Protein Structure and Fold Prediction Using Tree-Augmented Bayesian Classifier

A. Chinnasamy, W.-K. Sung, and A. Mittal

Pacific Symposium on Biocomputing 9:387-398(2004)

PROTEIN STRUCTURE AND FOLD PREDICTION USING TREE-AUGMENTED NAÏVE BAYESIAN CLASSIFIER

A. CHINNASAMY, W. K. SUNG
(arun, ksung)@comp.nus.edu.sg,
Department of Computer Science,
National University of Singapore,
3 Science Drive 2, Singapore 117543.

A. MITTAL
ankush@bits-pilani.ac.in,
Department of Computer Science,
Birla Institute of Technology and Science,
Pilani, India.

For determining the structure class and fold class of Protein Structure, computer-based techniques have became essential considering the large volume of the data. Several techniques based on sequence similarity, Neural Networks, SVMs, etc have been applied. This paper presents a framework using the Tree-Augmented Networks (TAN) based on the theory of learning Bayesian networks but with less restrictive assumptions than the naïve Bayesian networks. In order to enhance TAN's performance, pre-processing of data is done by feature discretization and post-processing is done by using Mean Probability Voting (MPV) scheme.
The advantage of using Bayesian approach over other learning methods is that the network structure is intuitive. In addition, one can read off the TAN structure probabilities to determine the significance of each feature (say, Hydrophobicity) for each class, which help to further understand the mystery of protein structure. Experimental results and comparison with other works over two databases show the effectiveness of our TAN based framework. The idea is implemented as the BAYESPROT web server and it is available at http://www-appn.comp.nus.edu.sg/∼bioinfo/bayesprot/Default.htm.

1 Introduction

In proteomics, finding the structure and the fold of a protein is very important since it helps to understand the functions, the catalytic and the structural roles of proteins. Protein structure can be determined experimentally by X-ray diffraction and NMR techniques. These methods are expensive, tedious, labor intensive and have their own limitations. This leads to the research in predicting the protein folding pattern, given only its primary structure [6]. This computational way of protein structure prediction can be classified into two general types [9].

1. **Homology methods:**
   a) Sequence Similarity Methods: These methods are based on the observation that two proteins have very similar structure if their sequences have high homology [3].
   b) Threading Methods: These methods predict the structure of a protein sequence by aligning with a known structure. [12].

2. **Discriminative Methods:** These methods extract some general "rules" from the known protein structures and applies the "rules" to a new protein sequence to make the prediction [16].

Sequence similarity has its limitation as it can apply only to those sequences which are similar in term of both sequences and structures [3]. Several discriminative methods based on statistical techniques, neural networks and SVMs have been applied in the past. The main difficulty in applying learning(discriminative) methods is, the folding prediction becomes less accurate with increasing number of classes. This study hopes to solve this issues using the Bayesian classifier framework.

Bayesian classifier theoretically is the best classifier provided the underlying distribution functions are well estimated [7]. However, Bayesian classifier requires a prior knowledge of many probabilities. This paper designs a framework called BAYESPROT with discretization of feature space and Tree-Augmented Network (TAN) Bayesian classifier as foundation to address the problem of structure and fold classification from database. In addition, Mean Probability Voting (MPV) method is employed to improve the performance.

For the prediction in this paper,we use the protein classification type in SCOP [22] database, that is, proteins are classified in hierarchical order of structures, folds, super families and families. Since finding the structural and the fold class is more significant, in this paper we applied our classification system to classify a protein into different structural and fold classes.

2 Review

Recently, machine learning tools have been largely used in the classification based on tertiary super classes. These methods are denoted as discriminative methods or data mining approaches. Since no direct relationship between sequence and structure are derived, much attention paid on statistical or machine learning techniques to classify the proteins using feature vector representations of available knowledge. Dubchak et al 1995, 1999 [5,6] conducted the classification studies based on neural networks. Ding and Dubchak I (2001) [4] classified the proteins into 27 fold classes using SVMs and neural networks based on three

multi-classification methods (OvO, uOvO, AvA) and concluded that SVM's performance is better than Neural networks. Their study introduces SVM to the protein classification problem. The accuracy measurement in their method assumes that the prediction is partially correct when ties exist(for ours, we assume the prediction fail). Also their method uses large number of classifiers. Cai et al.(2001) [19] used SVMs to classify the proteins into four major protein classes and compared the results with component coupled with neural network. Edler et al. (2001) [8] conducted a statistical study based on logistic regression, additive models, and projection pursuit on protein fold prediction with a dataset containing 268 proteins. Markowetz et al.(2003) [9] used Gaussian and various polynomial kernels based on SVMs and showed that their approach performed better than the work in [8]. From all these studies it is evident that among all the prediction methods, SVM performs better.

Though most works recently showed that SVMs have good generalization property and outperforms statistically than Neural network methods for the protein fold prediction, SVM methods are reported to result in high number of 'false positives'[4]. Besides, the number of binary classifiers is numerous and the computational time for the SVM training is high when the number of classes is large. It has also been shown that SVMs performances vary with change in dimensions of the feature vector and SVM methods might require feature selection [1]. Therefore, alternative method of learning are sought which might not have some of these defaults.

3 Overview of BAYESPROT

Figure 1 shows the overview of the BAYESPROT system. Given a database of several millions of protein sequences, their attributes are extracted and transformed into features, namely, composition (20), secondary structure (21), hydrophobicity (21), polarity (21), polarizability (21), and Van Der Waals Volume (21).

After the feature vector extraction, the values of features were discretized to four discrete states by frequency discretization method. Three separate TAN Bayesian classifiers were constructed using all concatenated feature vectors (126), composition feature vectors (20), and secondary structure feature vectors (21) respectively. The previous research and our experiments suggest that, amongst all the attributes, composition and secondary structure features are the most important for the protein structure prediction. Hence, we construct the TAN classifiers for composition and secondary structure separately and chose only these two to reduce the complexity. Next MPV is employed to predict the structural class. A similar procedure is required to classify the fold

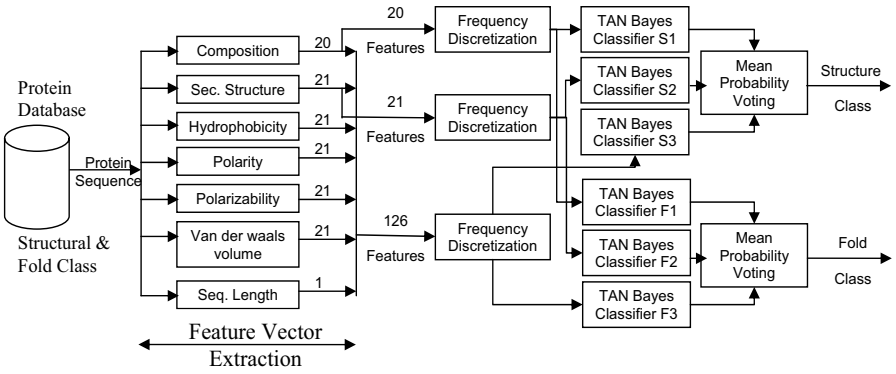

Figure 1: Architecture of BAYESPROT

class as shown in the Figure 1.

4 Dataset and feature vector representation

We used the datasets referred in two prominent recent works: Ding and Dubchak (2001) [4] and Markowetz et al.(2003) [9]. Summary of the two datasets (Dataset I and Dataset II) is tabulated in Table 2.

4.1 Dataset I

Dataset I used in our study was originally built for the study of [5] and later used by [4]. Both studies confirm that the dataset is reasonable as it is based on the PDB_select sets where two proteins have no more than 35% of the sequence identity for sequences longer than 80 residues. Dataset I is available at http://www.nersc.gov/∼cding/protein/.

4.2 Dataset II

Dataset II was built from the Database for expected Fold-Classes (DEF) for the statistical study [20]. Markowetzet et al.(2003) [9] used this dataset and concluded that SVM was better than previous statistical studies. Dataset II is available at http://www.dkfz.de/biostatistics/protein/gsme97.html.

4.3 Feature Vectors or Global Descriptors of Amino Acid Sequence

To apply machine learning algorithm, we have to turn the amino acid sequence of heterogeneous length into feature vector of homogeneous length. This feature vector construction is based on physical and stereo chemical properties of amino acids. This method was used and explained in [5] and [6]. Each protein sequence is represented by a set of six attribute feature vectors. Composition feature vector of length 20, which lists out the proportion of the 20 amino acids, is constructed in a straightforward manner. Apart from composition, the other attributes used are predicted secondary structure, polarity, polarizability, hydrophobicity and Van der Waals volume.

Except composition, feature vectors for the above five attributes are constructed in two steps.

Step1: For each attribute, twenty amino acids are divided into three groups,(see Table 1). For each protein sequence, every amino acid was replaced by the index 1, 2, or 3 depending on its grouping. For example protein sequence KLLSHCLLVTLAAHLPAEFTPAV will be replaced by 13322333323222322132232 based on the attribute hydrophobicity division of amino acids(see Table 1).

Step 2: For each converted sequences calculated in step1 three descriptors "composition" (C), "transition" (T), and "distribution"(D), are calculated based on the definition given below.

Composition: Composition is calculated for each group based on the simple formula, $C_i = ((n_i)/L) * 100$; where C_i represents the percent composition of each $group_i$, where n_i represents total number of $group_i$ residues in the sequences, and L represents the length of the sequence.

Transition: Transition (T_{ij}) is represented by the percent frequency with which $group_i$ is followed by $group_j$ or $group_j$ followed by $group_i$ where i, j takes the values 1, 2 and 3.

Distribution: Distribution descriptor D consists of the five numbers for each of the three groups: the fractions of the entire sequence, where the first residue of a given group is located, and where 25%, 50%, 75%, and 100% of those are contained.

Each attribute the feature vector contains 21 features: 3 composition features, 3 transition features and 5* 3 distribution features. Feature vector is of length 126 which is constructed by concatenating [21] all 5 attribute vectors of

Table 1: Amino acid attributes and corresponding groups.

Attribute	Group 1	Group 2	Group 3
secondary structure	Helix	Strand	Coil
Hydrophobicity	Polar R,K,E,D,Q,N	Neutral G,A,S,T,P,H,Y	Hydrophobic C,V,L,I,M,F,W
Polarizability	(0-2.78) G,A,S,C,T,P,D	(2.95-4.0) N,V,E,Q,I,L	(4.43-8.08) M,H,K,F,R,Y,W
Polarity	(4.9-6.2) L,I,F,W,C,M,V,Y	(8.0-9.2) P,A,T,G,S	(10.4-13.0) H,Q,R,K,N,E,D
Van der Waals volume	(0-0.108) G,A,S,D,T	(0.128-0.186) C,P,N,V,E,Q,I,L	(0.219-0.409) K,M,H,F,R,Y,W

length 105 (5*21=105), amino acid composition vector of length 20 and the sequence length of length 1.

5 Our Framework

5.1 Discretization

In our dataset, the feature vectors are of continuous nature. Though the Bayesian classifier supports both continuous and discrete probability distributions [11], it was experimentally found that the continuous probability distribution was not suitable for these datasets. Therefore, we pre-processed data by converting the continuous attribute data to discrete attribute data. One popular and simple discretization approach is range discretization. However, in range discretization, some of the discretized partitions become over-populated while others remain empty leaving to poor discretization. In order to avoid this problem, we employ frequency-based discretization which partitions the attributes into intervals each containing almost same number of instances. Several frequency based discretization methods were employed with '3' intervals, '4' intervals, '5' intervals, '7' intervals and '10' intervals. By experiment, method with '4' intervals yielded better classification performance than other methods and it was chosen.

5.2 TAN Bayesian Classifier

Bayesian Networks are directed acyclic graphs which combine both statistical and graph theory for representing conditional independencies [10]. A directed edge $A \longrightarrow B$ indicates the causal relationship (A causes B) and thus Bayesian

networks are quite intuitive. Optimal classifications can be achieved by reasoning about these probabilities along with observed data [14]. The classification is done by applying Bayes rules to compute the probability of a class C given the particular instance of attributes $A_1, \ldots, A_n$ and then predicting the class with the highest probability.

Structural relationship among the attributes is important for the Bayesian network classifier to construct the relationship amongst various nodes. However, no clear structural relationship is known at present due to the nature of problem. Structural learning is not possible with present database. Therefore, we chose TAN Bayesian classifier [13,15] rather than Bayesian network classifier as it is more relevant to the problem considering the feature vector properties and relations.

TAN Bayesian Classifier is an extension of naïve Bayesian classifier. Similar to naïve Bayesian classifier, TAN consists of a class node connecting to all child nodes each representing a feature. Moreover, each child node can has at most one other feature node as parent. Attractive property of the TAN Bayesian classifier is that it learns the probabilities from the data in polynomial time. For our case, we create a TAN Bayesian classifier which has a class node representing the protein structure/fold classes and connected to 126 child nodes for 126 feature vectors. In addition, it is assumed that composition node C_i has structural relationship with C_{i+1} , each attribute percent composition and each distribution vector has structural relationship. Three TAN Bayes classifiers have been constructed for the concatenated feature vectors of length 126, composition feature vectors of length 20 and secondary structure feature vectors of length 21 respectively. TAN Bayes classifier has been defined in the given equation where α is normalization constant.

$$P(Class|A_1, \ldots, A_n) = \alpha \cdot P(Class) \cdot \Pi_{i=1}^{n} P(A_i|parents(A_i)) \qquad (1)$$

5.3 Mean Probability Voting

Let P_i , PC_i and PS_i for $i = 1, 2, \ldots, k$ be the marginal probabilities from the TAN Bayesian classifiers which uses length 126 concatenated feature vectors, length 20 composition feature vectors and length 21 secondary structure feature vectors , respectively where 'k' represents the number of classes. Then mean probability MP_i, for $i = 1, 2, \ldots, k$ is calculated by taking average of P_i , PC_i and PS_i. The prediction of structural/fold class was done by selecting the class which has the highest mean probability (MP). It is accepted from the previous studies that composition [9] and secondary structure [6] are important

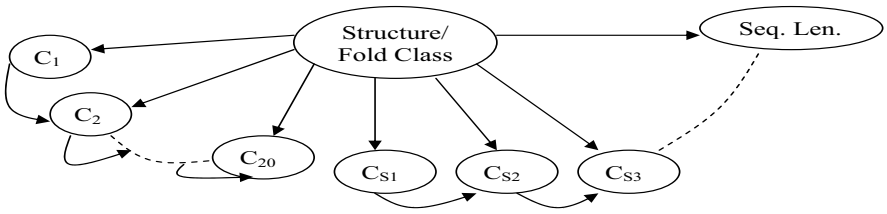

Figure 2: TAN Bayesian Classifier

Table 2: Structural and Fold Classification Results of BAYESPROT.

Dataset	Number of Classes	No. of Proteins in Train Data	No. of Proteins in Test data	Test Data Accuracy(%)	Cross Validation Accuracy (%)
	Structural Classes				
Dataset I	5	313	385	80.52	83.09
Dataset II	4	143	125	77.6	79.85
	Fold Classes				
Dataset I	27	313	385	58.18	59.77
Dataset II	42	143	125	74.40	75.75

in deciding the protein structures. And in our experiment, voting increased the accuracy by around 4%.

6 Experimental Results

6.1 Results

Both structural and fold classifications have been done using BAYESPROT with dataset I and dataset II. Table 2 summarizes the results of both dataset. Evaluation of classifier is done by testing with independent test dataset and 10-fold cross validation. In Dataset I, 27 fold classes used are from the structural classes α, β, α/β, $\alpha+\beta$, small and in Dataset II, 42 fold classes used are from the structural classes α, β, α/β,and $\alpha+\beta$.

For dataset I structural classes, the confusion matrix is shown in Table 3 while the sensitivity and the specificity for five structural classes are shown in Table 4. Except $\alpha+\beta$ super class, all other super classes are predicted with sensitivity greater than 70%.

From confusion matrix for structural classifier it is evident that a significant number of proteins of '$\alpha+\beta$' class are misclassified in 'α' and 'β' classes. Similarly, some 'β' class proteins are misclassified in 'α/β'. Specificity of each

Table 3: Confusion Matrix for Super Classifier (Dataset I)

Predicted Actual	α	β	α/β	$\alpha+\beta$	Small
α	49	6	2	4	0
β	6	91	16	4	0
α/β	4	5	132	4	0
$\alpha+\beta$	8	8	4	14	1
Small	0	5	0	1	24

Table 4: Sensitivity and Specificity for each class (Dataset I)

	Class	Sensitivity (%)	Specificity (%)
Structural	α	80.33	94.44
	β	77.78	92.16
Classes	α/β	91.03	90.83
	$\alpha+\beta$	40.00	96.29
	Small	88.89	99.72
Fold Classes	Average of 27 classes	50.89	61.76

structural class is very high compared to sensitivity. Confusion matrix and individual accuracy tables for Database II structural classes are available in the webpage http://www.comp.nus.edu.sg/∼bioinfo/bayesprot/results.htm.

From the experiment, it can be concluded that BAYESPROT classified six fold classes with accuracy greater than 60% and predicted 15 fold classes with accuracy greater than 50% in dataset I. Average specificity of 27 fold classes is 61.76% which is higher than average sensitivity 50.89%. Confusion matrix and detailed results for 27 fold classes and 42 fold classes are available in the webpage http://www.comp.nus.edu.sg/∼bioinfo/bayesprot/results.htm.

7 Analysis and Discussions

7.1 Dataset I: Comparison with Ding and Dubchak(2001)

In Ding and Dubchak [4] study, they used One-Versus-Others(OvO), Unique-One-Versus-Others(uOvO) or All-Versus-All(AvA) methods for multi classification which used binary SVM or Neural networks as building blocks.

Table 5 summarizes the result of 27 fold classes by BAYESPROT and SVM [4]. In 10-fold cross-validation study accuracy of 59.77% is achieved by BAYESPROT which is 31.57% higher than SVM AvA method. The number

Table 5: Comparative Results of BAYESPROT and SVM with Dataset I

	Test Dataset				Cross Validation	
Methods	BAYES PROT	SVM OvO	SVM uOvO	SVM AvA	BAYES PROT	SVM AvA
Accuracy (%)	58.8	41.8	45.2	56.0	59.77	45.4
No. of Clfrs. Used	3 TAN Bayes Clfrs.	168 Binary SVM Clfrs.	2457 Binary SVM Clfrs.	2106 Binary SVM Clfrs.	30 TAN BAYES Clfrs.	84,240 Binary SVM Clfrs.

of classifiers used for this cross-validation study is 10*3 (=30) TAN Bayesian Classifier, which is substantially less than the number of classifiers in SVM AvA where 84,240 binary SVM classifiers were employed.

It is important to note that the accuracy measurement used in our study and [4] are different by the way of calculating the number of proteins correctly classified by the classifier. In the method by [4], if the output for the three top classes C_1, C_2 and C_3 are 2, 2 and 1 respectively by voting results and the correct class is C_2, then the number of correctly predicted protein is counted as 0.5 in their work. However, our work considers such a case to be a misclassification and we do not increment the number of true positives. Thus, the superiority of BAYESPROT method over SVM can be observed.

Another thing to be considered is that the number of classifiers used in SVM and Neural networks is much higher than BAYESPROT. Learning complexity of SVM depends on the number of iterations and in many cases the learning complexity is quite high. But in BAYESPROT, since the dataset is complete and structure is known, the time required to learn the parameters is very less. In addition, the number of classifiers used in Bayesian network is substantially less than SVM as can be seen in Table 5.

7.2 Dataset II: Comparison with Markowetz et al.(2003)

Dataset II consists of 42 fold classes, 143 training proteins and 125 test proteins. In [9] study OvO SVM multi classification method was employed and achieved a high accuracy of 76.8% among various kernel for the test dataset and 70.9% for cross-validation.

Table 6 summarizes the BAYESPROT and SVM results. Distribution of number of proteins in all classes is quite less in dataset II which is not the case with dataset I. Out of 42 fold classes, 36 classes have proteins less than or equal to 4 in training dataset.

Table 6: Comparative Results of BAYESPROT and SVM with Dataset II

	Test Dataset				Cross Validation			
Methods	BAYES PROT	SVM RBF kernel	SVM Poly1 kernel	SVM Poly2 kernel	BAYES PROT	SVM AvA kernel	SVM Poly1 kernel	SVM Poly2 kernel
Accuracy (%)	74.40	76.8	71.2	68	75.75	69.8	70.9	65
No. of Clfrs. Used	3 TAN Bayes Clfrs.	42 Binary SVM Clfrs.	42 Binary SVM Clfrs.	42 Binary SVM Clfrs.	30 TAN BAYES Clfrs.	420 Binary SVM Clfrs.	420 Binary SVM Clfrs.	420 Binary SVM Clfrs.

7.3 Effects of Large number of Training Samples

Cross validation is a method to estimate the generalization error of a given model. We conducted 10 fold cross validation study to estimate the generalization error and to compare with previous SVM methods. From Table 5 and Table 6, it is clear evident that after performing cross validation over dataset I and dataset II, accuracy in BAYESPROT increases while the accuracy in SVM method decreases.

7.4 Interpreting the Classification Results

Analyzing the classification results is very important for solving biological problems. The biologists need to know the confident level of the resultant classes outputted by the classifiers for further analysis. Understanding the marginal differences between top predicted classes is also important in further confirming the structural class of the protein. Our classification approach supports this type of interpretations, as it gives the probability for each class.

This kind of interpretation is not possible in neural networks and difficult in SVM. Neural networks contain many hidden nodes and final output is based on threshold value. In SVM, as the number of classifiers is high, reading the distances between hyper plane and the classes are very difficult.

8 Conclusions and future work

In this paper, we presented a framework based on TAN and voting method that is shown to perform better than SVM on most cases. Since the network structure and the probabilities are well understood, the BAYESPROT framework also has several theoretical advantages relevant to biology researchers and thus it is a better tool for analyzing protein sequences. Further research is being carried out for incorporating better network structure than TAN to improve the performance.

References

1. A. Mittal et al., SPIE Conf. on Applns. of Art. Neural Networks in Image Procsg. VI, USA, 97-107, 2001.
2. A. Mittal and L.-F. Cheong, IEEE Transactions on Knowledge & Data Engg., vol 15, no4,(2003).
3. David W. Mount, Cold Spring Harbor Laboratory Press, (2001).
4. Ding CHQ, Dubchak I, Bioinformatics, 4(17):349-358, (2001).
5. Dubchak I, Muchnik I, Mayor C, Dralyuk I, Kim SH , Proteins Jun 1; 35(4):401-7, (1999).
6. Dubchak I, I. Muchnik, S.R. Holbrook and S.-H. Kim, Proced. of Natl. Acad. of Sci. of USA, 92, 8700-8704, (1995).
7. Duda, R.O. and Hart, P.E,and and D. G. Stork, and D. G. Stork , John Wiley & Sons, (2001).
8. Edler L et al., Math. and Computer Modelling 33, 1401-1417, (2001).
9. F. Markowetz, L. Edler, and M. Vingron, Biometrical Journal 45 3, 377-389, (2003).
10. Finn V. Jensen, Springer-Verlag, New york, (2001).
11. John, G.H., & Langley, P. , In Proced. of the 11th Conf. on Uncert. in AI, Montreal, Quebec, Morgan Kaufmann, pp. 338-345, (1995).
12. Jones D, Taylor W, Thornton J Nature,358:86-89, (1992).
13. Nir Friedman et al., Machine Learning 29(2-3): 131-163 (1997).
14. P. Domingos and M. Pazzani, Machine Learning, 29:103–130, 1997.
15. Pat Langley et al., In Procd. of the 10 Natnl. Conference on AI, pages 223-228. AAAI Press and MIT Press, (1992).
16. P. Wang and D. Zhang, the 14th IEEE Int. Conf. on Tools with AI. November pp. 252-257, (2002).
17. Ronan Collobert and Samy Bengio, J. of Machine Learning Research, vol 1, pages 143-160, 2001.
18. Sippl MJ, Flockner H, Structure 4, 15-19, (1996).
19. Yu-Dong Cai, Xiao-Jun Liu, Xue-biao Xu and Guo-Ping Zhou,BMC Bioinformatics 2:3, (2001).
20. J. Grassmann, M. Reczko, S. Suhai and L. Edler, In Proc. Int. Conf. Intell. Syst. Mol. Biol (ISMB 1999), pp. 106-12, (1999).
21. Joel R. Bock, David A. Gough, Bioinformatics vol 17-5, 455-460, (2001).
22. Murzin A. G., Brenner S. E., Hubbard T., Chothia C., J. Mol. Biol. 247, 536-540,(1995).