protein Science* 12, 4 (2003), 748-759.
- 56 Zhai, Y. and Saier, M.H., Jr. The beta-barrel finder (BBF) program, allowing identification of outer membrane beta-barrel proteins encoded within prokaryotic genomes. *Protein Science* 11, 9 (2002), 2196-2207.
- 57 Liu, Q., Zhu, Y., Wang, B., and Li, Y. Identification of beta-barrel membrane proteins based on amino acid composition properties and predicted secondary structure. *Computational Biology and Chemistry*, 27, 3 (2003), 355-361.
- 58 Deng, Y., Liu, Q. and Li, Y.-X. Scoring hidden Markov models to discriminate beta-barrel membrane proteins. *Computational Biology and Chemistry* 28, 3 (2004), 189-194.
- 59 Garrow, A. G., Agnew, A., and Westhead, D. R. TMB-Hunt: An amino acid composition based method to screen proteomes for beta-barrel transmembrane proteins. *BMC Bioinformatics* 6 (2005), 56.
- 60 Garrow, A.G., Agnew, A., and Westhead, D. R. TMB-Hunt: A web server to screen sequence sets for transmembrane beta-barrel proteins. *Nucleic Acids Research* 33, Web Server Issue (2005), W188-192.
- 61 Gromiha, M.M. and Suwa, M. Discrimination of outer membrane proteins using machine learning algorithms. *Proteins* 63, 4 (2006), 1031-1037.
- 62 Gromiha, M.M. and Suwa, M. Influences of amino acids properties for discriminating outer membrane proteins at better accuracy. *Biochim Biophys Acta*. 1764, 9 (2006), 1493-1497.

- 63 Gromiha, M.M. and Yabuki, Y. Functional discrimination of membrane proteins using machine learning techniques. *BMC Bioinformatics* 9 (2008), 135.
- 64 Ou, Y., Gromiha, M.M., Chen, S., and Suwa, M. TMBETADISC-RBF: discrimination of  $\beta$ -barrel membrane proteins using RBF networks and PSSM profiles. *Computational Biology and Chemistry* 32, 3 (2008), 227-231.
- 65 Mirus, O. and Schleiff, E. Prediction of  $\beta$ -barrel membrane proteins by searching for restricted domains. *BMC Bioinformatics* 6 (2005), 254.
- 66 Vogel, H. and Jaehnig, F. Models for the structure of outer-membrane proteins of *Escherichia coli* derived from Raman spectroscopy and prediction methods. *J. Molecular Biology* 190, 2 (1986), 191-199.
- 67 Schirmer, T. and Cowan, S. W. Prediction of membrane-spanning beta-strands and its application to maltoporin. *Protein Science* 2, 8 (1993), 1361-1363.
- 68 Gromiha, M. M. and Ponnuswamy, P. K. Prediction of transmembrane beta-strands from hydrophobic characteristics of proteins. *Int'l J. of Peptide and Protein Research* 42, 5 (1993), 420-431.
- 69 Gromiha, M.M., Majumdar, R., and Ponnuswamy, P. K. Identification of membrane spanning beta strands in bacterial porins. *Protein Engineering, Design & Selection* 10, 5 (1997), 497-500.
- 70 Diederichs, K., Freigang, J., Umhau, S., Zeth, K., and Breed, J. Prediction by a neural network of outer membrane beta-strand protein topology. *Protein Science* 7, 11 (1998), 2413-2420.
- 71 Jacoboni, I., Martelli, P. L., Fariselli, P., De Pinto, V., and Casadio, R. Prediction of the transmembrane regions of beta-barrel membrane proteins with a neural network-based predictor. *Protein Science* 10, 4 (2001), 779-787.
- 72 Gromiha, M. M., Ahmad, S., and Suwa, M. Neural network-based prediction of transmembrane beta-strand segments in outer membrane proteins. *J. Computational Chemistry* 25, 5 (2004), 762-767.
- 73 Randall, A., Cheng, J., Sweredoski, M., and Baldi, P. TMBpro: Secondary structure, beta-contact and tertiary structure prediction of transmembrane beta-barrel proteins. *Bioinformatics* 24, 4 (2008), 513-520.
- 74 Natt, N. K., Kaur, H., and Raghava, G. P. Prediction of transmembrane regions of beta-barrel proteins using ANN- and SVM-based methods. *Proteins* 56, 1 (2004), 11-18.

- 75 Ahn, C.S., Yoo, S.J., and Park, H.S. Prediction for beta-barrel transmembrane protein region using HMM. *KISS* 30, 2 (2003), 802-804.
- 76 Bagos, P.G., Liakopoulos, T.D., Spyropoulos, I.C., and Hamodrakas, S. J. PRED-TMBB: A web server for predicting the topology of beta-barrel outer membrane proteins. *Nucleic Acids Research* 32, Web Server Issue (2004), W400-404.
- 77 Martelli, P. L., Fariselli, P., Krogh, A., and Casadio, R. A sequence-profile-based HMM for predicting and discriminating beta barrel membrane proteins. *Bioinformatics* 18, Suppl 1 (2002), S46-53.
- 78 Diao, Y., Ma, D., Wen, Z., Yin, J., Xiang, J., and Li, M. Using pseudo amino acid composition to predict transmembrane regions in protein: cellular automata and Lempel-Ziv complexity. *Amino Acids* 34, 1 (2008), 111-117.
- 79 Waldispühl, J., Berger, B., Clote, P., and Steyaert, J. M. transFold: A web server for predicting the structure and residue contacts of transmembrane beta-barrels. *Nucleic Acids Research* 34, Web Server Issue (2006), W189-W193.
- 80 Bagos, P. G., Liakopoulos, T. D., and Hamodrakas, S. J. Evaluation of methods for predicting the topology of beta-barrel outer membrane proteins and a consensus prediction method. *BMC Bioinformatics* 6 (2005), 7.
- 81 Krogh, A. Hidden Markov models for labeled sequences. In *Proc. 12<sup>th</sup> ISPR Int'l Conference on Pattern Recognition*, 1994.
- 82 Rabiner, L. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE* 77, 2 (1989), 257-286.
- 83 Krogh, A. Two methods for improving performance of a HMM and their application for gene finding. In *Proc. 5<sup>th</sup> International Conference on Intelligent Systems for Molecular Biology*, 1997.
- 84 Murzin, A. G., Brenner, S. E., Hubbard, T. and Chothia, C. SCOP: A structural classification of proteins database for the investigation of sequences and structures. *J. Molecular Biology* 247, 4 (1995), 536-540.
- 85 Saier, M. H. Jr, Tran, C.V., and Barabote, R.D. TCDB: The transporter classification database for membrane transport protein analyses and information. *Nucleic Acids Research* 34, Database Issue (2006), 181-186.
- 86 Rey, S., Acab, M., Gardy, J. L., Laird, M. R., deFays, K., Lambert, C., and Brinkman, F. S. PSORTdb: A protein subcellular localization database for bacteria. *Nucleic Acids Research* 33, Database Issue (2005), D164-168.
- 87 Mitchell, T. M. *Machine Learning*. McGraw-Hill, 1997.

- 88 Bentley, J. L. Multidimensional binary search trees used for associative searching. *Commun. of the ACM* 18, 9 (1975), 509-517.
- 89 Friedman, J., Bentley, J., and Finkel, R. An algorithm for finding best matches in logarithmic expected time. *ACM Trans. Mathematical Software* 3, 3 (1977), 209-226.
- 90 Moore, A.W. and Pelleg, D. X-means: Extending k-means with efficient estimation of the number of clusters. In *Proc. 17<sup>th</sup> Int'l Conference on Machine Learning*, 2000.
- 91 Baldi, P., Brunak, S., Chauvin, Y., Andersen, C.A.F., and Nielsen, H. Assessing the accuracy of prediction algorithms for classification: An overview. *Bioinformatics* 16, 5 (2000), 412-424.
- 92 Wimley, W.C. The versatile beta-barrel membrane protein. *Current Opinion in Structural Biology* 13, 4 (2003), 404-411.
- 93 Juncker, A. S., Willenbrock, H., von Heijne, G., Brunak, S., Nielsen, H., and Krogh, A. Prediction of lipoprotein signal peptides in Gram-negative bacteria. *Protein Science* 12, 8 (2003), 1652-1662.
- 94 Chou, K.C. Prediction of protein cellular attributes using pseudo-amino-acid-composition. *Proteins: Structure, Function, and Genetics* 43, 3 (2001), 246-255.
- 95 Chou, K.C. A new branch of proteomics: Prediction of protein cellular attributes. In *Gene Cloning and Expression Technologies*, P.W. Weinrer and Q. Lu, Eds. Eaton Publishing, 2002, 57-70.
- 96 Collins, F. S., Brooks, L. D., and Chakravarti, A. A DNA polymorphism discovery resource for research on human genetic variation. *Genome Research* 8, 12 (1998), 1229-1231.
- 97 Stenson, P. D., Ball, E. V., Mort, M., Phillips, A. D., Shiel, J. A., Thomas, N.S.T, Abeyasinghe, S., Krawczak, M., and Cooper, D.N. Human gene mutation database (HGMD): 2003 update. *Human Mutation* 21, 6 (2003), 577-581.
- 98 Ng, P.C. and Henikoff, S. Predicting deleterious amino acid substitutions. *Genome Research* 11, 5 (2001), 863-874.
- 99 Ng, P.C. and Henikoff, S. Accounting for human polymorphisms predicted to affect protein function. *Genome Research*, 12, 3 (2002), 436-446.
- 100 Ng, P.C. and Henikoff, S. SIFT: Predicting amino acid changes that affect protein function. *Nucleic Acids Research* 31, 13 (2003), 3812-3814.

- 101 Bell, J. Predicting disease using genomics. *Nature* 429, 6990 (2004), 453-456.
- 102 Sherry, S. T., Ward, M. H., Kholodov, M., Baker, J., Phan, I., Smigielski, E.M., and Sirotkin, K. dbSNP: The NCBI database of genetic variation. *Nucleic Acids Research* 29, 1 (2001), 308-311.
- 103 Fredman, D., Munns, G., Rios, D., Sjöholm, F., Siegfried, M., Lenhard, B., Lehvälaiho, H., and Brookes, A. J. HGVBbase: A curated resource describing human DNA variation and phenotype relationships. *Nucleic Acids Research* 32, Database Issue (2004), D516-519.
- 104 Yip, Y. L., Scheib, H., Diemand, A. V., Gattiker, A., Famiglietti, L. M., Gasteiger, E., and Bairoch, A. The Swiss-Prot variant page and the ModSNP database: a resource for sequence and structure information on human protein variants. *Human Mutation* 23, 5 (2004), 464-470.
- 105 Cargill, M., Altshuler, D., Ireland, J., Sklar, P., Ardlie, K., Patil, N., Lane, C. R., Lim, E. P., Kalyanaraman, N., Nemes, J., Ziaugra, L., Friedland, L., Rolfe, A., Warrington, J., Lipshutz, R., Daley, G.Q., and Lander, E. S. Characterization of single-nucleotide polymorphisms in coding regions of human genes. *Nature Genetics* 22, 3 (1999), 231-238.
- 106 Henikoff, S. and Henikoff, J.G. Amino acid substitution matrices from protein blocks. In *Proc. 89<sup>th</sup> National Academy of Sciences*, 1992.
- 107 Herrgard, S., Cammer, S. A., Hoffman, B. T., Knutson, S., Gallina, M., Speir, J.A., Fetrow, J.S., and Baxter, S.M. Prediction of deleterious functional effects of amino acid mutations using a library of structure-based function descriptors. *Proteins: Structure, Function, and Genetics* 53, 4 (2003), 806-816.
- 108 Chasman, D. and Adams, R.M. Predicting the functional consequences of non-synonymous single nucleotide polymorphisms: Structure-based assessment of amino acid variation. *J. Molecular Biology* 307, 2 (2001), 683-706.
- 109 Needham, C.J., Bradford, J.R., Bulpitt, A.J., Care, M.A., and Westhead, D.R. Predicting the effect of missense mutations on protein function: Analysis with Bayesian networks. *BMC Bioinformatics* 7 (2006), 405.
- 110 Dayhoff, M.O, Schwartz, R. and Orcutt, B. Survey of new data and computer methods of analysis. In *Atlas of Protein Sequence and Structure*, M.O. Dayhoff, Ed., National Biomedical Research Foundation, 1978, 345-352.
- 111 Bader, G.D., Donaldson, I., Wolting, C., Ouellette, B.F., Pawson, T., and Hogue C.W. BIND: The bimolecular interaction network database. *Nucleic Acids Research* 29, 1 (2001), 242-245.

- 112 Krishnan, V.G. and Westhead, D.R. A comparative study of machine-learning methods to predict the effects of single nucleotide polymorphisms on protein function. *Bioinformatics* 19, 17 (2003), 2199-2209.
- 113 Markiewicz, P., Kleina, L.G., Cruz, C., Ehret, S., and Miller, J.H. Genetic studies of the lac repressor. XIV. Analysis of 4000 altered Escherichia coli lac repressors reveals essential and non-essential residues, as well as "spacers" which do not require a specific sequence. *J. Molecular Biology* 240, 5 (1994), 421-433.
- 114 Suckow, J., Markiewicz, P., Kleina, L.G., Miller, J., Kisters-Woike, B., and Muller-Hill, B. Genetic studies of the Lac repressor. XV: 4000 single amino acid substitutions and analysis of the resulting phenotypes on the basis of the protein structure. *J. Molecular Biology* 261, 4 (1996), 509-523.
- 115 Breiman, L. Random forest. *Machine Learning* 45, 1 (2001), 5-32.
- 116 Bao, L. and Cui, Y. Prediction of the phenotypic effects of non-synonymous single nucleotide polymorphisms using structural and evolutionary information. *Bioinformatics* 21, 10 (2005), 2185-2190.
- 117 Bao, L., Zhou, M., and Cui, Y. nsSNPAnalyzer: identifying disease-associated non-synonymous single nucleotide polymorphisms. *Nucleic Acids Research* 33 Web Server Issue (2005), W480-W482.
- 118 Ferrer-Costa, C., Orozco, M., and de la Cruz, X. Use of bioinformatics tools for the annotation of disease-associated mutations in animal models. *Proteins: Structure, Function, and Bioinformatics* 61, 4 (2005), 878-887.
- 119 Bromberg, Y. and Rost, B. SNAP: Predicting effect of non-synonymous polymorphisms on function. *Nucleic Acids Research* 35, 11 (2007), 3823-3835.
- 120 Bromberg, Y., Yachdav, G., and Rost, B. SNAP predicts effect of mutations on protein functions. *Bioinformatics* 24, 20 (2008), 2397-2398.
- 121 Kawabata, T., Ota, M., and Nishikawa, K. The protein mutant database. *Nucleic Acids Research* 27, 1 (1999), 355-357.
- 122 Capriotti, E., Calabrese, R., and Casadio, R. Predicting the insurgence of human genetic diseases associated to single point protein mutations with support vector machines and evolutionary information. *Bioinformatics* 22, 22 (2006), 2729-2734.
- 123 Yue, P., Li, Z., and Moulton, J. Loss of protein structure stability as a major causative factor in monogenic disease. *J. Molecular Biology* 353, 2 (2005), 459-473.

- 124 Tian, J., Wu, N., Guo, X., Guo, J., Zhang, J., and Fan, Y. Predicting the phenotypic effects of non-synonymous single nucleotide polymorphisms based on support vector machines. *BMC Bioinformatics* 8 (2007), 450.
- 125 Ju, W., Shan, J., Yan, C., and Cheng, H. Discrimination of disease-related non-synonymous single nucleotide polymorphisms using multi-scale RBF kernel fuzzy support vector machine. *Pattern Recognition Letters* 30, 4 (2009), 391-396.
- 126 Thomas, P.D., Campbell, M.J., Kejariwal, A., Mi, H., Karlak, B., Daverman, R., Diemer, K., Muruganujan, A., and Narechania, A. PANTHER: A library of protein families and subfamilies indexed by function. *Genome Research* 13, 9(2003), 2129-2141.
- 127 Thomas, P. D., Kejariwal, A., Guo, N., Mi, H., Campbell, M.J., Muruganujan, A., and Lazareva-Ulitsky, B. Applications for protein sequence-function evolution data: mRNA/protein expression analysis and coding SNP scoring tools. *Nucleic Acid Research* 34, Web Server Issue (2006), W645-W650.
- 128 Krawczak, M. and Cooper, D.N. The human gene mutation database. *Trends in Genetics* 13, 3 (1997), 121-122.
- 129 Schneider T.D., Stormo, G.D., Gold, L., and Ehrenfeucht, A. Information content of binding sites on nucleotide sequences. *J. Molecular Biology* 188, 3 (1986), 415–431.
- 130 Grantham R. Amino acid difference formula to help explain protein evolution. *Science* 185, 4154 (1974), 862–864.
- 131 Rudd, M.F., Williams, R.D., Webb, E.L., Schmidt, S., Sellick, G.S., and Houlston, R.S. The predicted impact of coding single nucleotide polymorphisms database. *Cancer Epidemiology Biomarkers & Prevention* 14, 11 Pt 1 (2005), 2598-2604.
- 132 Fleisher, T. A. Back to basics: primary immune deficiencies: Windows into the immune system. *Pediatrics in Review* 27, 10 (2006), 363–372.
- 133 Robinson J, Waller, M. J., Parham, P., de Groot, N., Bontrop, R., Kennedy, L.J., Stoehr, P., and Marsh, S.G. IMGT/HLA and IMGT/MHC: Sequence databases for the study of the major histocompatibility complex. *Nucleic Acids Research* 31, 1 (2003), 311–314.
- 134 Quinlan, J.R. and Rivest, R.L. Inferring decision trees using the minimum description length principle. *Information and Computation* 80, 3 (1989), 227-248.
- 135 Quinlan, J.R. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993.

- 136 Feng, D.F., Johnson, M.S., Doolittle, R.F. Aligning amino acid sequences: Comparison of commonly used methods. *J. Molecular Evolution* 21, 2 (1985), 112-125.
- 137 Zhang, C. and Kim, S.H. Environment-dependent residue contact energies for proteins. In *Proc. National Academy of Sciences* 97, 6 (2000), 2550-2555.
- 138 Nakashima, H., Nishikawa, K., and Ooi, T. Distinct character in hydrophobicity of amino acid compositions of mitochondrial proteins. *Proteins: Structure, Function, and Bioinformatic*, 8, 2 (1990), 173-178.
- 139 Miyazawa, S. and Jernigan, R. L. Estimation of effective inter-residue contact energies from protein crystal structures: Quasi-chemical approximation. *Macromolecules* 18, 3 (1985), 534-552.
- 140 Mi, H., Guo, N., Kejariwal, A., and Thomas, P.D. PANTHER version 6: Protein sequence and function evolution data with expanded representation of biological pathways. *Nucleic Acids Research* 35, Database issue (2007), D247-252.
- 141 Luscombe, N.M., Austin, S.E., Berman, H.M., and Thornton, J.M. An overview of the structures of protein–DNA complexes. *Genome Biology* 1, 1 (2001), 1–37.
- 142 Jones, S. and Thornton, J.M. Protein–DNA interactions: The story so far and a new method for prediction. *Comparative and Functional Genomics* 4, 4 (2003), 428–431.
- 143 Berman, H.M., Olson, W.K., Beveridge, D.L., Westbrook, J., Gelbin, A., Demeny, T., Hsieh, S.H., Srinivasan, A.R., and Schneider, B. The nucleic acid database: A comprehensive relational database of three-dimensional structures of nucleic acids. *Biophysical J.* 63, 3 (1992), 751–759.
- 144 Harrison, S.C. A structural taxonomy of DNA-binding domains. *Nature* 353, 6346 (1991), 715–719.
- 145 Wintjens, R. and Rooman, M. Structural classification of HTH DNA-binding domains and protein-DNA interaction modes. *J. Molecular Biology* 262, 2 (1996), 294-313.
- 146 Pabo, C.O. and Sauer, R.T. Transcription factors: structural families and principle of DNA recognition. *Annual Review of Biochemistry* 61(1992), 1053–1095.
- 147 Aravind, L., Anantharaman, V., Balaji, S., Babu, M.M., and Iyer, L.M. The many faces of the helix-turn-helix domain: Transcription regulation and beyond. *FEMS Microbiology Reviews* 29, 2 (2005), 231-262.

- 148 Shanahan, H.P., Garcia, M.A., Jones, S., and Thornton, J.M. Identifying DNA-binding proteins using structural motifs and the electrostatic potential. *Nucleic Acids Research* 32, 16 (2004), 4732-4741.
- 149 Nevill-Manning, C.G., Wu, T.D., and Brutlag, D.L. Highly-specific protein sequence motifs for genome analysis. In *Proc. National Academy of Science*, 1998.
- 150 Gribskov, M., Luthy, R., and Eisenberg, D. Profile analysis. In *Molecular Evolution: Computer Analysis of Protein and Nucleic Acid Sequences, Methods in Enzymology*, J. N. Abelson, M. I. Simon, and R. F. Doolittle, Eds. Academic Press, 1990, 146-159.
- 151 Narasimhan, G., Bu, C., Gao, Y., Wang, X., Xu, N., and Mathee, K. Mining protein sequences for motifs. *J. of Computational Biology* 9, 5 (2002), 707-720.
- 152 Mathee, K. and Narasimhan, G. Detection of DNA-binding helix-turn-helix motifs in proteins using the pattern dictionary method. In *Methods in Enzymology*, S. Adhya and S. Garges, Eds. Academic Press, 2003, 250-264.
- 153 Suzuki, M. and Brenner, S.E. Classification of multi-helical DNA-binding domains and application to predict the DBD structures of sigma factor, LysR, OmpR/PhoB, CENP-B, RapI, and Xy1S/Ada/AraC. *FEBS Letters* 372, 2-3 (1995), 215-221.
- 154 McLaughlin, W.A. and Berman, H.M. Statistical models for discerning protein structures containing the DNA-binding Helix-Turn-Helix motif. *J. Molecular Biology* 330, 1 (2003), 43-55.
- 155 Hubbard, S.J. and Thornton, J.M. 'NACCESS'. White paper. Computer Program, Department of Biochemistry and Molecular Biology, University College, London, 1993.
- 156 Hutchinson, E. G. and Thornton, J. M. PROMOTIF— A program to identify and analyze structural motifs in proteins. *Protein Science* 5, 2 (1996), 212-220.
- 157 Jones, S., Barker, J.A., Nobeli, I., and Thornton, J.M. Using structural motif templates to identify proteins with DNA binding function. *Nucleic Acids Research* 31, 11 (2003), 2811-282.
- 158 Kabsch, W. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica A* 32 (1976), 922-923.
- 159 Ferrer-Costa, C., Shanahan, H.P., Jones, S., and Thornton, J.M. HTHquery: A method for detecting DNA-binding proteins with a helix-turn-helix structural motif. *Nucleic Acids Research* 21, 18 (2005), 3679-3680.

- 160 Pellegrini-Calace, M. and Thornton, J. M. Detecting DNA-binding helix-turn-helix structural motifs using sequence and structure information. *Nucleic Acids Research* 33, 7 (2005), 2129-2140.
- 161 Finn, R.D., Mistry, J., Schuster-Böckler, B., Griffiths-Jones, S., Hollich, V., Lassmann, T., Moxon, S., Marshall, M., Khanna, A., Durbin, R., Eddy, S. R., Sonnhammer, E.L., and Bateman, A. Pfam: Clans, web tools and services. *Nucleic Acids Research* 34, Database Issue (2006), D247-251.
- 162 Rabiner, L.R. and Juang, B.H. An introduction to hidden Markov models. *IEEE ASSP Magazine* 3, 1 (1986), 4-15.
- 163 Viterbi, A.J. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm, *IEEE Trans. Information Theory* 13, 2 (1967), 260-269.
- 164 Baum, L.E., Petrie, T., Soules, G., and Weiss, N. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics* 41, 1 (1970), 164-171.
- 165 Choo, K. H., Tong, J. C. and Zhang, L. Recent applications of hidden Markov models in computational biology. *Genomics Proteomics Bioinformatics*, 2, 2 (2004), 84-96.
- 166 Krogh, A., Brown, M., Mian, I.S., Sjolander, K., and Haussler, D. Hidden Markov models in computational biology: Applications to protein modeling. *J. Molecular Biology* 235, 5 (1994), 1501-1531.
- 167 Eddy, S. R. Profile hidden Markov model. *Bioinformatics* 14, 9 (1998), 755-763.
- 168 Durbin, R., Eddy, S., Krogh, A., and Mitchison, G. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, 1998.
- 169 Delorenzi, M. and Speed, T. An HMM model for coiled-coil domains and a comparison with PSSM-based predictions. *Bioinformatics* 18, 4 (2002), 617-625.
- 170 Camproux, A.C. and Tufféry, P. Hidden Markov model-derived structural alphabet for proteins: The learning of protein local shapes capture sequence specificity. *Biochimica et Biophysica Acta* 1724, 3 (2005), 394-403.
- 171 Alexandrov, V. and Gerstein, M. Using 3D hidden Markov models that explicitly represent spatial coordinates to model and compare protein structures. *BMC Bioinformatics* 5 (2004), 2.

- 172 Hargbo, J. and Elofsson, A. Hidden Markov models that use predicted secondary structures for fold recognition. *Proteins: Structures, Functions and Bioinformatics* 36, 1 (2005), 68-76.
- 173 Barrett, C., Hughey, R., and Karplus, K. Scoring hidden Markov models. *CABIOS* 13, 2 (1997), 191-199.
- 174 Murphy, L.R., Wallqvist, A., and Levy, R.M. Simplified amino acid alphabets for protein fold recognition and implications for folding. *Protein Engineering* 13, 3 (2000), 149-152.
- 175 Li, W., Jaroszewski, L., and Godzik, A. Clustering of highly homologous sequences to reduce the size of large protein databases. *Bioinformatics* 17, 3 (2001), 282-283.
- 176 Adamczak, R., Porollo, A., and Meller, J. Combining prediction of secondary structure and solvent accessibility in proteins. *Proteins: Structures, Functions and Bioinformatics* 59, 3 (2005), 467-475.

## CURRICULUM VITAE

**Jing Hu**

Department of Computer Science  
Utah State University, Logan, UT 84322-4205  
Cell phone: (435)232-5883  
E-mail: [jing.hu@aggiemail.usu.edu](mailto:jing.hu@aggiemail.usu.edu)  
Webpage: <http://cc.usu.edu/~jinghu>

### EDUCATION

- |  |                 |
|--|-----------------|
| <p><b>PhD</b>, Computer Science<br/>Utah State University, Logan, Utah, USA<br/>Advisor: Dr. Changhui Yan<br/>Dissertation: "Prediction of Protein Function and Functional Sites From Protein Sequences"</p> | 08/2002-05/2009 |
| <p><b>BS</b>, Computer Science<br/>Xidian University, Xi'an, China</p>   | 08/1995-07/1999 |

### RESEARCH INTERESTS

Bioinformatics	Computational Biology
Machine Learning	Data Mining

### RESEARCH AND WORK EXPERIENCE

- |  |                 |
|--|-----------------|
| <p><b>Research Assistant</b><br/>Yan's Bioinformatics Group, Department of Computer Science, Utah State University, Logan, UT</p> <ul style="list-style-type: none"> <li>• Developed and applied data mining and machine learning methods to analyze and solve certain bioinformatics problems, such as prediction of protein functions and structures, identification of protein functional sites, deleterious non-synonymous single nucleotide polymorphisms (nsSNPs), etc.</li> <li>• Constructed and implemented software and web servers to provide bioinformatics services to academic community.</li> </ul> | 01/2006-05/2009 |
| <p><b>Research Assistant</b><br/>Evolvable Hardware Research Group, Department of Computer Science, Utah State University, Logan, UT</p> <ul style="list-style-type: none"> <li>• Evolved neural network modules on Celoxica FPGA boards using Handel-C language.</li> <li>• Applied evolutionary algorithm to evolve fully connected neural networks with specific functions.</li> </ul>  | 09/2005-05/2006 |
| <p><b>Telecommunication Engineer</b><br/>Alcatel Lucent (Former name: Alcatel Shanghai Bell), Shanghai, China</p> <ul style="list-style-type: none"> <li>• Configured and designed hardware of voice and data switch systems for over 20 branch companies of China Mobil Limited using Alcatel 1240 MSC exchange system.</li> <li>• Developed office application programs for the processing of design data.</li> </ul>  | 07/1999-07/2002 |

## TEACHING EXPERIENCE

**Instructor**, C++ Programming I (CS1400) Summer, 2008

Department of Computer Science, Utah State University, Logan, UT

- Delivered lectures, assigned homework, prepared exams, and administered all grades (19 students).
- Students' evaluations of the class:  
 Overall quality: **5.2/6.0** (Dept avg: 5.1; College: 5.0; University: 5.1)  
 Instructor's effectiveness: **5.3/6.0** (Dept avg: 5.2; College: 5.0; University: 5.2)

**Instructor**, C++ Programming I, lab class, 3 sections (CS1405) Summer, 2008

Department of Computer Science, Utah State University, Logan, UT

- Coordinated and supervised 3 sections of the lab classes of C++ programming (13 students).

**Instructor**, Topics in Bioinformatics (CS7960) Spring, 2008

Department of Computer Science, Utah State University, Logan, UT

- Co-instructed the class with Dr Changhui Yan.
- Delivered topics including Hidden Markov Model and its application in bioinformatics (6 PhD students).
- Assigned homework and projects.

## Teaching Assistant

Department of Computer Science, Utah State University, Logan, UT

- |  |                         |
|--|-------------------------|
| Quantum Computing (CS7910)                                 | Spring, 2006            |
| Introduction to Database Systems (CS5800)                  | Fall, 2005              |
| C++ Programming II (CS1410)                                | Fall, 2004-Spring, 2005 |
| Graphical User Interfaces and Windows Programming (CS5100) | Spring, 2004            |
| Advanced Software Engineering w/o project (CS5370/6370)    | Fall, 2003              |
| Compiler Design (CS5300)                                   | Fall, 2003              |
| C++ Programming I (CS1400)                                 | Fall, 2002-Spring, 2003 |
- Collaborated on class development, met with students upon request, and graded all assignments and programming projects.

## AWARDS AND HONORS

- Eleventh Annual Intermountain Paper and Poster Symposium, Utah State University, **Third place** in College of Science. 2008
- **Graduate Students Senates Enhancement Award (\$4,000)** from Graduate Students Senate, Utah State University. 2008
- Named to **the School of Graduate Studies Honor Roll**, School of Graduate Studies, Utah State University. 2004
- **Commending letter** from the president and the Utah State University Board of Trustees, Administration and Faculty for outstanding scholastic achievement. 2004

- **Xidian University Student Fellowship, Second Prize**, Xidian University, Xi'an, China, 1997,1998
- **Xidian University Student Fellowship, First Prize**, Xidian University, Xi'an, China, 1995

## SELECTED PUBLICATIONS

### Journals

- 2009 • **Hu, J.** and Yan, C. A Tool for Calculating Interface Residues on Proteins (TCIRP). Submitted to BMC Structural Biology (under review).
- 2008 • Yan, C., **Hu, J.** and Wang, Y. A Graph Kernel Method for DNA-Binding Site Prediction. Submitted to Bioinformatics (under review).
- **Hu, J.** and Yan, C. Identification of Deleterious Non-synonymous Single Nucleotide Polymorphisms Using Only Sequence-derived Information, BMC Bioinformatics, 2008, 9:297.
- **Hu, J.** and Yan, C. A Method for Discovering Transmembrane Beta-barrel Proteins in Gram-negative Bacterial Proteomes. Computational Biology and Chemistry, 2008, 32:298-301.
- Yan, C., **Hu, J.** and Wang, Y. Discrimination of Outer Membrane Proteins Using a K-nearest Neighbor Method. Amino Acids, 2008, 35(1):65-73.
- **Hu, J.** and Yan, C. Protein Subcellular Localisation Prediction with Improved Performance. International Journal of Functional Informatics and Personalised Medicine, 2008, 1: 321-328.
- **Hu, J.**, and Yan, C. An Improved Method for Alpha-helical Transmembrane Protein Topology Prediction. Bioinformatics and Biology Insights, 2008, 2: 67-74.
- Yan, C., **Hu, J.** and Wang, Y. Discrimination of Outer Membrane Proteins with Improved Performance. BMC Bioinformatics, 2008, 9:47.
- 2006 • Yan, C. and **Hu, J.** An Exploration to the Combining of Solvent Accessibility With Amino Acid Sequence in the Identification of Helix-Turn-Helix motifs, WSEAS Transaction on Biology and Biomedicine, 2006, 6(3): 477-484.

### Conferences

- 2008 • **Hu, J.** and Yan, C. Mining Sequence Features for DNA-binding Site Prediction. In Proceedings of IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, 2008, 219-222.

- 2007 • **Hu, J.** and Yan, C. Predicting Protein Subcellular Localizations Using Weighted Euclidian Distance. In Proceedings of IEEE 7<sup>th</sup> International Symposium on BioInformatics and BioEngineering, 2007, 1370-1373.
- 2006 • Yan, C. and **Hu, J.** Identification of Helix-Turn-Helix Motifs From Amino Acid Sequence. In Proceedings of IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, 2006, 1-7.
- Yan, C. and **Hu, J.** A Hidden Markov Model for the Identification of Helix-Turn-Helix Motifs. In Proceedings of WSEAS International Conference on Cellular and Molecular Biology-Biophysics and Bioengineering, 2006, 14-19.
- de Garis, H., Liu, R., Huang, D. and **Hu, J.** Artificial Brains. An Inexpensive Method for Accelerating the Evolution of Neural Network Modules for Building Artificial Brains, In Proceedings of the AGI Workshop, 2006, 144-158.
- 2005 • Flann N. S., **Hu J.**, Bansal M., Patel V. and Podgorski G. Biological Development of Cell Patterns: Characterizing the Space of Cell Chemistry Genetic Regulatory Networks. In proceedings of Eighth European Conference on Artificial Life, 2005, 57-66.

### Books

- Hugo de Garis (Author) and **Jing Hu** (Translator). The Artilect War: Cosmists vs. Terrans: A bitter Controversy Concerning Whether Humanity Should Build Godlike Massively Intelligent Machines. Published by Tsinghua University Press (China), ISBN 978-7-302-15015-2, 2007.