University of Nebraska - Lincoln
DigitalCommons@University of Nebraska - Lincoln

CSE Technical reports | Computer Science and Engineering, Department of

1-1-2004

A Study in Modeling Low-Conservation Protein Superfamilies

Chang Wang
University of Nebraska - Lincoln

Stephen Scott
University of Nebraska - Lincoln, sscott2@unl.edu

Jun Zhang
University of Nebraska - Lincoln, jzhang8@unl.edu

Qingping Tao
University of Nebraska - Lincoln

Dmitri E. Fomenko
University of Nebraska - Lincoln, dfomenko2@unl.edu

See next page for additional authors

Follow this and additional works at: http://digitalcommons.unl.edu/csetechreports

Part of the Computer Sciences Commons

Wang, Chang; Scott, Stephen; Zhang, Jun; Tao, Qingping; Fomenko, Dmitri E.; and Gladyshev, Vadim N., "A Study in Modeling Low-Conservation Protein Superfamilies" (2004). *CSE Technical reports.* Paper 35.
http://digitalcommons.unl.edu/csetechreports/35

This Article is brought to you for free and open access by the Computer Science and Engineering, Department of at DigitalCommons@University of Nebraska - Lincoln. It has been accepted for inclusion in CSE Technical reports by an authorized administrator of DigitalCommons@University of Nebraska - Lincoln.

Authors
Chang Wang, Stephen Scott, Jun Zhang, Qingping Tao, Dmitri E. Fomenko, and Vadim N. Gladyshev

This article is available at DigitalCommons@University of Nebraska - Lincoln: http://digitalcommons.unl.edu/csetechreports/35

A Study in Modeling Low-Conservation Protein Superfamilies

Chang Wang, *Student Member, IEEE,* Stephen D. Scott, *Member, IEEE,*
Jun Zhang, Qingping Tao, Dmitri E. Fomenko, and Vadim N. Gladyshev

Technical Report TR-UNL-CSE-2004-0003

Abstract

We present several algorithms for identification of new proteins in superfamilies with low primary sequence conservation. The low conservation of primary sequence in protein superfamilies such as Thioredoxin-fold (Trx-fold) makes conventional methods such as hidden Markov models (HMMs) difficult to use. Therefore, we use structural properties to build our classifiers. These structural properties include secondary structure patterns as well as various properties of the residues in the protein sequences. We use this information to model proteins via hidden Markov models, support vector machines and algorithms in the multiple-instance learning model. In 20-fold jack-knife tests, some of our models performed well, with relatively high true positive and true negative rates. We can identify 75% of the Trx-fold proteins in this jack-knife test (compared to only 5% for HMMs on primary sequence) while maintaining a 75% true negative rate. Since our techniques are general, they should be applicable to other superfamilies with low primary sequence conservation.

Index Terms

low primary sequence conservation, hidden Markov models, multiple-instance learning, support vector machines, thioredoxin-fold proteins, redox proteins.

I. Introduction

We study the problem of identifying new proteins in superfamilies whose primary sequence conservation is so low that conventional approaches (e.g. building hidden Markov models on

C. Wang, S. Scott, J. Zhang, and Q. Tao are with the Dept. of Computer Science, University of Nebraska, 115 Ferguson Hall, Lincoln, NE 68588-0115

D. Fomenko and V. Gladyshev are with the Dept. of Biochemistry, University of Nebraska, N151 Beadle Center, 1901 Vine St., Lincoln, NE 68588-0664

```
                    *   *
1A8L:   KLIVFVRKDHCQYCDQLKQLVQEL
1BED:   PVVSEFFSFYCPHCNTFEPIIAQL
1QK8:A  LVFFYFSASWCPPCRGFTPQLIEF
1F9M:A  PVVLDMFTQWCGPCKAMAPKYEKL
1MEK:   YLLVEFYAPWCGHCKALAPEYAKA
```

Fig. 1. Alignment of segments of five Trx-fold proteins, indexed by PDB ID.

primary sequence) are ineffective. For our experiments, we focus on the thioredoxin-fold (Trx-fold) superfamily.

Oxidation-reduction reactions in cells are catalyzed by various redox proteins, many of which use catalytic cysteine residues. Thiol-dependent redox proteins regulate many basic cellular processes, such as DNA synthesis, apoptosis, signal transduction and transcription [12], [5]. To understand the mechanism of cellular redox regulation, the first step is to identify redox proteins and to characterize the specific functions of these proteins [5], [2]. The thioredoxin superfamily is the major family of thiol-dependent oxidoreductases involved in cellular regulation, and its characterization is important for understanding of redox processes. In addition to thioredoxin, it includes protein disulfide isomerases, glutaredoxins, nucleoredoxins, peroxiredoxins, glutathione peroxidases and other redox enzymes.

Inter-family similarity within the Trx-fold superfamily is generally low, and sequence analysis tools such as SAM [14] cannot easily identify new families in the Trx-fold superfamily. For example, In Figure 1, active site segments of five Trx-fold proteins are shown. Only the two cysteines (C, marked by asterisks) are conserved in the alignment. These two cysteines form a redox motif designated the CxxC motif. This motif is conserved in the majority members of the superfamily, including thioredoxins, glutaredoxins, protein disulfide isomerases and other proteins. However, some of the Trx-fold proteins conserve other motifs (e.g. CxxS, SxxC, CxxT and TxxC).

In a more rigorous evaluation of the low primary sequence conservation of this superfamily, we used SAM to attempt to identify distinct Trx-fold protein families based on primary sequence alone (Section III-B) by running jack-knife tests on sets of highly dissimilar sequences. In these tests, only 5% of distinct Trx-fold proteins were correctly identified, indicating that

primary structure alone is insufficient in identification of Trx-fold protein families. SAM's poor performance is directly related to the lack of a good multiple alignment of the sequences: neither SAM nor Clustal were able to find a good primary sequence-based alignment of such highly dissimilar sequences (see Scott et al. [20] for results using Clustal).

In addition to the conserved motif mentioned above, for secondary structure, three α-helices and four β-sheets are organized in a specific pattern (a β-α-β-α-β-β-α motif). For most sequences, the CxxC motif is located between the first β-strand and the first α-helix in the fold, so the entire motif is β-CxxC-α-β-α-β-β-α [17], [13]. Therefore, even though the protein primary sequences are not conserved, one can use structural information to discriminate Trx-fold proteins. It should be noted, however, that some Trx-fold proteins allow insertions and deletions of secondary structures, which complicate the searches.

To compensate for the lack of primary sequence conservation, we use structural properties to identify new protein families. These structural properties include secondary structure patterns, as well as various properties of the residues in the protein sequences. We use this information to model proteins via hidden Markov models (HMMs) [14], support vector machines (SVMs) [19], and an algorithm [21] in the *multiple-instance learning model* [7]. The latter approach produced our strongest results, though a combination of HMMs and SVMs also performed well.

In a 20-fold jack-knife test on Trx-fold proteins, the three MIL approaches (motif-based alignment method, secondary-based alignment method and α-β signature method; see Section II-C) achieved 75% , 70% and 70% true positive rates (respectively) and 75%, 70% and 76.1% true negative rates. The hidden Markov models based on predicted secondary structure (see Section II-A) achieved true positive rates above 50% and true negative rates above 80%. The true positive and true negative rates of our SVM (Section II-B) were 50% and 88%. By combining the last two methods, we could identify 75% of the Trx-fold proteins in the jack-knife test with a true negative rate of 73%, making this combination comparable to MIL. Since our techniques are not specific to the Trx-fold superfamily, we believe that these techniques should be applicable to other superfamilies with low primary sequence conservation, especially when there is other conservation within the superfamily, e.g. secondary structure.

The rest of this paper is as follows. In Section II we describe the algorithms we employ in our study. Then in Section III we summarize our experimental results. Finally, we conclude in Section IV with a discussion of future work.

II. Our Algorithms

We apply three fundamental approaches to this problem. The first employs hidden Markov models (HMMs), but the models are built on structural information rather than on primary sequence. The second approach involves deriving summary statistics on structural information on the sequences (similar to that used in the QFC algorithm [15]) and using these statistics as attributes to an SVM, which is a robust algorithm for classification. In our third approach we treat this problem as a *multiple-instance* problem in machine learning [7] and apply a new algorithm [21] to learn a classifier that will separate Trx-fold proteins from non-Trx-fold proteins. May all your publication endeavors be successful.

A. HMMs on Structural Information

Given the high conservation of secondary structure in the Trx-fold superfamily, it is natural to build hidden Markov models on secondary structures. In general, we do not expect to be able to use known secondary structures when classifying sequences, so we predict[1] secondary structure with PSI-PRED [18] and PREDATOR [9]. Thus we built our models on the reduced alphabet $\{\alpha, \beta, \mathrm{loop}\}$ rather than the 20 amino acids. Due to this, we replaced the prior distributions normally used (which assume that e.g. "A" means alanine) with new priors (specifically, Dirichlet mixture priors) that are based on our new alphabet and our sequences when mapped to this alphabet. Developing a new prior depends on having a good multiple alignment, so we built our priors on sequences from PDB, where secondary structure is exactly known. We selected PDB sequences with obvious Trx-fold characteristics (CxxC motif in primary structure and β-α-β-α-β-β-α motif in secondary structure), which made possible with SAM [14] a good multiple alignment based on secondary structure. Our method for construction of priors is based on the work of Sjölander et al. [22], which starts with a base prior and modifies it based on symbol frequencies in each column of the multiple alignment. Since no base priors are available for our alphabet, we used a uniform distribution over $\{\alpha, \beta, \mathrm{loop}\}$ as the base prior when building our new priors. We used these priors along with predicted secondary structure to build our models with SAM.

[1]For comparison purposes, we also built models on true structures and tested them on predicted structures.

While predicted secondary structure is a natural first approach, PREDATOR and PSI-PRED (like other structure prediction algorithms) have fairly high per-residue error rates. This introduces significant noise in remapped sequences and thus affects our model. Hence we also looked at other sequence mappings. Andorf et al. [1] and Wang et al. [24] remapped the 20-character amino acid alphabet to a reduced one that captures structural properties. They used the reduced alphabet representations of protein sequences in the data-driven discovery of sequence motif-based decision trees for classifying protein sequences into functional families. Their results raise the possibility that the use of different alphabets might provide different, but complementary, insights into protein structure-function relationships. So in addition to the remapping to secondary structure elements as outlined above, we remapped our sequences from the 20-character amino acid alphabet to a reduced one based on hydrophobicity, charge, volume and mass (Table I). Each column of Table I shows a criterion for remapping and the class that the particular residue was remapped to based on that criterion. For each of these remappings, we built an HMM with SAM.

B. Modeling with QFC-Based Summary Features

In the QFC algorithm [15], the physico-chemical properties of the amino acids in the molecules are characterized using various indices and standard measurements, such as GES hydropathy index [8], [11], solubility [4], polarity, pI, Kyte-Doolittle index [16], α helix index [6], and molecular weight. A protein sequence is described by a set of variables x_1 through x_n, and for each x_i, there is a value x_{ij} for the ith amino acid index (property) value at the jth position of the sequence. Thus x_{i1} through x_{im} constitutes a profile of the protein in terms of the ith amino-acid property index (e.g. Figure 2). Then each raw profile is smoothed by applying the Sliding Window Recognizer [23], which transforms the profile as follows: $x'_{ij} = \sum_{k=-d}^{d} w_{j-k} x_{j-k}$, where d is the kernel size and w is the kernel window.

We followed a procedure similar to the method used by Kim et al. [15]. We first computed moving window profiles of putative Trx-fold (for positive training data) and non-Trx-fold (for negative training data) proteins based on each property, and then smoothed the profiles with a width-16 Gaussian kernel. We then mapped each sequence's set of smoothed profiles to a set of attributes associated with that sequence. The *average periodicity* attributes describe how often each property's profile crosses a neutral value. For example, in Figure 2, we count the

TABLE I

DEFINITION OF THE REMAPPINGS OF THE 20-RESIDUE ALPHABET.

Residue	Charge	Volume	Mass	Hydro-4	Hydro-6
A	None	Small	Small	$[-2.0, 0.7]$	$[-0.6, -2.0]$
C	None	Medium	Medium	$[-2.0, 0.7]$	$[-0.6, -2.0]$
D	Neg	Medium	Med-Large	$[8.2, 12.3]$	$[8.2, 9.2]$
E	Neg	Med-Large	Med-Large	$[8.2, 12.3]$	$[8.2, 9.2]$
F	None	Large	Large	$[-3.7, -2.6]$	$[-3.7, -2.6]$
G	None	Small	Small	$[-2.0, 0.7]$	$[-0.6, -2.0]$
H	Neg	Med-Large	Med-Large	$[3.0, 4.8]$	$[3.0, 4.8]$
I	None	Med-Large	Med-Large	$[-3.7, -2.6]$	$[-3.7, -2.6]$
K	Pos	Med-Large	Med-Large	$[8.2, 12.3]$	$[8.2, 9.2]$
L	None	Med-Large	Med-Large	$[-3.7, -2.6]$	$[-3.7, -2.6]$
M	None	Med-Large	Med-Large	$[-3.7, -2.6]$	$[-3.7, -2.6]$
N	None	Medium	Med-Large	$[3.0, 4.8]$	$[3.0, 4.8]$
P	None	Medium	Medium	$[-2.0, 0.7]$	$[0.2, 0.7]$
Q	None	Med-Large	Med-Large	$[3.0, 4.8]$	$[3.0, 4.8]$
R	Pos	Med-Large	Large	$[8.2, 12.3]$	$[12.3, 12.3]$
S	None	Small	Medium	$[-2.0, 0.7]$	$[-0.6, -2.0]$
T	None	Medium	Medium	$[-2.0, 0.7]$	$[-0.6, -2.0]$
V	None	Med-Large	Medium	$[-3.7, -2.6]$	$[-3.7, -2.6]$
W	None	Large	Large	$[-2.0, 0.7]$	$[-0.6, -2.0]$
Y	None	Large	Large	$[-2.0, 0.7]$	$[0.2, 0.7]$

number of times the Kyte-Doolittle index crosses the neutral value 2.0 (44) and then divide this by the length of the sequence (104). So the value of attribute "crosscv-KD2.0" for 1fb0 is $44/104 = 0.423$. (For a complete list of the neutral values we used, see Table II.) These features were used to train a support vector machine (SVM) with a Guassian kernel.

In addition to QFC's summary statistics, we added features that summarize the predicted secondary structure. First we predicted each sequence's secondary structure as in Section II-A,

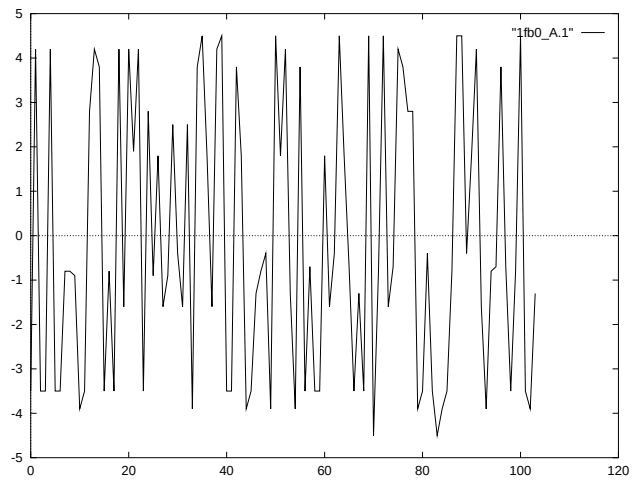

Fig. 2. Plot of a profile of 1fb0 from PDB, based on Kyte-Doolittle index. On the x axis is the amino acid position and on the y axis is the value of the index.

TABLE II

NEUTRAL VALUES OF THE QFC-BASED PROPERTIES THAT WE USED FOR OUR SVM.

Property	Neutral Value Used
GES hydropathy index	1.38
Kyte-Doolittle index	-0.5
Solubility	65
PI	6
Polarity	8
Molecular weight	136
Alpha helix index	1

and from these predictions we generated the following features: the fraction of residues in the sequence that were predicted as α helices, β sheets and loops. These features were tested in conjunction with the average periodicity features of Table II.

C. Multiple-Instance Learning Approaches

SVMs are algorithms in the conventional machine learning model. As such, the sequence profiles as described in Section II-B must be summarized into a single set of numbers such as

the average periodicity of each property. To use the profiles directly, one must use the *multiple-instance learning model* [7], in which each example is represented as a multiset (called a *bag*) of attribute vectors rather than as a single attribute vector as in the conventional learning model. Simply put, in this new model a bag is labelled as positive (Trx-fold) if and only if the attribute vectors in it satisfy some function. For example, the algorithm of Scott et al. [21] (which is adapted from Goldman et al. [10]) looks for a set of points S such that each Trx-fold protein has a point near each point of a size-k subset $S' \subseteq S$ and that all non-Trx-fold proteins have points near at most $k-1$ points of S. E.g. this algorithm might find that all Trx-fold proteins satisfy one of the following conditions and that few non-Trx-fold proteins satisfy any of them: (1) a Kyte-Doolittle value of -4.5 around position 85 *and* a Kyte-Doolittle value of -0.75 near position 10; (2) a Kyte-Doolittle value of 3.5 near position 55 *and* Kyte-Doolittle value of 4.25 near position 92 *and* Kyte-Doolittle value of 1.5 near position 25; etc. Intuitively, Scott et al.'s algorithm searches for a set of boxes in e.g. Figure 2 that represent ranges of values of properties that are needed by a sequence for it to be Trx-fold.

We mapped our data to the multiple-instance learning model in the following way. We first found the primary sequence motif in each (positive and negative) sequence and extracted a window of size 204 around it (20 residues upstream, 180 downstream, which is a region known to contain the entire Trx fold). We then mapped all sequences to their profiles based on the 7 properties of Kim et al. [15], yielding 7-dimensional data, which we then smoothed with a Guassian kernel.

Since each 7-tuple $x_i = (x_{i1}, \ldots, x_{i7})$ in each profile is tied to a particular residue r_{x_i} in the original sequence, we need to add an 8th coordinate x_{i8} to x_i that corresponds to r_{x_i}'s position in the sequence. The simplest method is to set x_{i8} to be the index of r_{x_i} in the sequence. However, since the length of the subsequence that contains the Trx fold can vary significantly among sequences, setting x_{i8} to be the index of r_{x_i} in the sequence will likely misalign the profiles of the sequences. This can make it difficult or impossible for a learning algorithm to identify the regions of profiles that distinguish Trx-fold proteins from non-Trx-fold proteins.

A natural way around this problem is to multiply align the sequences and set x_{i8} to be the index of r_{x_i} in the multiple alignment. However, conventional multiple alignments are not feasible in our case due to low primary sequence conservation. Thus we instead used multiple alignments based on information that is conserved in the Trx-fold superfamily. We used two methods. In

the first method, we first aligned the conserved motif (typically CxxC) in all sequences. Then we used the next 180 symbols of each sequence, discarding everything else that lay beyond that point. If a sequence was not long enough to go 180 symbols past the CxxC, it was linearly rescaled so that the last symbol was in position 180. Finally, since it is also known that Trx-fold proteins extend at most 20 positions upstream of the motif, we also used these 20 positions, yielding a sequence of length at most 204, mapped to a space that spans $[1, 204]$. We then set x_{i8} to be the index of r_{x_i} in this alignment. We refer to this method as *motif-based alignment*. In our second alignment method, which we call *secondary-based alignment*, we used SAM to multiply align the secondary structure patterns (predicted by PSI-PRED) of the sequences and used residue r_{x_i}'s position in this multiple alignment to set x_{i8}.

In our final application of multiple-instance learning to this problem, we represent each sequence by its α-β *signature*. This signature models the ordering of α helices and β sheets in a given sequence, as predicted by PSI-PRED. Starting from the first residue of the sequence, we move one position to the right for each α helix seen and one position up for each β sheet. To reduce the number of points that represent a sequence, we only place a point when there is a change, e.g. from α to β. To implicitly align the sequences, we define the origin of the two-dimensional space to correspond to the first residue of the active site motif (e.g. the first C of CxxC). Figure 3 gives an example of the signature for a hypothetical sequence. Such signatures nicely fit the multiple-instance learning model and should intuitively be able to separate Trx-fold proteins from non-Trx-fold proteins due to the conserved secondary structure motif.

III. Experimental Results

A. Random Data Sets

As a first test of our techniques, we applied them on large, random data sets, constructed as follows. First we extracted 47 Trx-fold proteins from PDB [3], including thioredoxins and glutaredoxins, all containing the CxxC motif. Since these 47 had structural information, they allowed us to test our techniques when models were built on true secondary structure. We then combined these 47 proteins with a set of 226 other known Trx-fold proteins (for which secondary structure is not known) and 320 known non-Trx-fold proteins from the NCBI Non-redundant Database. We filtered our positive and negative sets to reduce similarity, yielding 183 positives and 197 negatives. We then built three HMMs: one on the sequences' primary structure, one

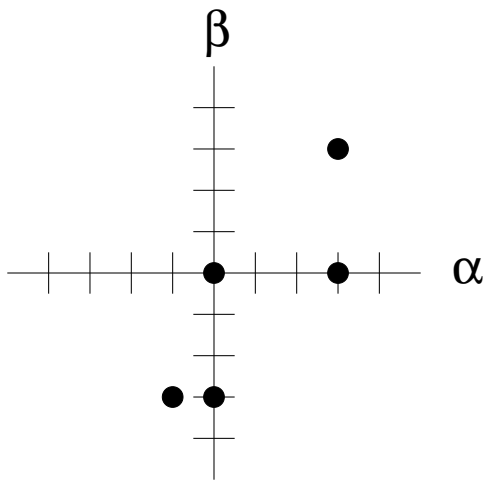

Fig. 3. The α-β signature of the hypothetical sequence $\alpha\beta\beta\beta$CxxC$\alpha\alpha\alpha\beta\beta\beta$.

on predicted secondary structure, and one on true secondary structure[2]. In all three cases, the sequences were aligned, built and calibrated in SAM. Then the test set (consisting of all Trx-fold and non-Trx-fold proteins not used for training) was searched with each model. In the secondary structure test sets, only predicted structure was used since when performing database searches, the true secondary structures would not be known.

We found that HMM trained on primary structure can achieve true positive and true negative rates of more than 0.99. This shows that HMM trained on primary structure is very effective at finding sequences so long as the model was built on other, related sequences (related in primary structure). In contrast, HMM trained on predicted secondary structures can achieve both true positive and true negative rates at about 0.82, while HMM trained on true secondary structure (but tested on predicted secondary structure) only achieved true positive and true negative rates at about 0.70. A possible explanation of True Secondary's worse performance is that errors in predicting secondary structure adds noise to the test sequences. Thus an HMM built on predicted secondary structure is also training on this noise, which might makes it less sensitive to structure prediction inaccuracies in the database.

To test models built on the QFC-based attributes, we split our filtered data set into three sets

[2]Since true secondary structure was used for one test, we used the PDB sequences for building all three models and the remaining positive and negative sequences for testing.

of approximately equal sizes and ran three tests. For each test, we trained an SVM and an MIL model using the features described in Section 2.2 on two sets and tested on the third. In this experiment, SVM averaged 0.81 for the true positive rate, and 0.85 for the true negative rate. MIL on motif based alignment averaged 0.74 for the true positive rate, and 0.88 for the true negative rate.

Since HMM on primary structure can achieve true positive and true negative rates of over 99%, it is superior to our methods in identifying new sequences that are similar to the sequences it was trained on. However, in the next section we will show that this is not the case when sequences are highly dissimilar.

B. Jack-Knife Tests

Within our data set, there are many similar sequences, which means that the experiments of Section III-A are inappropriate to evaluate our methods for the purpose they were designed: to identify new families that are highly dissimilar to known ones, i.e. identify sequences that primary sequence-based HMMs cannot. Since our goal is to identify new families, the sequences in our data set should be highly dissimilar to each other. Thus we constructed a new positive set of putative Trx-fold proteins such that primary sequence conservation between each pair of sequences was so low that SAM was unlikely to identify any one with a model built on the rest. We started by randomly selecting one sequence from a set S of 1100 putative Trx-fold sequences, placing it in our positive set P, and built an HMM M on P using SAM. We then used M to score the other 1099 putative Trx-fold sequences from $S \setminus P$ ("$\setminus$" denotes set difference, i.e. those sequences that are in S but not in P) and added to P the one with the highest E-value (i.e. the least similar one). We then iteratively built a new HMM on P, scored the remaining sequences in $S \setminus P$, and added to P the sequence with largest E-value until $|S| = 25$. We then further filtered S by building an HMM on each individual sequence and testing that model on the remaining sequences. We discarded any sequence that, when tested against a model, produced an E-value less than 0.01. Twenty sequences remained, which we then used as our set of positives (see Table III). A jack-knife test using SAM on primary structure only found one of these sequences. In these 20 sequences, 17 of them have the CxxC motif, 2 of them have the CxxS

TABLE III

THE 20 POSITIVE SEQUENCES USED IN OUR JACK-KNIFE TESTS.

accession number	motif	putative class
gi: 13400018	SxxC, CxxC	Thioredoxin
gi: 14602058	CxxC	Thioredoxin
gi: 19698793	CxxC	Thioredoxin
gi: 2194076	CxxC	Thioredoxin
gi: 7512732	CxxC	Thioredoxin
gi: 443281	CxxC	Thioredoxin
gi: 1076496	CxxC	PDI
gi: 840745	CxxC	Thioredoxin
gi: 1421133	CxxC	Gluteredoxin
gi: 129727	CxxC, CxxC	PDI
gi: 15229353	CxxS	Gluteredoxin
gi: 14787802	KxxC	PDI
gi: 14729415	CxxC, CxxC	PDI
gi: 13122603	CxxC	Gluteredoxin
gi: 11494247	CxxC	Thioredoxin
gi: 15150492	CxxC	Gluteredoxin
gi: 13358154	CxxC	Thioredoxin
gi: 24372070	CxxC	Thioredoxin
gi: 23483739	CxxS, CxxS	PDI
gi: 16763418	CxxC	PDI

motif, and the final sequence has neither CxxC nor CxxS motif[3].

Due to the small number of putative positive proteins available in our new data set, we performed a jack-knife (leave-one-out cross-validation) test. We held out one positive protein for use in testing and used the rest for training, repeating once for each of the 20 positive proteins

[3]In Table III, gi:14787802 is a putative Trx-fold protein. Its actual motif is unknown, but believed to be KxxC.

in the data set. So for each HMM-based experiment, the model was built on 19 positive proteins and the test set (the one that is searched by the model) consisted of all 20 positive proteins and all our negative proteins[4]. Since the SVM and the multiple-instance learning algorithm require both positive and negative proteins for training, we split our set of negative proteins into 8 equal-sized sets. We then trained our algorithms on the 19 positive proteins plus one of the 8 sets of negative proteins, and tested on the held-out positive protein plus the remaining 7 sets of negative proteins. We repeated this for each of the 8 sets of negative proteins. Thus we ran $20 \times 8 = 160$ experiments for each algorithm.

Our results are in Table IV. For HMM-based experiments, we used an E-value cutoff of 0.1 as in Section III-A. Since each jack-knife round for SVM and multiple-instance learning involved 8 experiments (one for each negative set), we gave the algorithm credit for correctly classifying the held-out positive protein if it successfully identified it at least half the time. The TP rates in the tables are the fractions (out of 20) of the set of positive proteins that each algorithm correctly identified. For the HMM-based algorithms, TN is the fraction of negative proteins that had E-values above 0.1. For SVM and multiple-instance learning, TN is that algorithm's accuracy on the negative proteins over all 160 experiments. The three MIL models, the SVM and the HMM built on predicted secondary structure (PSI-PRED, New Prior) were the overall best performers, correctly identifying 0.75, 0.70, 0.70, 0.50 and 0.50 positives and over 0.75, 0.76, 0.70, 0.88 and 0.81 of the negatives. We can also draw a conclusion from the table that using new priors can improve the results greatly over a naive uniform prior (i.e. the base prior we used to construct the new prior). For the SVM, the secondary structure information, the molecular weight and the Kyte-Doolittle hydropathy scale were most important in separating the positives from the negatives. The remapping schemes based on hydrophobicity, charge, volume and mass did not work well. One probable reason is that we could not get good alignments from them and so we could not build good HMMs.

Interestingly, there is little correlation among the methods we tested in terms of the positive sequences they found. Table V summarizes each algorithm's performance on each of the 20 Trx-fold proteins from the jack-knife test. An "H" in an entry indicates that the algorithm was

[4]We used the 19 training sequences in our test set so we could compare the E-values of the hold-out to those of sequences that the model was built on. However, all error rates reported are only on sequences that were not used to build the models.

TABLE IV

SUMMARY OF RESULTS ON THE JACK-KNIFE TESTS ON THE SET OF 20 TRX-FOLD PROTEINS. "TP" IS TRUE POSITIVE RATE, "TN" IS TRUE NEGATIVE RATE, "NH" IS NEAR HIT RATE (HMM E-VALUE IN $[0.1, 1)$).

Algorithm	TP	NH	TN
HMM Primary	0.05	0.00	0.98
HMM Pred. Second (PSI-PRED+Uniform Prior)	0.30	0.10	0.85
HMM Pred. Second (PSI-PRED+New Prior)	0.50	0.05	0.81
HMM Pred. Second (Predator+Uniform Prior)	0.10	0.00	0.94
HMM Pred. Second (Predator+New Prior)	0.25	0.25	0.81
SVM (QFC features+fraction of α,β,loop)-PSI-PRED	0.50	N/A	0.88
SVM (QFC fretures+fraction of α,β,loop)-Predator	0.35	N/A	0.92
MIL (Motif-based alignment)	0.75	N/A	0.75
MIL (α-β signature)	0.70	N/A	0.76
MIL (Secondary-based alignment)	0.70	N/A	0.70
Volume	0.10	0.00	0.96
Mass	0.15	0.00	0.98
Charge	0.0	0.00	0.96
Hydro-4	0.05	0.05	0.99
Hydro-6	0.15	0.00	0.99

successful in finding that protein. We see that the five proteins missed by SVM (column 6) are hit by Predicted Secondary (column 2). Since both of these algorithms have high TN rates, it suggests that taking a union of these classifiers' hits would work well. Indeed, a classifier that predicts Trx when either Predicted Secondary or SVM says "yes" would cover of the 75% positives with a true negative rate 73%. However, while combining an MIL classifier with either SVM or Predicted Secondary improves the true positive rate to 75–85%, the true negative rates drop to 59–69%. Thus it is better to either use an MIL algorithm in isolation or take the union of SVM and Predicted Secondary.

TABLE V

SUMMARY OF WHICH SEQUENCES WERE FOUND BY EACH CLASSIFIER IN THE 20-FOLD JACK-KNIFE TEST. "H" INDICATES A HIT, "M" A MISS, AND "NH" A NEAR HIT (E-VALUE IN $[0.1, 1.0)$). FOR HMM-BASED ALGORITHMS, PRIM MEANS PRIMARY STRUCTURE, PSI MEANS SECONDARY STRUCTURE IS PREDICTED BY PSI-PRED, PRE MEANS SECONDARY STRUCTURE IS PREDICTED BY PREDATOR, U MEANS USING UNIFORM PRIOR, N MEANS USING NEW PRIOR. FOR THE MULTIPLE-INSTANCE LEARNING (MIL) ALGORITHMS, MOTIF MEANS THE MODEL IS ON MOTIF-BASED ALIGNMENT, SECONDARY MEANS THE MODEL IS ON SECONDARY-BASED ALIGNMENT, α-β MEANS THE MODEL IS ON α-β SIGNATURE, HMM MEANS HIDDEN MARKOV MODEL, SVM MEANS A CLASSIFIER BUILT FROM A SUPPORT VECTOR MACHINE.

Gi Number	HMM Prim	**HMM PSI N**	HMM PSI U	HMM PRE N	HMM PRE U	**SVM PSI**	SVM PRE	**MIL Motif**	**MIL** α-β	MIL Secondary
13400018	M	M	M	M	M	M	H	H	H	M
14602058	M	NH	NH	H	M	M	M	M	H	H
19698793	M	H	H	H	M	M	M	H	H	H
2194076	M	H	H	M	M	M	M	M	H	M
7512732	M	M	H	H	H	H	H	H	H	H
443281	H	H	H	H	M	H	M	H	H	H
1076496	M	M	M	M	M	M	M	M	M	H
840745	M	H	H	NH	M	H	M	H	H	H
1421133	M	H	M	NH	M	H	M	M	H	H
129727	M	H	M	NH	M	M	H	H	H	H
15229353	M	M	M	M	M	M	M	H	H	M
14787802	M	M	M	M	M	H	M	M	M	M
14729415	M	M	M	M	M	H	M	H	M	H
13122603	M	M	M	M	M	H	H	H	H	H
11494247	M	H	M	H	M	H	M	H	H	H
15150492	M	M	M	M	M	H	H	H	H	H
13358154	M	H	H	M	M	M	M	H	M	H
24372070	M	M	M	NH	H	M	M	H	M	H
23483739	M	H	NH	M	M	M	H	H	M	M
16763418	M	H	M	NH	M	H	H	H	H	M

IV. Conclusion

We proposed numerous solutions to the problem of identifying new families within a superfamily with low primary sequence conservation such as the thioredoxin-fold superfamily. Our approaches focus on structural information rather than primary sequence. Jack-knife tests indicate that the most accurate methods are based on algorithms in the multiple instance learning (MIL) model, followed by HMMs on predicted secondary structure and support vector machines. Further, taking a union of the results from HMM-secondary and SVM yields a performance comparable to that of MIL. (The most credible hits would of course be those that are detected by multiple models.)

While some of the features used by our algorithms exploited the secondary structure motif found in the Trx-fold superfamily, we did not use the specific motif itself anywhere. Thus we believe that these techniques should be applicable to other superfamilies with low primary sequence conservation, especially when there is other conservation within the superfamily, e.g. secondary structure or conservation of specific QFC properties. Future work includes such application, especially to G protein-coupled receptors and some specific classes of oxidoreductases.

Acknowledgment

The authors thank Haifeng Ji, Peggy Wen, Kevin Karplus, Gregory Kryukov, Hasan Otu, and Yan Zhang for their help with the experiments, and Etsuko Moriyama for help with QFC. The project described was supported by NIH Grant Number RR-P20 RR17675 from the IDeA program of the National Center for Research Resources. It was also supported in part by NSF grants CCR-0092761, CCR-9877080, and EPS-0091900.

References

[1] C. M. Andorf, D. L. Dobbs, and V. G. Honavar. Discovering protein function classification rules from reduced alphabet representations of protein sequences. In *Proceedings of the Fourth Conference on Computational Biology and Genome Informatics*, pages 1200–1206, 2002.

[2] F. Aslund and J. Beckwith. The thioredoxin superfamily: redundancy, specificity, and gray-area genomics. *Journal of Bacteriology*, 181:1375–1379, 1999.

[3] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, and P.E. Bourne. The protein data bank. *Nucleic Acids Research*, 28:235–242, 2000.

[4] T. Brown. *Molecular Biology Labfax*. Academic Press, second edition, 1998.

[5] L. Debarbieux and J. Beckwith. Electron avenue: pathways of disulfide bond formation and isomerization. *Cell*, 99:117–119, 1999.

[6] G. Deleage and B. Roux. An algorithm for protein secondary structure prediction based on class prediction. *Protein Engineering*, 1:289–294, 1987.

[7] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Perez. Solving the multiple-instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1–2):31–71, 1997.

[8] D. M. Engelman, T. A. Steitz, and A. Goldman. Identifying nonpolar traasbilayer helices in amino acid sequences of membrane proteins. *Annual Review of Biophysics and Biophysical Chemistry*, 15:321–353, 1986.

[9] D. Frishman and P. Argos. Seventy-five percent accuracy in protein secondary structure prediction. *Proteins*, 27:329–335, 1997.

[10] S. A. Goldman, S. K. Kwek, and S. D. Scott. Agnostic learning of geometric patterns. *Journal of Computer and System Sciences*, 6(1):123–151, February 2001.

[11] G. Von Haijne. Membrane protein structure prediction: Hydrophobicity analysis and the positive-inside rule. *Journal of Molecular Biology*, 225:487–494, 1992.

[12] A. Holmgren. Thioredoxin and glutaredoxin systems. *Journal of Biological Chemistry*, 264(24):13963–13966, 1989.

[13] A. Holmgren and M. Bjornstedt. Thioredoxin and thioredoxin reductase. *Methods in Enzymology*, 252:199–208, 1995.

[14] R. Hughey and A.Krogh. Hidden markov models for sequence analysis: Extention and analysis of the basic method. *Computer Applications in the Biosciences*, 12(2):95–107, 1996.

[15] J. Kim, E. N. Moriyama, C. G. Warr, P. J. Clyne, and J. R. Carlson. Identification of novel multi-transmembrane proteins from genomic databases using quasi-periodic structural properties. *Bioinformatics*, 16:767–775, 2000.

[16] J. Kyte and R. F. Doolittle. A simple method for displaying the hydropathic character of a protein. *Journal of Molecular Biology*, 157:105–132, 1982.

[17] J. Martin. Thioredoxin—a fold for all reasons. *Structure*, 3:245–250, 1995.

[18] L. J. McGuffin, K. Bryson, and D. T. Jones. The psipred protein secondary structure prediction sever. *Bioinformatics*, 16:404–405, 2000.

[19] B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2002.

[20] S. D. Scott, H. Ji, P. Wen, D. E. Fomenko, and V. N. Gladyshev. On modeling protein superfamilies with low primary sequence conservation. Technical Report UNL-CSE-2003-4, Dept. of Computer Science, University of Nebraska, 2003.

[21] S. D. Scott, J. Zhang, and J. Brown. On generalized multiple-instance learning. Technical Report UNL-CSE-2003-5, Dept. of Computer Science, University of Nebraska, 2003.

[22] K. Sjölander, K. Karplus, M. Brown, R. Hughey, A. Krogh, I. S. Mian, and D. Haussler. Dirichlet mixtures: A method for improving detection of weak but significant protein sequence homology. *Computer Applications in the Biosciences*, 12(4):327–345, 1996.

[23] J. L. Tukey. *Exploratory data analysis*. Addison Wesley, 1977.

[24] X. Wang, D. Schroeder, D. L. Dobbs, and V. G. Honavar. Data-driven discovery of protein function classifiers: Decision trees based on meme motifs outperform prosite patterns and profiles on peptidase families. In *Proceedings of the Fourth Conference on Computational Biology and Genome Informatics*, pages 1193–1199, 2002.