Journal of Bioinformatics and Computational Biology
Vol. 5, No. 3 (2007) 739–753
© Imperial College Press

A HIDDEN MARKOV MODEL FOR PREDICTING PROTEIN INTERFACES

CAO NGUYEN

Department of Computer Science and Engineering
University of Colorado at Denver and Health Sciences
Denver, CO. 80217, USA
dcnguyen@ouray.cudenver.edu

KATHELEEN J. GARDINER

Eleanor Roosevelt Institute at the University of Denver and
Department of Biochemistry and Molecular Genetics
Department of Pediatrics
University of Colorado at Denver and Health Sciences Center
Denver, CO. 80262, USA
kgardine@du.edu

KRZYSZTOF J. CIOS

Department of Computer Science and Engineering
University of Colorado at Denver and Health Sciences Center
Department of Computer Science, University of Colorado at Boulder
Department of Preventive Medicine and Biometrics
School of Medicine, UCDHSC, and 4cData, LLC Denver, CO. 80217, USA
krys.cios@cudenver.edu

Received 31 August 2006
Revised 26 December 2006

Protein–protein interactions play a defining role in protein function. Identifying the sites of interaction in a protein is a critical problem for understanding its functional mechanisms, as well as for drug design. To predict sites within a protein chain that participate in protein complexes, we have developed a novel method based on the Hidden Markov Model, which combines several biological characteristics of the sequences neighboring a target residue: structural information, accessible surface area, and transition probability among amino acids. We have evaluated the method using 5-fold cross-validation on 139 unique proteins and demonstrated precision of 66% and recall of 61% in identifying interfaces. These results are better than those achieved by other methods used for identification of interfaces.

Keywords: Protein–protein interaction; protein interface; support vector machine; hidden Markov model; protein function; sequence profile.

1. Introduction

The goal of proteomics research is to understand protein functions. Critical to reaching this goal is information regarding protein–protein interactions. Protein interactions are essential for biological processes that range from the formation of macromolecular structures and enzymatic complexes to the regulation of signal transduction pathways. Protein interfaces are defined as domains for selective recognition between molecules and for the formation of complexes. Identification of these interfaces is important, not only for understanding protein function but also for understanding the effects of mutations and for efficient drug design. Currently, the only reliable method for predicting interfaces requires knowledge of tertiary structures of protein complexes. However, because such information requires sophisticated and expensive NMR and X-ray diffraction studies, only a small fraction of known protein sequences have tertiary structures determined. Thus, there is a need to develop predictive methods for identifying interfaces.[1]

A number of researchers have analyzed features of protein interface sites. Chothia and Janin[2] determined that such sites are more hydrophobic, flat or protruding than other surfaces. Jones and Thornton[3] studied properties of protein surfaces surrounding interacting residues and concluded that several physical characteristics, such as Accessible Surface Area (ASA), were very influential. Sheinerman and Honig[4] looked at the contribution of electrostatic interactions, the presence of a significant population of charged and polar residues, on interfaces. Ofran and Rost[5] found that amino acid composition and preferences for residue-residue interactions could be used to differentiate six types of protein interfaces, each corresponding to a different functional association between residues.

Based on the features of known protein interfaces, several computational methods have been reported for their prediction. One approach was based on physical and chemical characteristics of protein interfaces. Jones and Thornton[6] predicted interface patches using the sum of values of *solvation, residue interface propensity, hydrophobicity, flatness, protrusion index and ASA*. The algorithm by Kini and Evans[7] used the fact that proline residues occurred near interfaces at a frequency 2.5 times higher than expected by chance. Pazos *et al.*[8] used multiple sequence alignments and correlated mutations based on the hypothesis that residues participating in functionally required interactions tend to show compensatory mutations during evolution. Gallet *et al.*[9] reported that interfaces can be predicted by using the hydrophobic moment and averaged hydrophobicity.

Another approach to predict interfaces is to use machine learning methods, such as neural networks, Bayesian statistics and support vector machines (SVM), with sequence profiles and/or ASA as input data. Zhou and Shan[10] and Fariselli *et al.*[11] used a neural network to learn whether or not exposed residues at the protein surface were in a contact patch. Koike and Takagi[12] applied SVM to classify interfaces of residues for both homo and hetero-dimers. Bradford and Westhead[13] combined a support vector machine approach with the previous work of Jones and Thornton[6] to predict protein-protein binding site. Bordner and Abagyan[14] detected protein interfaces by using a SVM trained on a combination of evolutionary conservation signals plus local surface properties. Chen and Zhou[15] extended their own previous work[10] by developing a consensus method that combined two neural networks consecutively. Yan *et al.*[16] reported a two-stage method in which residue clusters belonging to interfaces were detected through the SVM and then used as input to a Bayesian network classifier to identify the most likely class, interface or non-interface, for the target residues.

The above machine learning approaches share a common characteristic: the input to their models is the encoded identities of n contiguous amino acid residues, corresponding to a window of size-n containing the target residue. Each residue in the window is represented by a 20-dimensional vector whose values are the corresponding frequencies in sequence profiles of the residue (e.g. in the method of Yan *et al.*, the 20-dimensional vector is a 20-bit vector with 1-bit for each letter-code of the 20 amino acid alphabet). These methods are limited because a large amount of data is needed for training, and more importantly, because the biological relationships between sequence profiles and interfaces are not well established.

Here, we report a novel approach to the problem of predicting protein interfaces that takes into account biological characteristics of a target residue and its nearest neighbors. These include structural information, transition probability among amino acids around the target residue (in both strands from the N-terminal to C-terminal and in the opposite direction), and ASA value. We designed a Hidden Markov Model (HMM) to combine and reflect all these characteristics. Our method achieved accuracy of 73%, precision of 66%, recall of 61% and correlation of 0.42, on a 139 member protein dataset using 5-fold cross-validation.

2. Methods

2.1. *Material*

The method of collecting protein sequence data for the purpose of training and testing of our HMM is similar to those reported previously.[10,12] We extracted non-homologous proteins from all multiple chain protein entries in the PDB (December 2005 release) by using the PSI-BLAST[19] with the cutoff of identity $< 30\%$ and E-value > 0.14. Also with a confidence level of $> 95\%$, we excluded all chains shorter than 50 residues as well as those longer than 1100 residues. After this filtering, this dataset consists of 139 nonhomologous complex-forming protein chains in the PDB including 68 homo-dimers and 71 hetero-dimers. The list of proteins is available at http://isl.cudenver.edu/hmm/139pdb.htm. The set of 77 protein sequences used by Yan *et al.* forms the second dataset.

2.2. *Definition of an interface*

There are several methods to detect which residues in a protein are part of the protein interface. One method is to identify all residues within one member of a

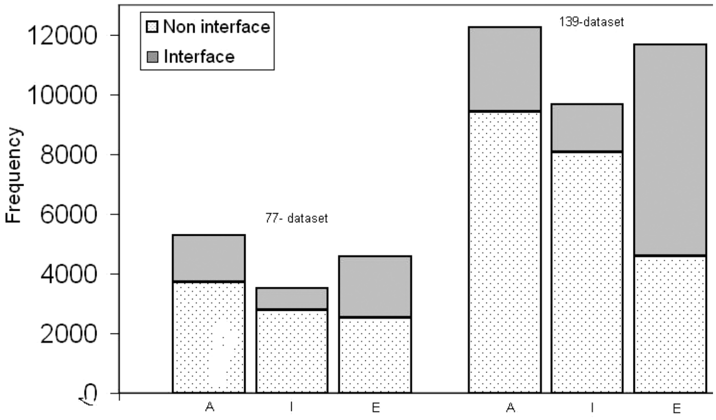


Fig. 1. Example distribution of interfaces and non-interfaces according to the three categories in the two datasets, where A is Ambivalent, I is Internal, and E is External.

complex that lie within 5 Å of a residue within a second member of the complex.[10,12] In this work, we used the definition of Jones and Thornton,[3] namely, we calculated the difference in the ASA upon complex formation. Using the DSSP program of Kabsch and Sander,[20] we extracted the ASA for each residue in the subunits forming the complex and in the complex. When the ratio of ASA of a residue to its nominal maximum area in the subunit is at least 25%, the residue is called a surface residue. A surface residue is defined as an interface residue when its calculated ASA value in the complex is decreased by at least 1 Å^2. By using this definition, also used in Yan *et al.*,[16] we obtained a total of 11 555 interfaces from a total of 33 632 residues in the 139 protein dataset (see Fig. 1).

2.3. *Architecture of the HMM*

A first-order discrete HMM is a stochastic generative model for linear problems such as sequences or time series defined by a finite set $\mathtt{D}$ of states, a discrete alphabet S of symbols, a probability transition matrix $\mathtt{T} = [t_i(j)]$ and a probability emission matrix $\mathtt{E} = [e_k(b)]$. Each state k emits symbol b according to $\mathtt{E}$, creating an observable sequence of symbols. The states are interconnected by state transition probabilities $\mathtt{T}$. Starting from an initial state, a sequence of states is generated by moving from state to state according to the transition probabilities $\mathtt{T}$ until the end state is reached.[21]

Based on structural information, a protein sequence is reduced from 20-letter amino acid alphabet to three-letter categories of biochemical similarity[22]: ambivalent (Ala, Cys, Gly, Pro, Ser, Thr, Trp, Tyr), external (Arg, Asn, Asp, Gln, Glu, His, Lys), and internal (Ile, Leu, Met, Phe, Val) (see Fig. 1). We observe that, for both datasets, the fraction of interface proteins in the external category is much

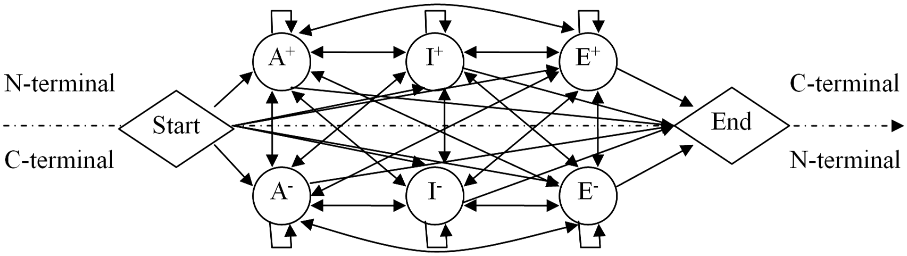


Fig. 2. HMM architecture: there are six inter-connected states plus two fictitious start/end states using the three categories. The state designated with a superscript "+" emits symbols located in interface domains. The emission probability and transition probability parameters are trained by converted protein sequences in both strands, from N-terminal to C-terminal, and in the opposite direction.

higher than that in the internal or ambivalent categories (see Fig. 1). This grouping not only significantly reduces the dimensionality of the input space but it also increases the accuracy of the HMM. For completeness we also show in Table A.1 and Fig. 7, in the Appendix, the results obtained by using different HMM models, corresponding to four different monomer groupings based on: (a) chemical, (b) functional, (c) charge, and (d) hydrophobic alphabets. In our HMM method, the states of the class D are defined as $\mathtt{D} = \{\mathrm{A}^+, \mathrm{E}^+, \mathrm{I}^+, \mathrm{A}^-, \mathrm{E}^-, \mathrm{I}^-\}$ and the set of symbol alphabet are defined as $\mathtt{S} = \{\mathrm{A, E, I}\}$, where A stands for ambivalent, E for external and I for internal (see Fig. 2).

Given the input sequence $x = x_1 x_2 \cdots x_N$ with $x_i \in \mathtt{S}$, we want to find the output state sequence $\pi = \pi_1 \pi_2 \cdots \pi_N$ with $\pi_i \in \mathtt{D}$ such that the probability $P(x, \pi|\theta)$ is maximized: $\pi^* = \mathrm{argmax}_\pi\ P(x, \pi|\theta)$ with the parameters θ are $\mathtt{E}$ and $\mathtt{T}$ matrices.

2.4. ***Training of the HMM and making predictions***

2.4.1. *Training*

To train the model the 139 amino-acid letter sequences are first converted into the three category letter sequences (A, I, E). Next, we use two *operators* on each sequence. The *expansion* operator expands each residue in the three-category-letter sequence into an odd k-residue window containing the residue and $\lfloor k/2 \rfloor$ neighboring residues, on both sides of the residue. It should be noted that when windows i and $i + 1$ are connected the last residue of window i and the first residue of window $i + 1$ are put together, which results in a residue pair. Because this pair does not exist in the original sequence it is eliminated from training data. The *contraction* operator cuts the expanded sequence into connected windows of size m, where each window must have in the middle an interface residue, and $\lfloor m/2 \rfloor$ neighboring residues on either side of the interface. In using the contraction operator the following rule is applied. Let window i be a window formed with an interface residue in

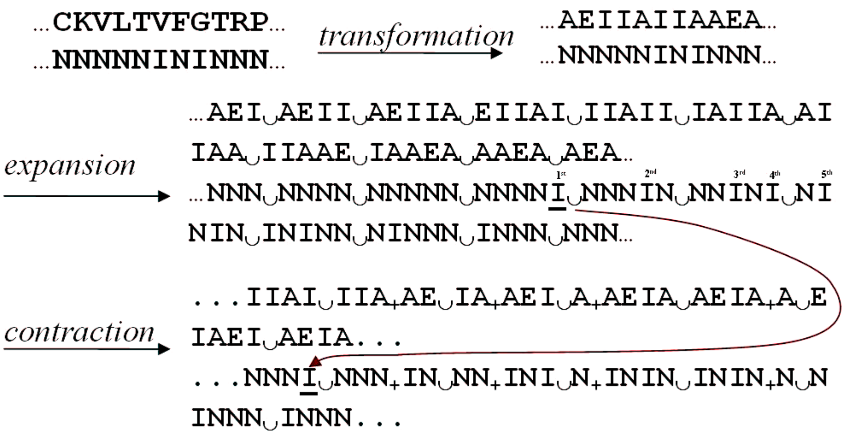

...CKVLTVFGTRP...
...NNNNNININNN...
transformation
...AEIIAIIAAEA...
...NNNNNININNN...
expansion
...AEI∪AEII∪AEIIA∪EIIAI∪IIAII∪IAIIA∪AI
IAA∪IIAAE∪IAAEA∪AAEA∪AEA...
1st
2nd
3rd
4th
5th
...NNN∪NNNN∪NNNNN∪NNNNI∪NNNNIN∪NNINI∪NI
NIN∪ININN∪NINNN∪INNN∪NNN...
contraction
...IIAI∪IIA+AE∪IA+AEI∪A+AEIA∪AEIA+A∪E
IAEI∪AEIA...
...NNNI∪NNN+IN∪NN+INI∪N+ININ∪ININ+N∪N
INNN∪INNN...

Fig. 3. Example of transformation from an original sequence into a training sequence that constitutes input to the model. In the expansion (with $k = 5$), the 1st letter **A** is expanded to **A**EI (because it has only two residues on the right flank side), the 2nd **E** is expanded to A**E**II (because it has one residue on the left and two residues on the right flank side), and the 3rd **I** is expanded to AE**I**IA (because it has two residues on both flank sides), and so on. The transition between the last residue of window i and the first residue of window $i+1$ (marked as $\cup$) is not used in training task. The contraction operator (with $m = 7$) cuts: the 1st observed interface **I** into a size-7 window NNN**I**$\cup$NNN; the 2nd observed interface **I** into a size-7 window NNN**I**N$\cup$NN, merged with the previous window (marked as +) to become NNN**I**$\cup$NNN+**I**N$\cup$NN; the 3rd observed interface **I** into N$\cup$NN**I**NI$\cup$N, merged with the previous window (marked as +) to become NNN**I**$\cup$NNN+**I**N$\cup$NN+**I**NI$\cup$N. The 4th observed interface **I** belongs to the previous window, thus *no window is formed for the 4th interface*, and so on.

the middle, then if the next observed interface residue belongs to this window i, the said interface residue is omitted, and as a result, no window is formed. Otherwise, window $i+1$ is formed with the said interface in the middle and merged with window i (see Fig. 3). These two operators exploit the fact that interface residues tend to form clusters within the amino acid sequence.[23] First, the expansion operator expands residues around the target residues, and thus improves the transition probability among the amino acids around the target residues. Second, the contraction operator cuts the expanded sequence into connected windows, where the middle residue is an interface. Thus, we again focus on clustering of interface residues. We empirically determined values of $k = 5$ for the first operator and of $m = 7$ for the second. The transformed sequences are input into the model in both strands from the *N_terminal* to the *C_terminal*, and in the reverse direction, to learn the emission and transition probabilities, E and T.

2.4.2. *Predictions*

A protein sequence is converted into the three-category letter sequence before it is used as an input to the model. In this phase, *the two operators will not be used* because we do not have the knowledge of interfaces of the protein. Then we use the Viterbi algorithm,[21] to find the most probable path of output state sequence $\pi = \pi_1\pi_2 \cdots \pi_N$. The initial state π_1 is chosen according to an initial distribution of the states. Interfaces are residues emitted by those states designated with a superscript "+".

2.5. *Assessment of predictions*

The following four measures are used to assess the performance of our HMM method:

$$\text{Precision, or } \textit{positive predictive value} = \frac{TP}{TP+FP},$$

$$\text{Recall, or } \textit{sensitivity} = \frac{TP}{TP+FN},$$

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN},$$

Matthews correlation coefficient

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

where

- — TP = the number of residues predicted to be interfaces that actually are interfaces (true positives)
- — TN = the number of residues predicted not to be interfaces that actually are not interfaces (true negatives)
- — FP = the number of residues predicted to be interfaces that actually are not interfaces (false positives)
- — FN = the number of residues predicted not to be interfaces that actually are interfaces (false negatives)

The Matthews correlation coefficient (MCC) is always between -1 and $+1$ and measures how well the predicted class labels agree with the actual class labels. A value of the MCC of -1 means complete disagreement, $+1$, complete agreement, and 0 is considered as a random prediction.[17] Accuracy gives the overall evaluation of the estimated probability of correct predictions. In this study, we did not use accuracy because accuracy is sensitive to the distribution of interface proteins and our data is skewed (the negative class $\gg$ positive class). Thus if we sacrificed true positives to predict all examples as negative we could have obtained high accuracy of a classifier.[24–26]

3. Results and Discussion

3.1. *HMM prediction results*

We trained our HMM by using a five-fold cross-validation on the dataset of 139 proteins. Table 1 summarizes the results in terms of the following performance

Table 1. Performance of the HMM method on the 139 protein dataset using five-fold cross-validation.

	HMM Method			Two-Stage Method of Yan *et al.*[16]	
	Homo-Dimers	Hetero-Dimers	Overall	First Stage (SVM)	Second Stage (Bayesian)
Precision	0.65	0.68	0.66	0.61	0.39
Recall	0.64	0.57	0.61	0.59	0.40
MCC	0.43	0.41	0.42	0.40	0.10
Accuracy	0.74	0.72	0.73	0.73	0.65

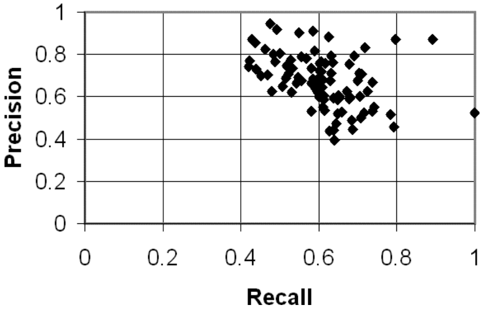


Fig. 4. Distribution of precision and recall of prediction result for 139 proteins using five-fold cross-validation.

measures: MCC, precision, recall and accuracy. The method's accuracy is 73% with a correlation coefficient of 0.42. This result shows that it performs significantly better than a random predictor.[14] Of the residues predicted as interfaces, 66% are actual interfaces (precision), and of all interfaces, 61% are identified as such (recall). Figure 4 shows precision and recall for each predicted protein data point in our dataset. Thus, the HMM method is a good predictor.

Work of other authors[5,6,11,15] indicated that the characteristics of homo- and hetero-dimer interfaces are different. We separately tested homo- and hetero-dimers in our dataset and found that the performance results of homo-dimers are slightly better than those of hetero-dimers.

3.2. *Comparison with reported result*

Zhou and Shan[10] predicted interfaces using neural networks. The input values were sequence profiles and ASA of the target residue and the neighbor residues. The precision and recall for their classifier was 51% and 50%, respectively. Koike and Takagi[12] used SVM with sequential neighboring profiles, i.e. profiles of residues in the neighborhood, and reported precision and recall of 50% and 54–56%, respectively, for a homo-hetero mixed test data. However, because of differences in definitions, results of these methods cannot be directly compared with our results.

For direct comparison, we therefore tested our HMM method on the 77 protein dataset reported by Yan *et al.*[16] We selected Yan *et al.*'s method for comparison because it used open source SVM and Bayesian network implementations,[18] and thus was easy to replicate. To the best of our knowledge, the two-stage classifier method of Yan *et al.* has given the best results so far. Using exactly the same definition to extract the interfaces, we evaluated the two-stage classifier method of Yan *et al.* using five-fold cross-validation on our 139 protein dataset.

3.2.1. *HMM comparison with the two-stage classifier on 139-protein dataset*

To evaluate the method of Yan *et al.* on our 139 protein dataset, we follow their procedure and use open-source SVM and Bayesian function implementations. The input to the SVM consists of the encoded identities of nine amino acid residues consisting of the target residue and the four residues on each side. Each residue in the window is represented by a 20-bit vector, with 1-bit for each letter representing the 20 amino acids. Thus, each input data point is of dimension 9×20 plus class attribute ($1 =$ interaction site, $0 =$ noninteraction site). The results are shown in Table 1. They indicate that, in terms of precision, recall and MCC, the performance measures of our HMM are slightly better than the SVM used in the first stage.

Next, we evaluate the SVM method by using a different way of encoding residues.[10,12] Each residue in the window, instead of association only with nearest neighbor information, as in the method of Yan *et al.*, is now coded as a 20-dimensional vector, with its components corresponding to frequencies in the multiple sequence alignment of the protein, taken from the HSSP file.[27] When using these input vectors, the SVM's performance is 58% precision, 38% recall, and 0.27 MCC. These values are worse than using only the nearest-neighbor information as input. Thus, the biological relationships between sequence profiles and interfaces are still questionable.

In the second stage, to compare with the method of Yan *et al.*, a Bayesian network classifier is trained to identify an interaction site residue based on the class labels ($1 =$ interface, $0 =$ non-interface) of its neighbors. The class label outputs of the SVM in the first stage constitute input to the Bayesian classifier in the second stage. Table 1 shows the results of the second stage using BayesNetB from the Weka package,[18] with the input being the class labels z of the eight residues surrounding a target residue, four on each side. The method of Yan *et al.* classifies the target residue as an interface if $p(1|z) > \lambda \cdot p(0|z)$, where λ ranges from 0.01 to 1, in increments of 0.01, so as to maximize the correlation coefficient. The HMM performed better on the original 139 proteins than the method of Yan *et al.* using all measures.

3.2.2. *HMM comparison with two-stage classifier on 77-protein dataset*

To compare Yan *et al.*'s method, we use their datasets (77 proteins) and results. The results in Table 2 shows that our HMM method achieves higher recall (53%) than the two-stage method of Yan *et al.* (39%) when tested on their 77-protein

Table 2. Comparison of the HMM method with two other methods using the 77-protein data set.

		Two-Stage Method of Yan *et al.*[16]		
	HMM Method	First Stage (SVM)	Second Stage (Bayesian)	Gallet *et al.*[19] Method
Precision	0.58	0.44	0.58	0.30
Recall	0.53	0.43	0.39	0.44
MCC	0.30	0.19	0.30	○0.02
Accuracy	0.68	0.66	0.72	0.51

dataset, while the other performance measures are equal. Also, the performance measures of the HMM method are significantly better than the method of Gallet *et al.*[9] (we use results provided by the authors) when tested on the same dataset. All supplementary materials, including the HMM program and datasets, are available at our web site at http://isl.cudenver.edu/hmm.

4. Conclusions

On the dataset of 139 proteins, our six-state HMM method achieved 73% accuracy, 66% precision, 61% recall, and Matthews correlation coefficient of 0.42. Results of other HMM variations we designed are shown in Table A.1. In Fig. 5 we illustrate the performance of the six-state HMM method in terms of tertiary structure of two complexes: the PDB 1BM9 and 1EFN. In several comparisons, our method performed better than the two-stage method of Yan *et al.* on their dataset of 77 proteins, as well as when testing their method on our 139 proteins dataset. We have also shown that if we use sequence profiles to encode the residues,[10,12] the performance of the SVM method substantially decreases. This suggests that the HMM method might be used for predicting effects of point mutations on protein interactions, or solvent accessibility, from protein sequences.

Validation with the CAPRI targets

Critical Assessment of Prediction of Interactions (CAPRI: http://capri.ebi.ac.uk) is a community wide experiment to assess the capacity of protein docking algorithms on targets based on structures of the unbound components. To evaluate our method, we trained the HMM by using the 139 protein dataset and used the resulting classifier to identify the protein interfaces on the targets. The prediction of the ligand HEMK in the target 20 (protein Methyltransferase HEMK complexed with Release Factor 1, hetero-dimer) is shown in Fig. 6. The HMM identified 32 interfaces out of 59 positive class residues and 132 non-interfaces out of 217 negative class residues.

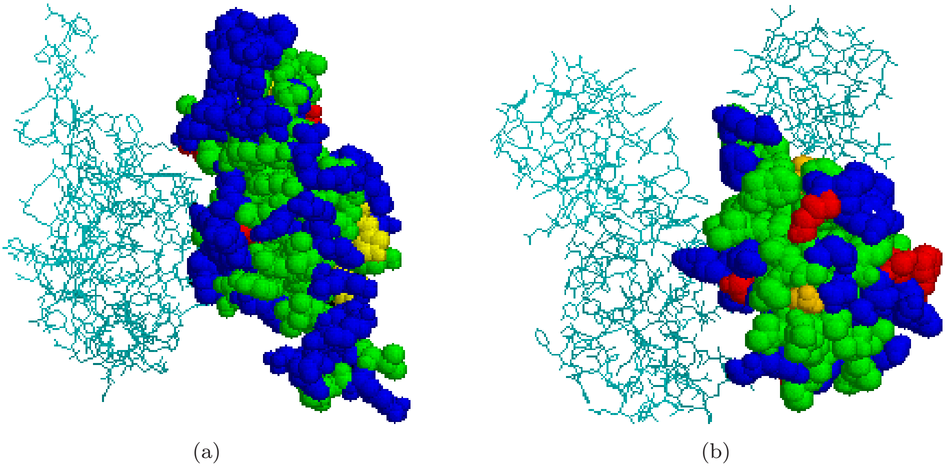


Fig. 5. Overall view of complexes with identified interfaces. The predicted protein in each complex is shown in space-fill coded as: Blue: interfaces classified correctly (TPs), Yellow: residues incorrectly classified as interfaces (FPs), Green: non-interfaces correctly identified as such (TNs), Red: residues incorrectly identified as non-interfaces (FNs); binding proteins are shown as wire-frames: (a) 1BM9 (replication terminator protein from Bacillus Subtilis) (homo-dimer): the predicted protein (chain A), the binding protein (chain B) (b) 1EFN (HIV-1 NEF protein in complex with R96i mutant FYN SH3 domain) (hetero-dimer): the predicted protein (chain B), the binding protein (chains A, C, D). The HMM identifies more interfaces in the homo-dimer. The figures were created using RasMol (http://www.openrasmol.org).

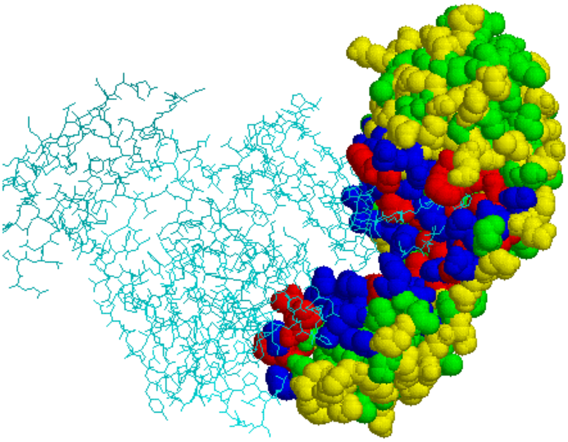


Fig. 6. Overall view of the ligand HEMK in the target 20 (hetero-dimer) with identified interfaces. The residues of interest are shown in space-fill coded as: blue, TPs; yellow, FPs; green, TNs; red, FNs; binding proteins are shown as wire-frames. The figure was created using RasMol (http://www.openrasmol.org).

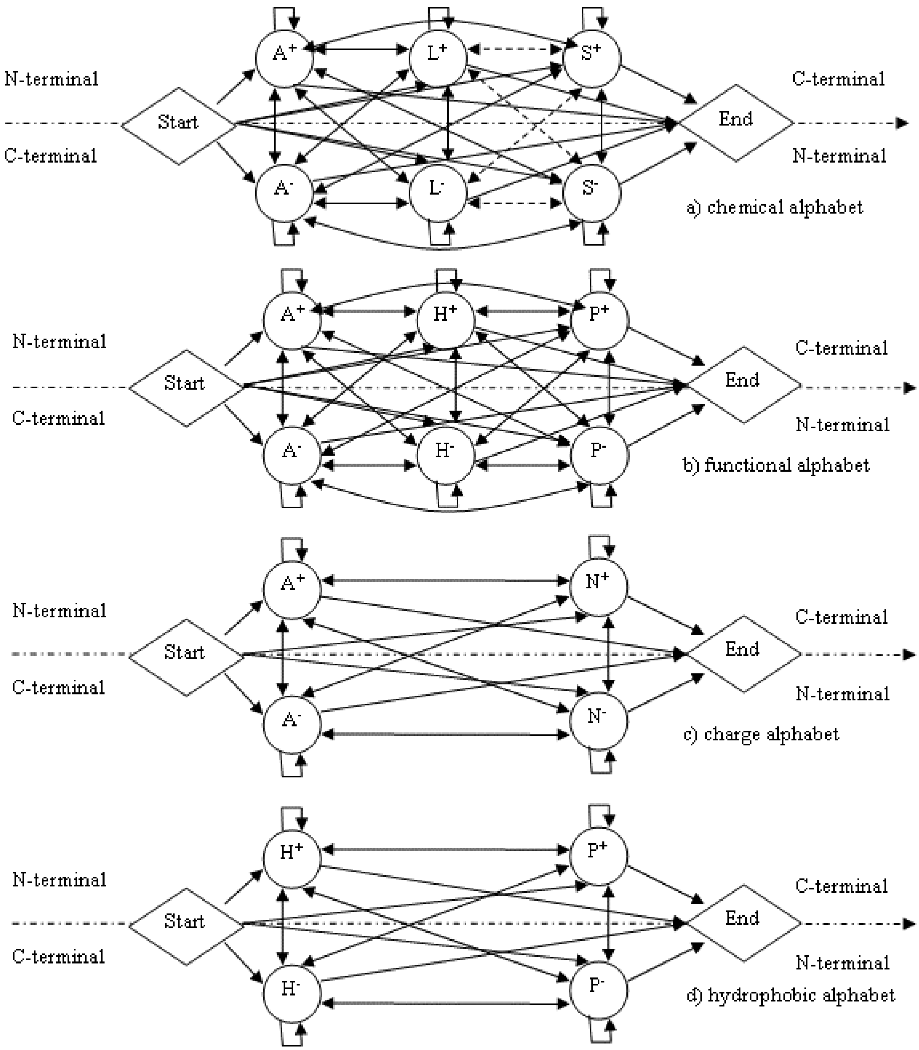


Fig. 7. HMM architectures for monomer grouping based on: (a) *Chemical alphabet*: there are sixteen inter-connected states $(A^+, L^+, M^+, R^+, B^+, H^+, I^+, S^+, A^-, L^-, M^-, R^-, B^-, H^-, I^-, S^-)$ plus two fictitious *start/end* states according to the eight categories. (b) *Functional alphabet*: there are six inter-connected states $(A^+, H^+, P^+, A^-, H^-, P^-)$ plus two fictitious *start/end* states according to the three categories. (c) *Charge alphabet*: there are four inter-connected states (A^+, N^+, A^-, N^-) plus two fictitious *start/end* states according to the two categories. (d) *Hydrophobic alphabet*: there are four inter-connected states (H^+, P^+, H^-, P^-) plus two fictitious *start/end* states according to the two categories. The state designated with a superscript "+" emits symbols located in interface domains. The emission probability and transition probability parameters are trained by converted protein sequences in both strands, from N-terminal to C-terminal, and in the opposite direction.

Acknowledgments

The authors acknowledge support from the Vietnamese Ministry of Education and Training (CN), the National Institutes of Health (HD047671 and HD49460) (KG), and the Butcher Foundation (KJC and CN).

Appendix

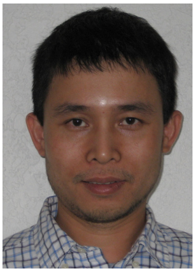
Table A.1. Results of the HMM method when monomers are grouped by:

	77 Proteins	139 Proteins
(1) *Chemical alphabet*: (*A*) acidic (Asp, Glu), (*L*) aliphatic (Ala, Gly, Ile, Leu, Val), (*M*) amide (Asn, Gln), (*R*) aromatic (Phe, Trp, Tyr), (*B*) basic (Arg, His, Lys), (*H*) hydroxyl (Ser, Thr), (*I*) imino (Pro), (*S*) sulfur (Cys, Met)		
Precision	0.53	0.66
Recall	0.51	0.61
MCC	0.24	0.42
(2) *Functional alphabet*: (*A*) acidic and basic (Asp, Glu, Arg, His, Lys), (*H*) hydrophobic nonpolar (Ala, Ile, Leu, Met, Phe, Pro, Trp, Val), (*P*) polar uncharged (Asn, Gln, Ser, Thr, Pro, Cys, Met)		
Precision	0.59	0.69
Recall	0.39	0.49
MCC	0.26	0.39
(3) *Charge alphabet*: (*A*) acidic and basic (Asp, Glu, Arg, His, Lys), (*N*) neutral (Ala, Ile, Leu, Met, Phe, Pro, Trp, Val, Asn, Gln, Ser, Thr, Pro, Cys, Met)		
Precision	0.59	0.69
Recall	0.39	0.49
MCC	0.26	0.39
(4) *Hydrophobic alphabet*: (*H*) hydrophobic (Ala, Ile, Leu, Met, Phe, Pro, Trp, Val), (*P*) hydrophilic (Arg, Asn, Asp, Cys, Gln, Glu, Gly, His, Lys, Ser, Thr, Tyr)		
Precision	0.44	0.51
Recall	0.74	0.77
MCC	0.19	0.30

References

1. Valencia A, Pazos F, Computational methods for prediction of protein interactions, *Curr Opin Struct Biol* **12**:368–373, 2002.
2. Chothia C, Janin J, Principles of protein-protein recognition, *Nature* **256**:705–708, 1975.
3. Jones S, Thornton JM, Principles of protein–protein interaction, *Proc Natl Acad Sci USA* **93**:13–20, 1996.
4. Sheinerman FB, Honig B, On the role of electrostatic interactions in the design of protein–protein interfaces, *J Mol Biol* **318**:161–177, 2002.
5. Ofran Y, Rost B, Analysing six types of protein–protein interfaces, *J Mol Biol* **325**:377–387, 2003.

6. Jones S, Thornton JM, Prediction of protein–protein interaction sites using patch analysis, *J Mol Biol* **272**:133–143, 1997.
7. Kini RM, Evans HJ, Prediction of potential protein–protein interaction sites from amino acid sequence identification of a fibrin polymerization site, *FEBS letters* **385**:81–86, 1996.
8. Pazos F, Helmer-Citterich M, Ausiello G, Valencia A, Correlated mutations contain information about protein–protein interaction, *J Mol Biol* **271**:511–523, 1997.
9. Gallet X, Charloteaux B, Thomas A, Brasseur R, A fast method to predict protein interaction sites from sequences, *J Mol Biol* **302**:917–926, 2000.
10. Zhou H, Shan Y, Prediction of protein interaction sites from sequence profile and residue neighbor list, *Proteins* **44**:336–343, 2001.
11. Fariselli P, Pazos F, Valencia A, Casadi R, Prediction of protein-protein interaction sites in heterocomplexes with neural networks, *Eur J Bio-Chem* **269**:1356–1361, 2002.
12. Koike A, Takagi T, Prediction of protein–protein interaction sites using support vector machines, *Protein Engineering, Design & Selection* **17**:165–173, 2004.
13. Bradford J, Westhead D, Improved prediction of protein–protein binding sites using a support vector machines approach, *Bioinformatics* **21**(8):1487–1494, 2005.
14. Bordner A, Abagyan R, Statistical analysis and prediction of protein–protein interfaces, *Proteins: Structure, Function, and Bioinformatics* **60**(3):353–366, 2005.
15. Chen H, Zhou H, Prediction of interface residues in protein–protein complexes by a consensus neural network method: Test against NMR data, *Proteins: Structure, Function, and Bioinformatics* **61**:21–35, 2005.
16. Yan C, Dobbs D, Honavar V, A two-stage classifier for identification of protein-protein interface residues, *Bioinformatics* **20**:371i–378i, 2004.
17. Baldi P, Brunak S, Chauvin Y, Andersen F, Assessing the accuracy of prediction algorithms for classification: An overview, *Bioinformatics* **16**:412–424, 2000.
18. Witten I, Frank E, *Data Mining: Practical Machine Learning Tools and Techniques*, Morgan Kaufmann, 2005.
19. Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ, Gapped BLAST and PSI-BLAST: A new generation of protein database search programs, *Nucleic Acids Res* **25**:3389–3402, 1997.
20. Kabsch W, Sander C, Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features, *Biopolymers* **22**:2577–2637, 1983.
21. Baldi P, Brunak S, *Bioinformatics: The Machine Learning Approach*, The MIT Press, 2001.
22. Karlin S, Ost F, Blaisdell BE, Patterns in DNA and amino acid sequences and their statistical significance, Waterman MS (ed.), in *Mathematical Methods for DNA Sequences*, CRC Press, pp. 133–157, 1989.
23. Ofran Y, Rost B, Predicted protein–protein interaction sites from local sequence information, *FEBS Lett* **544**:236–239, 2003.
24. Cios KJ, Pedrycz W, Swiniarski R, *Data Mining Methods for Knowledge Discovery*, Kluwer Academic Publishers, 1998.
25. Cios KJ, Kurgan L, CLIP4: Hybrid inductive machine learning algorithm that generates inequality rules, *Information Sciences* **163**:37–83, 2004.
26. Kurgan L, Cios KJ, Scott D, Highly scalable and robust rule learner: Performance evaluation and comparison, *IEEE Transactions on Systems Man and Cybernetics, Part B* **36**:32–53, 2006.
27. Dodge C, Schneider R, Sander C, The HSSP database of protein structure-sequence alignments and family profiles, *Nucleic Acids Res* **26**(1):313–315, 1998.



Cao Nguyen is a PhD student in Computer Science and Information Systems program, with the option in Computational Biology, at the University of Colorado at Denver and Health Sciences Center (UCDHSC). He conducts research with Dr. Cios in the area of mathematical modeling (hidden Markov models, clustering, and fuzzy cognitive maps) for prediction of protein–protein interfaces and protein functions. Nguyen holds an MS degree in Computer Science from the Vietnam National University. He has been awarded full scholarship for his study in the U.S.A.

Katheleen J. Gardiner received her PhD degree from the University of Colorado and is currently a Professor at the Eleanor Roosevelt Institute at the University of Denver and an Adjoint Associate Professor in the Department of Biochemistry and Molecular Genetics at the UCDHSC. Dr. Gardiner is an internationally recognized researcher in the field of Down syndrome. Her specific research interests are in the identification of genes encoded by human chromosome 21 that contribute to learning and memory deficits in Down syndrome and the use of data from mouse and other model organisms to predict associated pathway perturbations. She has authored over 100 journal articles, meeting reports, and book chapters. Her research is currently funded by the National Institutes of Health and the Fondation Jerome Lejeune.

Krzysztof J. Cios received his MS and PhD degrees from the AGH University of Science and Technology, Krakow, the MBA degree from the University of Toledo, Ohio, and the DSc degree from the Polish Academy of Sciences. He is currently a Professor at the University of Colorado at Denver and Health Sciences Center, and Associate Director of the University of Colorado Bioenergetics Institute. He directs Data Mining and Bioinformatics Laboratory. Dr. Cios is a well-known researcher in the areas of learning algorithms, biomedical informatics and data mining. NASA, NSF, American Heart Association, Ohio Aerospace Institute, NATO, US Air Force and NIH have funded his research. He published 3 books, about 150 journal and conference articles and 12 book chapters; serves on editorial boards of *Neurocomputing, Journal of Integrative Neuroscience, IEEE Engineering in Medicine and Biology Magazine, International Journal of Computational Intelligence, and Biodata Mining*; edited 5 special issues of journals. Dr. Cios has been the recipient of the Norbert Wiener Outstanding Paper Award, the *Neurocomputing* Best Paper Award, the University of Toledo Outstanding Faculty Research Award, and the Fulbright Senior Scholar Award. Dr. Cios is a Foreign Member of the Polish Academy of Arts and Sciences.