Using Global Sequence Similarity to Enhance Biological Sequence Labeling

Cornelia Caragea
Computer Science Department
Iowa State University
cornelia@cs.iastate.edu

Jivko Sinapov
Computer Science Department
Iowa State University
jsinapov@cs.iastate.edu

Drena Dobbs
Department of Genetics and Cell Biology
Iowa State University
ddobbs@iastate.edu

Vasant Honavar
Computer Science Department
Iowa State University
honavar@cs.iastate.edu

Abstract

Identifying functionally important sites from biological sequences, formulated as a biological sequence labeling problem, has broad applications ranging from rational drug design to the analysis of metabolic and signal transduction networks. In this paper, we present an approach to biological sequence labeling that takes into account the global similarity between biological sequences. Our approach combines unsupervised and supervised learning techniques. Given a set of sequences and a similarity measure defined on pairs of sequences, we learn a mixture of experts model by using spectral clustering to learn the hierarchical structure of the model and by using bayesian approaches to combine the predictions of the experts. We evaluate our approach on two important biological sequence labeling problems: RNA-protein and DNA-protein interface prediction problems. The results of our experiments show that global sequence similarity can be exploited to improve the performance of classifiers trained to label biological sequence data.

1. Introduction

Advances in high throughput data acquisition technologies have resulted in rapid increase in the amount of data in biological sciences. For example, progress on sequencing technologies has resulted in the release of hundreds of complete genome sequences. With the exponentially growing number of biological sequences from genome projects and high-throughput experimental studies, sequence annotations do not keep pace with sequencing. The wet-lab experiments to determine the annotations (e.g., functional site annotations) are still difficult and time consuming. Hence, there is an urgent need for development of computational tools that can accurately annotate biological data.

Machine learning methods currently offer one of the most cost-effective approaches to construction of predictive models in applications where representative training data are available. The supervised learning problem [10] can be formally defined as follows: Given an *independent and identically distributed* (*iid*) dataset $\mathcal{D}$ of labeled instances $(\mathbf{x}_i, y_i)_{i=1,\cdots,n}$, $\mathbf{x}_i \in \mathbf{R}^d$ and $y_i \in Y$, where Y is the set of all possible class labels, a hypothesis class H representing the set of all possible hypotheses that can be learned, and a performance criterion P, the learning algorithm L outputs a hypothesis $h \in H$ (i.e., a classifier) that optimizes P. During classification, the task of the classifier h is to accurately assign a new instance $\mathbf{x}_{test}$ to a class label $y \in Y$.

Most biological data involve sequence data, e.g., nucleic or amino acid sequences. *Biological sequence labeling* is an example of supervised learning problem. The labeled instances $(\mathbf{x}_i, \mathbf{y}_i)_{i=1,\cdots,n}$, are pairs of input/output sequences, $\mathbf{x}_i = (x_{i,1}x_{i,2}\cdots x_{i,m})$ and $\mathbf{y}_i = (y_{i,1}y_{i,2}\cdots y_{i,m})$, where $y_{i,j}$ in the output sequence is the class label for $x_{i,j}$ in the input (or observation) sequence, $j = 1, \cdots, m$. Given a new input sequence $\mathbf{x}_{test}$, the task of the classifier $h \in H$ is to predict a class label for each element that appears at each position along the sequence.

A large volume of work has been carried out to label biological sequence data. Terribilini *et al.* [18] trained Naive Bayes classifiers to identify RNA-protein interface residues in a protein sequence. Qian and Sejnowski [16] trained Neural Networks to predict protein secondary structure, i.e., classifying each residue in a protein sequence into one of the three classes: helix (H), strand (E) or coil (C). Caragea *et al.* [5] and Kim *et al.* [12] used Support Vector Machines to identify residues in a protein sequence that undergo post-translational modifications.

Typically, to solve the biological sequence labeling problem using standard machine learning algorithms, each element in a sequence is encoded based on a local, fixed-length window corresponding to the target element and its sequence context (an equal number of its sequence neighbors on each side) [9]. The classifier is trained to label the target element. This procedure can produce reliable results, especially if we suspect that there exists a local sequence pattern around each functional site.

However, there are cases where the local amino acid distribution around functionally important sites in a given set of proteins is highly variable. For example, in identifying RNA-protein and DNA-protein interface residues from protein sequences, there is typically no consensus sequence around each site.

Machine learning classifiers designed to distinguish "positive" examples from the "negative" ones, must "learn" to do this by training on characteristics associated with known "positive" and "negative" examples. When the features that distinguish them are complex, training more specific classifiers to focus on particular subsets of the data is essential. The greater the commonality among members of a subset, the more likely it is that a machine learning approach will be successful in identifying the predictive characteristics.

Against this background, we hypothesize that classifiers trained to label biological sequence data can be improved by taking into account the global sequence similarity between the protein sequences in addition to the local features extracted around each site. The intuition behind this hypothesis is that the more similar two sequences are, the higher the correlation between their functional sites for a particular problem. Therefore, we propose to improve the biological sequence labeling problem by using a machine learning approach, that is, a mixture of experts model that considers the global similarity between protein sequences when building the model and making the predictions.

We evaluate our approach to learning a mixture of experts model on two biological sequence labeling tasks: RNA- and DNA-protein interface prediction tasks and demonstrate that taking into account global sequence similarity can improve the performance of the classifiers trained to label biological sequence data.

The rest of the paper is organized as follows: In Section 2, we review two related approaches to learning multiple models. In Section 3, we describe our approach to learning a mixture of experts model. In Section 4, we briefly introduce the machine learning algorithms applied in this study. In Section 5, we describe the data sets construction and parameter setting. In Section 6, we present experiments and results on the RNA- and DNA-protein interface prediction tasks. We conclude our study in Section 7 and highlight some directions for future work.

2. Related Work

2.1. Hierarchical Mixture of Experts

The Hierarchical Mixture of Experts model (HME) was first proposed by Jordan and Jacobs (1994) [11] to solve nonlinear classification and regression problems while learning linear models: the input space is divided into a set of nested regions and simple (e.g., linear) models are fit to the data that fall in these regions. Hence, instead of using a "hard" partitioning of the data, the authors use a "soft" partitioning, i.e., the data is allowed to simultaneously lie in more than one region.

The HME has a tree-structured architecture known *apriori*. It consists of *gating networks* that sit at the internal nodes and *expert networks* that sit at the leaf nodes of the tree. The expert networks output class distributions for each input x, while the gating networks learn how to combine the predictions of the experts up to the root of the tree which returns the final prediction. The parameters of the gating networks are learned using Expectation Maximization algorithm [6]. The gating and the expert networks are generalized linear models.

2.2. Ensemble of Classifiers

An ensemble of classifiers is a collection of independent classifiers, each classifier being trained on a subsample of the training data [8]. The prediction of the ensemble of classifiers is computed from the predictions of the individual classifiers using majority voting. An example is misclassified by the ensemble if a majority of the classifiers misclassifies it. When the errors made by the individual classifiers are uncorrelated, the predictions of the ensemble of classifiers are often more reliable.

3. Learning Mixture of Experts Models

Here we present our approach to learning a mixture of experts model that takes into account the global similarity between biological sequences. Unlike the HME model [11], we assume that the structure of our model is not known *apriori*. Hence, to learn its hierarchical structure, we use spectral clustering techniques. The leaf nodes consist of expert classifiers, while the gating nodes combine the output of each classifier to the root of the tree which makes the final prediction. The gating nodes combine the predictions of the expert classifiers based on an estimate of the cluster membership of a test protein sequence. Similar to Jordan and Jacobs [11], we considered a "soft" partitioning of the data, i.e., each sequence in the training set simultaneously lies in all clusters of the hierarchical structure with a different weight in each cluster.

The combination scheme of the predictions of the expert classifiers and the "soft" partitioning of the data that considers the global sequence similarity differentiate our model from an ensemble of classifiers model.

3.1. Learning the Structure of the Model

To learn the hierarchical structure of our model, we use hierarchical clustering, an unsupervised learning technique [10] that attempts to uncover the hidden structure that exists in the unlabeled data. Given a data set $\mathcal{D}$ of unlabeled protein sequences $(\mathbf{x}_i)_{i=1,\ldots,n}$, and a similarity measure S defined on pairs of sequences, the clustering algorithm C partitions the data into dissimilar clusters of similar sequences producing a tree-structured architecture (see Figure 1).

We first compute the pairwise similarity matrix $\mathbf{W}_{n\times n}$ for the protein sequences in the training set based on a common global sequence alignment method. Second, using this similarity matrix, we apply 2-way spectral clustering algorithm, described in the next subsection, to recursively bipartition the training set of protein sequences until a splitting criterion is met.

The output of the algorithm is a hierarchical clustering of the protein sequences, i.e., a tree $\mathcal{T}$ such that each node (cluster) consists of a subset of sequences. The root node is the largest cluster containing all the protein sequences in the training set. Once a cluster is partitioned into its two subclusters, it becomes their parent in the resulting tree structure. We store all the intermediate clusters computed by the algorithm. If the number of sequences at a given cluster falls below some percentage of the total sequences in the training set, then the node becomes a leaf and thus is not further partitioned (we used 10% in our experiments).

Figure 1 shows the tree structure produced by the 2-way spectral clustering algorithm when applied to a set of 147 RNA-protein sequences. The similarity matrix is computed based on the Needleman-Wunsch global alignment algorithm. In the figure, to keep the tree smaller, we stopped bipartitioning a node when the number of sequences at a given cluster falls below 30% of the total sequences in the training set.

3.2. 2-Way Spectral Clustering

Spectral clustering has been successfully applied in many domains, including image segmentation [17], document clustering [7], grouping related proteins according to their structural SCOP classification [15].

Spectral clustering falls within the category of graph partitioning algorithms that partition the data into disjoint clusters by exploiting the eigenstructure of a similarity matrix. In general, to find an optimal graph partitioning is NP complete. Shi and Malik [17] proposed an approximate spectral clustering algorithm that optimizes the *normalized cut* (NCut) objective function. It is a divisive, hierarchical clustering algorithm that recursively bi-partitions the graph until some criterion is reached, producing a tree structure.

Let $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$ be the set of sequences to be partitioned and let S be a similarity function between pairs of sequences. The 2-way spectral clustering algorithm consists of the following steps:

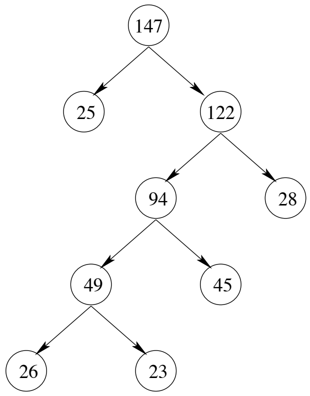


Figure 1. The resulting hierarchical structure produced by spectral clustering when applied to a set of 147 RNA-protein sequences. The number in each node indicates the number of protein sequences belonging to it. The Needleman-Wunch global alignment score was used as a pairwise similarity measure during the clustering process.

1. Let $\mathbf{W}_{n\times n} = [S(i,j)]$ be the symmetrical matrix containing the similarity score for each pair of sequences.

2. Let $\mathbf{D}_{n\times n}$ be the degree matrix of $\mathbf{W}$, i.e., a diagonal matrix such that $\mathbf{D}_{ii} = \sum_j S(i,j)$.

3. Solve the eigenvalue system $(\mathbf{D} - \mathbf{W})x = \lambda \mathbf{D} x$ for the eigenvector corresponding to the second smallest eigenvalue and use it to bipartition the graph.

4. Recursively bipartition each subgraph obtained at Step 3. if necessary.

Note that the quality of the clusters found by the 2-way spectral clustering algorithm depends heavily on the choice of the similarity function S.

3.3. Estimating the Parameters of the Model

Following the approach taken by Jordan and Jacobs [11], we make use of the "soft" partitioning of the biological sequence data. Thus, having the hierarchical clustering $\mathcal{T}$ stored, we devise a procedure that allows each sequence in the training set to simultaneously lie in all clusters, with a different weigth in each cluster.

For each sequence $\mathbf{x}_i$, $i = 1, \cdots, n$ in the training set $\mathcal{D}$, we compute its cluster membership as follows (Figure 2):

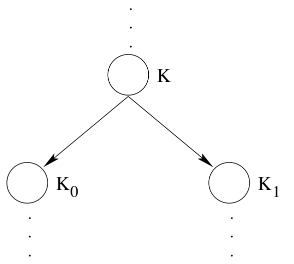


Figure 2. Estimating cluster membership of a sequence.

1. Find the K closest sequences to $\mathbf{x}_i$ at the parent node based on the similarity function used to construct the hierarchical clustering $\mathcal{T}$ (in our experiments we used K equal to 20% of the sequences at the parent node).

2. Let K_0 out of K sequences go to the left child node, and K_1 out of K go to the right child node.

3. The estimated probability of $\mathbf{x}_i$ for being in child node j is computed as $p(\mathbf{x}_i \in V_j|\mathbf{x}_i \in par(V_j)) = K_j/K$, where $j = 0, 1$.

We recursively place the sequence $\mathbf{x}_i$ in all the nodes of $\mathcal{T}$ with different weights, starting from the root, based on its estimated cluster membership computed above. Thus, the sequence weight at the root is 1 (all the sequences in the training set lie at the root of the tree), and the weight at any other node in the tree is the product of the sequence weights on the path from the root to that node.

Let $V^l_1, V^l_2, \cdots, V^l_M$ be the leaf nodes and $V^g_1, V^g_2, \cdots, V^g_N$ be the internal or gating nodes in the hierarchical clustering $\mathcal{T}$. During learning, we train either a collection of M Naïve Bayes classifiers or a collection of M Logistic Regression classifiers, one classifier at each leaf node V^l_k, $k = 1, \cdots, M$. Naïve Bayes and Logistic Regression are briefly described in the next section.

To solve the *biological sequence labeling problem*, one approach is to predict each element $x_{i,j}$ in the sequence $\mathbf{x}_i$ independently, i.e., to assume that the observation-label pairs $(x_{i,j}, y_{i,j})_{j=1,m}$ are independent of each other (the *label independence assumption*). However, $x_{i,j}$ may not contain all the information necessary to predict $y_{i,j}$. Hence, it is fairly common to encode each element $x_{i,j}$ in the sequence $\mathbf{x}_i$ based on a local, fixed-length window corresponding to the target element and its sequence context (an equal number of its sequence neighbors on each side) $x'_{i,j} = x_{i,j-t}, \cdots, x_{i,j}, \cdots, x_{i,j+t}$. The classifier is trained to label the target element $x_{i,j}$ [9].

During classification, given a test sequence $\mathbf{x}_{test}$, we extract the local windows corresponding to its elements. Each classifier at the leaf nodes V^l_k returns the class membership for each window in the test sequence,

$$p_{V^l_k}(y_{test,j} = y|x'_{test,j}, \mathbf{x}_{test}), \text{ for all } y \in Y$$

The gating nodes V^g_k, $k = 1, \cdots, N$ in the hierarchical clustering $\mathcal{T}$ combine the predictions of the classifiers to the root node that makes the final prediction. Thus, each gating node combines the predictions from its child nodes (which can be leaf nodes or descendent gating nodes) using the formula:

$$p_{V^g_k}(y|x'_{test,j}, \mathbf{x}_{test}) = \sum_{V_i \in child(V^g_k)} p_{V_i}(y|x'_{test,j}, \mathbf{x}_{test}) p_{V_i}(\mathbf{x}_{test} \in V_i|\mathbf{x}_{test} \in V^g_k)$$

Finally, the window $x'_{test,j}$ is assigned to the class y that maximizes the posterior probability from the root gating node, V_{root}:

$$y = \arg\max_{y \in Y} p_{V_{root}}(y|x'_{test,j}, \mathbf{x}_{test})$$

4. Machine Learning Classifiers

4.1. Naïve Bayes

Naïve Bayes (NB) [13] is a supervised learning algorithm that belongs to the class of generative models, in which the probabilities $p(\mathbf{x}|y)$ and $p(y)$ of the input $\mathbf{x}$ and the class label y are estimated from the training data using maximum likelyhood estimates. Typically, the input $\mathbf{x}$ is high-dimensional, represented as a set of features (attributes), $\mathbf{x} = (x_1, x_2, \cdots, x_d)$, making it impossible to estimate $p(\mathbf{x}|y)$ for large values of d. However, the Naïve Bayes classifier makes the assumption that the features are conditionally independent given the class:

$$p(x_1, x_2, \ldots, x_d|y) = \prod_{i=1}^{d} p(x_i|y)$$

Therefore, training a Naïve Bayes classifier reduces to estimating probabilities $p(x_i|y)$, $i = 1, \cdots, d$, and $p(y)$, from the training data, for all class labels y.

During classification, Bayes Rule is applied to compute $p(y|\mathbf{x}_{test})$:

$$p(y|\mathbf{x}_{test}) = \frac{p(\mathbf{x}_{test}|y)p(y)}{p(\mathbf{x}_{test})}$$

The class label with the highest posterior probability is assigned to the new input $\mathbf{x}_{test}$.

4.2. Logistic Regression

Logistic Regression (LR) [14] is a supervised learning algorithm that belongs to the class of discriminative models. Here, we consider the case of binary classification, where the set of class labels $Y = \{0, 1\}$. Logistic Regression directly calculates the posterior probability $p(y|\mathbf{x})$ and makes the predictions by threshoding $p(y|\mathbf{x})$. It does not make any assumptions regarding the conditional independence of the features and models the conditional probability of the class label y given the input $\mathbf{x}$ as follows:

$$p(y = 1|\mathbf{x}; \beta, \theta) = \frac{1}{1 + e^{(-\beta^T \mathbf{x} - \theta)}}$$

where $[\beta, \theta]$ are the parameters of the model that can be estimated either by maximizing the conditional likelihood on the training data or by minimizing the loss function.

During classification, Logistic Regression predicts a new input $\mathbf{x}_{test}$ as 1 if and only if

$$\beta^T \mathbf{x}_{test} + \theta > 0$$

5. Data Sets and Parameter Settings

We used two datasets to perform experiments: **RNA-protein and DNA-protein interface** data sets. RNA- and DNA-protein interactions play a pivotal role in protein function. Reliable identification of such interaction sites from protein sequences has broad applications ranging from rational drug design to the analysis of metabolic and signal transduction networks.

The RNA- and DNA-protein interface data sets consist of RNA- and DNA-binding protein sequences, respectively, extracted from structures in the Protein Data Bank (PDB) [3]. We downloaded all the protein structures of known RNA- and DNA-protein complexes from PDB solved by X-ray crystallography and having X-ray resolution between 0 and 3.5Å. As of May 2008, the number of RNA-protein complexes was 435 and DNA-protein complexes was 1259. A residue was identified as interface residue using Entangle with the default parameters [1].

Furthermore, to remove redundancy in each data set, we used BlastClust, a toolkit that clusters sequences with statistically significant matches, available at http://toolkit.tuebingen.mpg.de/blastclust. In constructing our non-redundant sequence data sets, we applied various identity cutoffs, starting from 30% and ending at 90% in steps of 10. For example, in the 30% identity cutoff sequence data set, two sequences were pairwise matched if they were 30% or more identical over an area covering 90% of the length of each sequence. We randomly selected a sequence from each cluster returned by BlastClust. Thus, the resulting non-redundant RNA-protein sequence data set for 30% identity cutoff has 180 protein sequences. The total number of amino acid residues is 33,235.

We represented residues identified as interface residues in a protein sequence as positive instances (+) and those not identified as interface residues as negative instances (-). As mentioned before, we encoded each residue by a local window of fixed length, `winLength` $= 21$, corresponding to the target residue and ten neighboring residues on each side.

Data Sets	Number of Sequences	Number of + Instances	Number of - Instances
RNA-prot 30%	180	5398	27837
RNA-prot 60%	215	6689	32073
RNA-prot 90%	246	7798	34675
DNA-prot 30%	257	5326	53494
DNA-prot 60%	289	5974	58031
DNA-prot 90%	317	6551	60877

Table 1. Number of sequences as well as number of positive (+) and negative (-) instances in the non-redundant RNA- and DNA-protein sequence data sets for 30%, 60%, and 90% identity cutoffs.

Table 1 shows the number of sequences as well as the number of positive (+) and negative (-) instances in the non-redundant RNA- and DNA-protein sequence data sets for 30%, 60%, and 90% identity cutoffs.

Interesting to note is that very many sequences in both RNA- and DNA-protein interface data sets are 90% or more identical over an area covering 90% of the length of each sequence and are removed from the data sets, e.g., in the DNA-protein interface data set, the number of sequences reduces from 1259 to 317 sequences in the 90% identity cutoff data set. On the other hand, the difference in the number of sequences in the non-redundant datasets is very small (Table 1).

6. Experiments and Results

6.1. Performance Evaluation

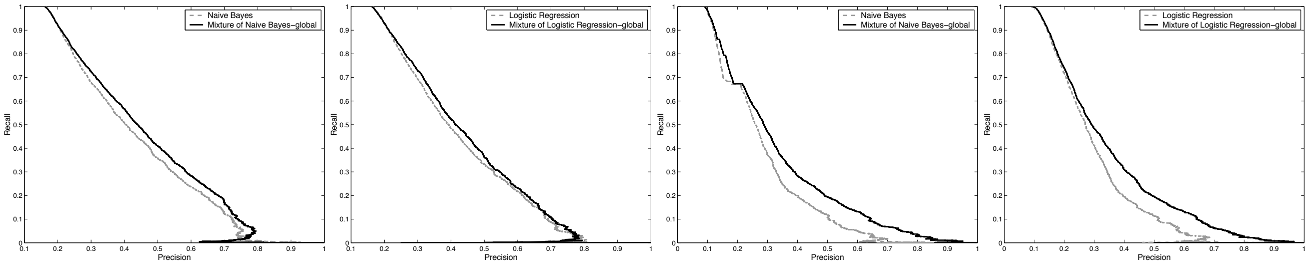
To assess the performance of classifiers in this study, we report the following measures: Precision, Recall, Correlation Coefficient (CC), and F-Measure (FM). If we denote true positives, false negatives, false positives, and true negatives by TP, FN, FP, and TN respectively, then these measures can be defined as follows:

$$\texttt{Precision} = \frac{TP}{TP + FP}$$

$$\texttt{Recall} = \frac{TP}{TP + FN}$$

$$\texttt{CC} = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FN)(TP + FP)(TN + FP)(TN + FN)}}$$

$$\texttt{FM} = \frac{2 \times \texttt{Precision} \times \texttt{Recall}}{\texttt{Precision} + \texttt{Recall}}$$



a) RNA-Protein Interface

b) DNA-Protein Interface

Figure 3. Precision-Recall curves for Naive Bayes and Mixture of Naive Bayes models as well as Logistic Regression and Mixture of Logistic Regression models on the non-redundant RNA- and DNA-protein sequence data sets at 30% identity cutoff. The hierarchical structures of the mixture of experts models are constructed based on global sequence similarity.

To obtain the estimates for TP, FN, FP and TN, we performed 10-fold sequence-based cross-validation [4] wherein the set of sequences is partitioned into 10 disjoint subsets (folds). At each run of a cross-validation experiment, 9 subsets are used for training and the remaining one is used for testing the classifier. The values for TP, FN, FP and TN are obtained using the default threshold $\theta = 0.5$, i.e., an instance is classified as positive if the probability of being in the positive class returned by the classifier is greater than or equal to 0.5, and as negative otherwise.

With any classifier, it is possible to tradeoff the Recall against Precision. Hence, it is more informative to compare the Precision-Recall curves which show the tradeoff over their entire range of possible values than to compare the performance of the classifiers for a particular choice of the tradeoff.

To evaluate how good a classifier is at discriminating between the positive and negative examples, we also report the Area Under the Receiver Operating Characteristic Curve (AUC) on the test set, which represents the probability of correct classification [2].

6.2. Experimental Design and Results

The goal of this study is to evaluate whether the performance of classifiers trained to label biological sequence data can be improved by taking into account global sequence similarity between the protein sequences in the data set in addition to the local features extracted around each residue. For both RNA- and DNA-protein interface prediction tasks, we compared two standard machine learning models, Naïve Bayes (NB) and Logistic Regression (LR), with mixture of experts models that have a hierarchical structure constructed using 2-way spectral clustering based on various similarity functions. The mixture of experts models consist of NB and LR models at the leaves, respectively. Our implementation is built on Weka, an open source machine learning software available at http://www.cs.waikato.ac.nz/ml/weka/.

In our first set of experiments, we computed the entries in the similarity matrix $\mathbf{W}$ by applying the Needleman-Wunsch global alignment algorithm on each pair of sequences. The Blosum62 substitution matrix was used for costs. The resulting entries in the matrix $\mathbf{W}$ are normalized and scaled so that each value is between 0 and 1.

In Figure 3 we compare the Precision-Recall curves for Naïve Bayes and mixture of Naïve Bayes models as well as Logistic Regression and mixture of Logistic Regression models on both RNA- and DNA-protein interface prediction tasks, where the hierarchical structure of the mixture of experts models is constructed by taking into account global sequence similarity. As illustrated in the figure, for both prediction tasks, the Precision-Recall curves for the mixture of experts models dominate the Precision-Recall curves of NB and LR models, that is, for any choice of Precision, the mixture of experts models offer a higher Recall than NB and LR. While this is true for any identity cutoff for both RNA- and DNA-protein sequence data sets, in Figure 3 we choose to show results only for 30% identity cutoff due to space constraints. The curves demonstrate that even for a very stringent cutoff, the mixture of experts that captures global similarity between sequences in the data set outperform the other models.

In Table 2, we also show the classification results after evaluating the baseline models, NB and LR, and the mixture of experts models with NB and LR at the leaves, ME-NB-global and ME-LR-global, respectively, on the RNA- and DNA-protein sequence data sets for two identity cutoffs: 30% and 90%. The values in the tables are obtained using the default threshold $\theta = 0.5$. Again, it can be seen that the mixture of experts models that capture the global sequence similarity outperform the baseline models.

In our second set of experiments, to verify that indeed global sequence similarity is instrumental in improving the performance of classifiers, and that the improvement does not come from the more sophisticated structure of the model, we computed the entries in the similarity ma-

	RNA-protein 30%					RNA-protein 90%				
Classifier	Precision	Recall	CC	FM	AUC	Precision	Recall	CC	FM	AUC
NB	0.58	0.25	0.31	0.35	0.75	0.58	0.30	0.33	0.40	0.77
ME-NB-global	0.61	**0.27**	**0.34**	**0.38**	**0.77**	**0.61**	**0.32**	**0.36**	**0.42**	**0.78**
ME-NB-local	**0.62**	0.25	0.33	0.35	0.76	**0.61**	0.30	0.34	0.40	0.77
ME-NB-random	0.59	0.24	0.31	0.35	0.75	0.59	0.30	0.33	0.40	0.77
LR	**0.62**	0.18	0.28	0.29	0.76	**0.63**	0.23	0.31	0.34	0.77
ME-LR-global	0.60	**0.23**	**0.31**	**0.34**	**0.77**	0.61	**0.27**	**0.33**	**0.38**	**0.78**

Classifier	DNA-protein 30%					DNA-protein 90%				
	Precision	Recall	CC	FM	AUC	Precision	Recall	CC	FM	AUC
NB	0.59	0.05	0.16	0.10	0.75	0.56	0.07	0.18	0.13	0.75
ME-NB-global	0.62	**0.12**	**0.25**	**0.20**	**0.77**	**0.65**	**0.15**	**0.29**	**0.25**	**0.78**
ME-NB-local	**0.65**	0.06	0.18	0.12	0.76	0.64	0.08	0.21	0.15	0.76
ME-NB-random	0.58	0.05	0.15	0.09	0.75	0.56	0.07	0.18	0.13	0.75
LR	0.57	0.07	0.18	0.12	0.79	0.57	0.08	0.18	0.14	0.79
ME-LR-global	**0.57**	**0.14**	**0.26**	**0.23**	**0.80**	**0.63**	**0.17**	**0.29**	**0.26**	**0.81**

Table 2. Experimental results with Naive Bayes (NB) and Logistic Regression (LR) models, and Mixture of Experts (ME) models on the non-redundant RNA- and DNA-protein sequence data sets, where the identity cutoffs are 30% and 90%. The results are shown for default threshold $\theta = 0.5$. ME-NB-global and ME-LR-global use NB and LR at the leaves and exploits the global sequence similarity to construct the hierarchical structure. ME-NB-local exploits the local sequence similarity to construct the hierarchical structure. ME-NB-random randomizes the global similarity matrix and constructs the hierarchical structure based on the randomized matrix.

trix $\mathbf{W}$ by applying Smith-Waterman local alignment algorithm with Blosum62, thus taking into account local sequence similarity (the matrix $\mathbf{W}$ is normalized and scaled as before). We also randomize the global similarity matrix computed previously and use this randomized matrix to construct the hierarchical structure of the mixture of experts models. The model based on the randomized matrix is similar to an ensemble of classifiers (see Section 2).

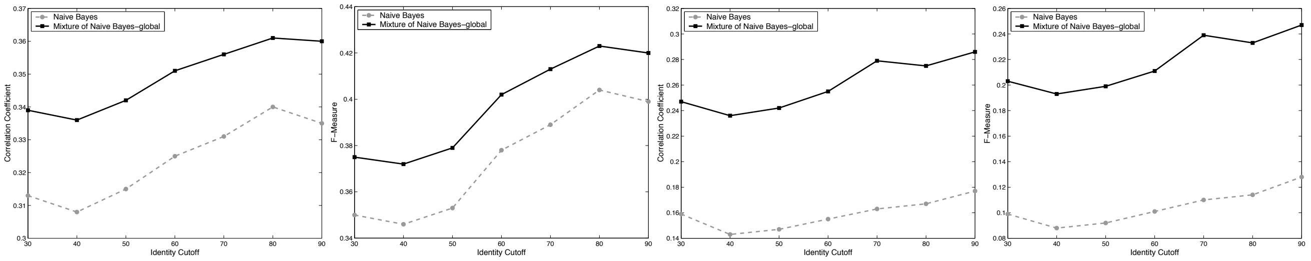
In Table 2 we compare the performance of Naïve Bayes (NB) and mixture of Naïve Bayes models using global (ME-NB-global) and local (ME-NB-local) sequence similarities, as well as a random (ME-NB-random) sequence similarity for the default threshold $\theta = 0.5$. The results of our experiments show that the mixture of experts models that capture global sequence similarity outperform the other models in terms of a majority of standard measures for comparing the performance of classifiers (the results are similar for the mixture of Logistic Regression models, data not shown). For example, for 30% identity cutoff, Correlation Coefficient increases from 0.33 (local similarity) to 0.34 (global similarity) on the RNA-protein data set, and from 0.18 (local similarity) to 0.25 (global similarity) on the DNA-protein data set. Hence, this and the previous results demonstrate that global similarity is instrumental in improving the performance of classifiers trained to label biological sequence data.

In our third set of experiments, we evaluated the effect of the identity cutoff to construct the non-redundant data sets on the Correlation Coefficient and F-Measure. Thus, we started from 30% and ended at 90% identity cutoff and recorded the values of Correlation Coefficient and F-Measure for NB and mixture of NB that capture global sequence similarity (Figure 4). Interesting to note is that even at 30% identity cutoff, a very stringent cutoff, the difference in the Correlation Coefficient and F-Measure is significant, for both RNA- and DNA-protein data sets, showing that the mixture of experts models that capture global sequence similarity indeed improve the performance of classifiers trained to label biological sequence data.

7. Discussion and Conclusions

Analyzing newly discovered proteins and detecting functionally important sites in protein sequences has broad applications in biology, e.g., rational drug design. Computational tools to do that are of particular importance because protein structures for newly sequenced proteins are usually unavailable in the public domains.

An approach is to exploit the idea that the more similar two sequences are, the higher the correlation between their functional sites. Hence, when two sequences are *highly sim-*



a) RNA-Protein Interface

b) DNA-Protein Interface

Figure 4. Comparison of Correlation Coefficient and F-Measure for Naive Bayes and Mixture of Naive Bayes models that capture global sequence similarity on non-redundant RNA- and DNA-protein data sets constructed using various identity cutoffs, starting from 30% and ending at 90% in steps of 10.

ilar, the predictions of their functional sites become trivial using homology modeling, i.e., sequence alignment. However, this approach fails to identify functional sites if the sequences are *non-homologous*, as is the case with our non-redundant datasets. Therefore, it is valuable to develop prediction methods that can be successfully applied to *non-redundant sequence data sets*. Standard machine learning classifiers were trained to label biological sequence data using local features around each residue in a sequence.

In this work we sought to improve the performance of classifiers that make predictions on residues in protein sequences by taking into account the global similarity between the protein sequences in the data set in addition to the local features around each residue. We evaluated mixture of experts models that consider the global similarity between protein sequences when building the model and making the predictions on the RNA- and DNA-protein interface prediction tasks. The results of our experiments show that indeed global sequence similarity can be exploited to improve the performance of classifiers trained to label biological sequence data.

As the quality of the clustering obtained using spectral clustering depends heavily on the similarity function, future work will include further analysis of other various similarity functions.

References

[1] J. Allers and Y. Shamoo. Structure-based analysis of protein-rna interactions using the program entangle. *J mol Biol*, 311:75–86, 2001.

[2] P. Baldi, S. Brunak, Y. Chauvin, C. Andersen, and H. Nielsen. Assessing the accuracy of prediction algorithms for classification: an overview. *Bioinformatics*, 16(5):412–424, 2000.

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

[4] C. Caragea, J. Sinapov, D. Dobbs, and V. Honavar. Assessing the performance of macromolecular sequence classifiers. In *IEEE 7th International Symposium on Bioinformatics and Bioengineering*, pages 320–326, 2007.

[5] C. Caragea, J. Sinapov, A. Silvescu, D. Dobbs, and V. Honavar. Glycosylation site prediction using ensembles of support vector machine classifiers. *BMC Bioinformatics*, 8(438), 2007.

[6] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society*, 39:1–38, 1977.

[7] I. S. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. In *Proceedings of SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 269–274, 2001.

[8] T. G. Diettrich. Ensemble methods in machine learning. *Lecture Notes in Computer Science*, 1857:1–15, 2000.

[9] T. G. Diettrich. Machine learning for sequential data: A review. In *Proceedings Joint IAPR International Workshop on Structural, Syntactic, and Statistical Pattern Recognition*, pages 15–30, 2002.

[10] R. Duda, E. Hart, and D. Stork. *Pattern Classification*. Second Edition, Wiley, 2001.

[11] M. I. Jordan and R. A. Jacobs. Hierarchical mixtures of experts and the em algorithm. *Neural Computation*, 6:181–214, 1994.

[12] J. H. Kim, J. Lee, B. Oh, K. Kimm, and I. Koh. Prediction of phosphorylation sites using SVMs. *Bioinformatics*, 20(17):3179–3184, 2004.

[13] T. M. Mitchell. *Machine Learning*. McGraw Hill, 1997.

[14] A. Y. Ng and M. I. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Advances in Neural Information Processing Systems (NIPS)*. NIPS, 2002.

[15] A. Paccanaro, J. A. Casbon, and M. A. S. Saqi. Spectral clustering of protein sequences. *Nucleic Acids Research*, 34(5):1571–1580, 2006.

[16] N. Qian and T. Sejnowski. Predicting the secondary structure of globular proteins using neural networks models. *J. Molecular Biology*, 202:865–884, 1988.

[17] J. Shi and J. Malik. Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[18] M. Terribilini, J.-H. Lee, C. Yan, R. L. Jernigan, V. Honavar, and D. Dobbs. Predicting rna-binding sites from amino acid sequence. *RNA Journal*, In Press, 2006.