Gene family classification using a semi-supervised learning method

Nan Song
Advisors: John Lafferty, Dannie Durand

January 17, 2007

Abstract

A crucial task in the completion of genomic sequencing projects is annotation of predicted genes. Annotation is facilitated by classification of sequences into families. Since members of a family tend to have similar properties, attributes of well studied genes can be used to predict the properties of newly discovered genes. Typically, only a small number of genes have been studied experimentally. The functions of the majority of genes are unknown. Thus, gene family classification can be cast as a learning problem with only a small number of labeled data. This scenario is a natural application for semi-supervised learning algorithms that extract information not only from labeled data, but also from the structure of unlabeled data.

In this study, I applied a graph-based semi-supervised algorithm developed by Zhu and Lafferty to gene family classification and compared it with supervised learning using an analogous kernel function. A novel similarity score was used that avoids false positives due to insertion of similar sequence fragments into unrelated sequences. The performance of both methods was evaluated using 18 hand curated families. In addition to overall performance comparison, the importance of parameter selection for different families was investigated. My results show that semi-supervised learning outperformed supervised learning for one third of the families, and performed equally well in the rest families. The effect is especially significant when the number of labeled family members is small.

1 Introduction

A *gene family* is a set genes that arose from the same origin. Since genes in the same family share related functions, gene family classification is a powerful tool to infer evolutionary, functional and structural properties of genes. Gene family classification has been used as the basis for high throughput gene annotation in many genome projects. Gene family classification is also useful in selecting targets for experimental study. For experimental biologists, top ranked predictions are especially of interest.

Various unsupervised clustering methods have been developed to group proteins into different families, including hierarchical clustering [24, 41, 55], spectral clustering [36], and Markov clustering [11], and GENERAGE[12]. Although much progress has been made in this area, current methods are still error prone because mutation results in many false positives and false negatives. However, in many cases, some labeled data is available that could be used to improve classification

performance. For example, when a genome project is completed, although a large number of new gene predictions need to be verified, some of these genes have already been studied in depth by independent laboratories and this information can be used to improve accuracy of classification.

Semi-supervised learning is a natural approach to the scenario where a small number of genes have been studied experimentally and are well characterized. We would like to predict which genes are in the same family as those known genes. Unlike supervised classification methods, semi-supervised methods extract information not only from labeled data, but also from the unlabeled data, by exploiting the graph structure. Thus, semi-supervised learning is especially suitable for applications in which labeled data is difficult to obtain. Various semi-supervised learning methods, including generative mixture models [8, 14, 35], transductive SVMs [20], co-training [5], information regularization [48] and graph-based models [3, 4, 48, 50, 56], have been developed. Graphical semi-supervised learning methods are a natural choice for protein classification because the protein universe is easily modeled as a similarity graph, where the vertices are the proteins and the weights of edges correspond to the degree of sequence similarity between pairs of protein sequences.

In this paper, I applied the graph-based semi-supervised learning algorithms designed by Zhu and Lafferty [56] to the problem of protein classification. The problem of identifying sequences that are in the same family as the known sequences can be formally framed as a binary classification problem. We are given as input a sequence similarity graph, where the vertices represent protein sequences. Two vertices are connected by an edge if the corresponding sequences are similar. We are also given set of labeled data. These can include known family members (positive examples) and sequences known not to belong to the family (negative examples). The goal is to separate the set of protein sequences into two groups: one is composed of family members and the other one is composed of the remaining proteins.

I used an empirical approach to study the effectiveness of this semi-supervised learning method in classifying protein families. I applied the method to a test set composed of eighteen well studied families using amino acid sequences from mouse. The effectiveness of the method was evaluated using AUC scores and the number of false negatives. I also investigated the stability of the method by testing how strongly the performance depends on parameter choices. The semi-supervised learning algorithm I used exploits graph structure. I compared its performance with that of a comparable supervised classification method to understand how much information we obtain from the structure of the graph.

The reminder of the paper is organized as follows. In section 2, I present a brief overview the graph-based semi-supervised learning algorithm for binary classification. I review current knowledge of the evolutionary process of proteins in section 3 and explain how these processes challenge protein classification. In section 4, I present the methodology used in my empirical study of the semi-supervised learning method in protein family classification. The results are summarized in Section 5, along with detailed discussion of four typical families to highlight the properties of the semi-supervised learning method in protein family classification.

2 Overview of the graph-based semi-supervised learning algorithm

In this section, I review Zhu and Lafferty's algorithm for binary classification [56]. The algorithm takes a weighted similarity graph, G = (V, E), as input. The vertices represent elements in the dataset. The edge weights are proportional to the similarity between data points; *i.e.*, edges between similar data points have large weights. The vertices include labeled data (L) and unlabeled data(U). Usually $L \ll U$. As shown in Table 1, $L = \{(x_1, y_1), (x_2, y_2), ..., (x_l, y_l)\}$ is used to represent the labeled dataset. If x_i is a positive example, then $y_i = 1$. If x_i is a negative example, then $y_i = 0$. $U = \{x_{l+1}, ..., x_{l+u}\}$ is used to represent unlabeled data. The goal of the semi-supervised learning algorithm is to assign a real value $f(x_i)$ to every vertex x_i in the unlabeled data set, such that family members have higher values than nonfamily members. Then a cutoff can be used to separate family members from nonfamily members in the unlabeled dataset.

$W_{i,j}$	Edge weight between vertices x_i and x_j
$S_{i,j}$	Similarity between vertices x_i and x_j
σ	Scaling factor in the smoothing function
L	The set of labeled data
U	The set of unlabeled data
(x_i, y_i)	Labeled data points $y_i = 1$, if x_i is a family member $y_i = 0$, otherwise
$(x_i, f(x_i))$	Unlabeled data points $f(x_i)$: real value score assigned to x_i
LF	Number of labeled family members
LN	Number of labeled nonfamily members

Table 1: Notation used in this paper

This graph-based semi-supervised method is based on the assumption that if two data points are close to each other, they are likely to be in the same class. The method can be formulated as label propagation. The labeled data act as source to send out signals to adjacent data points, and further propagate through the graph. The unlabeled data receive these signals. The estimation of $f(x_i)$ is determined by the information they received. The flow of label propagation is influenced by the edge weights. Instead of using raw similarity scores, various smoothing functions have been used to assign weights to edges. In this paper, I used the exponential smoothing function

$$W(i,j) = exp(\frac{S(i,j) - 1}{\sigma}), \tag{1}$$

where $W(i,j)$ is the weight of the edge between x_i and x_j, and $S(i,j)$ is the raw similarity score. The hyperparameter, σ, is critical in determining the relative importance of close and distance data points. Therefore, its value can affect the performance of the algorithm. I used a similarity score, $S(i,j)$, that ranges from zero to one. Since the power of the exponential in Equation (1) is $(1 - S(i,j))$, the range of $W(i,j)$ is also from zero to one. The limited range of $W(i,j)$ facilitates the study of the impact of σ, as discussed further in Sections 4 and 5.

The graph-based semi-supervised learning algorithm assigns the scores, $f(x_i)$, to minimize the following formula:

$$1/2 * \sum_{j \in D} \sum_{i \in D} W(x_i, x_j)(y_i - f(x_j))^2, \tag{2}$$

with the constraint that $f(x_i) = y_i$, when $x_i \in L$. The optimization of this formula is equivalent to solving the following equation:

$$f_u = (D_{uu} - W_{uu})^{-1} W_{ul} f_l \tag{3}$$

where $D = diag(d_i)$ is the diagonal matrix with $d_i = \sum_j W_{ij}$.

Compared to the supervised method, the advantage of the semi-supervised method is that it derives information from the graph structure. To understand how much information is obtained from the graph, I compare the performance of graph based semi-supervised classification with the corresponding supervised classification method. That algorithm selects values of $f(x_i)$ that minimize

$$1/2 * \sum_{j \in D} \sum_{i \in L} W(x_i, x_j)(y_i - f(x_j))^2. \tag{4}$$

Notice, although Equation (2) and Equation (4) are similar, Equation (2) considers the distances among all pairs of data points. In contrast, Equation (4) only considers the distances among pairs where one element is labeled and the other is unlabeled.

3 The sequence similarity graph

In this work, the protein universe is modeled as a graph $G = (V, E)$, where V is the set of all amino acid sequences and two vertices are connected by an edge if the associated sequences have significant similarity. The structure of this graph is influenced by the processes of molecular evolution.

Proteins evolve through duplication, mutation and domain shuffling. Duplication is the major force to increase gene family size. Mutations change protein sequences, leading to functional differentiation. Domains are sequence fragments that can fold independent of context and act as structural modules. These sequence fragments can be duplicated and transferred to different proteins in the evolutionary process called *domain shuffling*. Since domains are structural, evolutionary and functional units in proteins, domain shuffling enables rapid evolution of new functions for proteins.

Ideally, all family members form a single clique, and there are no edges connecting this clique to vertices from other families. In this case, the task of identifying gene families is simply to find the clique. However, because the complex evolutionary history of protein families, neither of these ideals is generally true. The accumulation of mutations can reduce the similarity between genes in the same family to the point where it is not recognizable. As a result, genes in the same family may not form a clique. In some cases, they may not even form a connected component. On the other hand, mutations may cause genes in different families to have significant sequence similarity. Domain shuffling can also result in significant sequence similarity between genes in different families. As a result, proteins from different families can be connected by edges in the similarity graph.

In summary, evolutionary processes can result in both missing edges and wrong edges in the sequence similarity graph. This deformation of the graph is the major challenge to graph-based protein classification methods. In the following sections, I will discuss how these issues influence the performance of the supervised and semi-supervised learning methods used in this study.

4 Experiments

To assess performance of different classification methods, benchmark datasets of known families are needed. However, currently there are no suitable datasets available. I hand-curated a novel benchmark dataset from mouse sequences. Using this dataset, I evaluated the effectiveness of the semi-supervised learning method in separating family members from nonfamily members. I investigated the stability of the method under different parameters, and compared it with the supervised learning method.

Graph construction I collected 7000 complete mouse sequences from the SwissProt Version 44 [2] *, released in 09/2004. Similarity is typically assessed using a direct measure of similarity (e.g., bitscores) or a measure of the statistical significance of sequence similarity (e.g. E values) [1]. However, these measures will result in wrong edges due to convergent evolution and/or domain shuffling. Instead, I used a novel similarity score that avoids false positives due to insertion of similar sequence fragments into unrelated sequences [45]. The range of this similarity score is (0, 1). For sequence pairs that do not have significant similarity, a pseudo-score of 0.001 was assigned to facilitate the computation.

Test set construction To test the various methods, we required sets of proteins from the same family. I constructed a list of sequences from eighteen known families using reports from nomenclature committees † and articles from the literature. Types of evidence presented in these articles include intron/exon structure, phylogenetics and synteny. As a result of this process, I have a list of 1,137 proteins each known to be from one of the following families: ACSL[7, 31], ADAM (a family of proteins containing A Disintegrin And Metalloproteinase domain) [34, 46, 53], DVL (Dishevelled protein family) [43, 51], Fox (Forkhead transcription factor family) [21, 32], GATA (GATA binding proteins) [28, 37], Kinase [6, 16, 40, 44], Kinesin [19, 25, 33], Laminin [10, 17], Myosin [13, 15, 39], Notch [23, 30, 49], PDE (Phosphodiesterases) [9], SEMA (Semaphorin) [38, 54], TNFR (Tumor Necrosis Factor Receptors) [26, 29], TRAF (Tumor necrosis factor Receptor Associated Factors) [18], USP (Ubiquitin Specific Proteases) [22, 52] and WNT[27, 42]. Our test families exhibit one of the four following graph structures:

- Family members form a clique. Some family members have strong sequence similarity ($S(i,j) \geq 0.4$) to genes outside the family.
- Family members form a clique. All family members have weak sequence similarity ($S(i,j) \leq 0.4$) to genes outside the family.

*http://us.expasy.org/sprot/

†http://www.gene.ucl.ac.uk/nomenclature/

- Family members are in the same connected component, but do not form a clique.
- Family members are in more than one connected component.

The structure of all 18 families is summarized in Table 2.

	Family size	Similar	Clique	Connected	Not connected
ACSL	5	4	W		
DVL	3	31	W		
Fox	30	95	W		
GATA	6	8	W		
SEMA	16	108	W		
TRAF	6	110	W		
Tbox	9	11	W		
WNT	19	2	W		
Kinesin	18	298	S		
Laminin	11	380	S		
Myosin	12	764	S		
Notch	4	260	S		
ADAM	26	139		X	
FGF	20	0		X	
Kinase	293	1196			X
PDE	15	9			X
TNFR	24	86			X
USP	18	42			X

Table 2: Graph structure of test families. Similar: number of sequences that are not in the same family but have significant sequence similarity ($S(i,j) > 0$) to one or more family members. W: Edges to sequences outside the family have weak edge weights ($S(i,j) \geq 0.4$). S: Edges to sequences outside the family have strong edge weights ($S(i,j) < 0.4$).

Basis for evaluation I evaluated classifier performance numerically using Area Under receiver operating characteristic Curve (AUC) scores. The AUC score provides a single measure of classification accuracy, without considering the cutoff [47]. Suppose we are given two classes $\{a\}$ and $\{b\}$ of size p and q, respectively. Let $a_i, i = 1, 2, ..., p$ and $b_j, j = 1, 2, ..., q$, be the values assigned to the members of $\{a\}$ and $\{b\}$ respectively. The AUC score is calculated as follows:

$$AUC = \frac{\sum_{i=1}^{p} \sum_{j=1}^{q} X(a_i, b_j)}{pq},$$

where

$$X(a_i, b_j) = \begin{cases} 1 & \text{if } a_i > b_j \\ 0.5 & \text{if } a_i = b_j \\ 0 & \text{if } a_i < b_j \end{cases} \quad (5)$$

AUC scores range from 0.5 to 1. A value of 1.0 indicates perfect separation of the two classes, while random assignment of elements to the two classes would result in a score of roughly 0.5. If we consider sensitivity and specificity to be equally important, a larger AUC score indicates a better classifier.

For some applications, such as selecting targets for experimental biologists, AUC scores alone are not sufficiently informative. Since it is expensive and time consuming to perform experiments to study proteins in depth, experimental biologists tend to start with the top ranked prediction and stop at the first false prediction. Therefore, for experimental biologists, the top ranked are especially of interest and a method that ranks fewer family members after the first nonfamily member is desirable. For this reason, I also considered the number of missed family members (false negatives) observed after the first nonfamily member (false positive). Although the number of false negatives provide some information, it doesn't tell us whether or not proteins of lower rank are separable.

Experiments performed I evaluated how the performance of semi-supervised learning method changed with different parameters. Tested parameters include the hyperparameter, σ, in the smoothing function, the number of labeled family members (LF), and the number of labeled nonfamily members (LN).

The σ values I tried include 0.05, 0.1, 0.5, 1, 2, 10, and 100. These values cover a variety of situations as regard to the relationship of $S(i,j)$ and $W(i,j)$ as shown in Fig. 1. When $\sigma = 0.05$, the weight $W(i,j)$ drops dramatically with the decrease of the similarity score, $S(i,j)$. When the similarity score $S(i,j)$ is less than 0.8, the weight $W(i,j)$ is close to zero. This implies that only close data points have significant contribution to the label of the studied data point, while the data points with similarity less than 0.8 have negligible contribution. In contrast, when $\sigma = 100$, the weights for most data points are close to one. This indicates that almost all data points in the graph have equal contribution, no matter how distant they are from the studied data point.

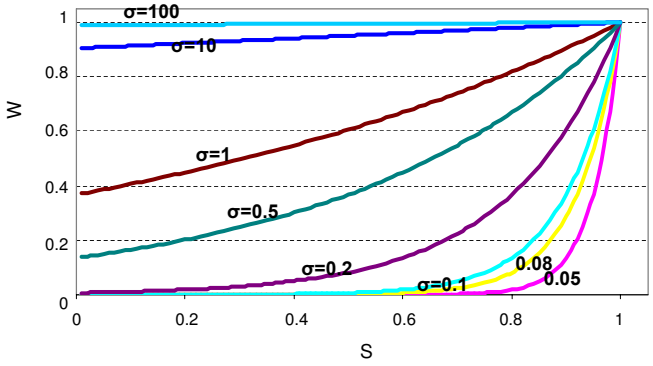

Figure 1: The effects of σ values on $W(i,j)$, where $W(i,j)$ is defined as $exp((1 - S(i,j))/\sigma)$.

The LF values I tested vary from 10% to 70% of the family size, as shown in Table 3. The LN values I tested include 100, 500 and 1000. This corresponds to approximately 1% to 10% of the number of vertices outside the family. I tried 20 different samples for each LF, LN pair. In each sample, the labeled family members are randomly selected from the family members and the

labeled nonfamily members are randomly selected from sequences that do not have significant matches to any sequence in that family. This is to ensure that the negative examples are true negatives. For each sample, I experimented with different σ values to determine the impact of smoothing on performance. For each combination of LF, LN, and σ, I calculated the average AUC score and the average number of false negatives over 20 samples. In the following section, I reported the maximum average AUC score and the minimum average false negatives over all values of σ. In general, a wide range of σ values can achieve the best performance.

Family	Size	LF
ACSL	5	1, 3
ADAM	26	3, 5, 7, 9,15
DVL	3	1
FGF	20	3, 5, 7, 11, 15
Fox	30	3, 9, 15
GATA	6	1,3
Kinase	293	3, 7, 11, 15, 20, 50, 150
Kinesin	18	2, 6, 9
Laminin	11	1, 3, 5, 7
Myosin	12	2, 4, 6, 9
Notch	4	1, 2
PDE	15	2, 5, 7, 10
SEMA	16	2, 5, 8
TNFR	24	2, 4, 8, 12, 18
TRAF	9	1, 3
Tbox	6	2, 5
USP	18	2, 4, 6, 9,13
WNT	19	2, 9

Table 3: Family size and the number of labeled family members tested for each family.

5 Results and discussion

In this paper, I evaluated performance of the semi-supervised and the supervised learning methods using 18 hand curated families. Tables 4 and 5 summarize the best average AUC score and the minimal average false negatives after the first false positive for each LF, LN pair. These results suggest that the graph-based semi-supervised method can effectively identify protein family members and the semi-supervised method has equal or better performance compared to the supervised method. For 11 families, both semi-supervised and supervised methods have perfect performance, *i.e.*, the AUC score is equal to one and the number of false negatives is equal to zero. For the remaining seven families, when LF is small, for each LF, LN pair, the AUC score of the semi-supervised method is equal to or larger than that of the supervised method (Table 4). For five of these seven families, the semi-supervised method also has fewer false negatives than the supervised method (Table 5). All these suggest that generally the semi-supervised method

has equal or better performance than the supervised method. Moreover, the semi-supervised method is more suitable than the supervised method in selecting targets for experimental biologists. Detailed analysis shows that the performance of both semi-supervised and supervised methods depend on the graph structure of test families. For almost all families that form a clique, both methods have perfect performance. For families that do not form a clique, generally the semi-supervised method outperforms the supervised method.

	Smallest tested LF				Largest tested LF			
	LN = 100		LN = 1000		LN = 100		LN = 1000	
	Super	Semi	Super	Semi	Super	Semi	Super	Semi
Clique, strong								
ACSL	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
DVL	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Fox	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
GATA	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
SEMA	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Tbox	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
TRAF	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
WNT	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Clique, weak								
Kinesin	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Laminin	0.9999	0.9999	0.9999	0.9999	1.0000	1.0000	1.0000	1.0000
Myosin	0.9999	0.9999	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Notch	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Connected								
ADAM	0.9552	1.0000	0.9983	1.0000	1.0000	1.0000	1.0000	1.0000
FGF	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Not Connected								
Kinase	0.9525	0.9645	0.9749	0.9738	0.9664	0.9672	0.9802	0.9768
PDE	0.9292	0.9364	0.9640	0.9589	0.9561	0.9603	0.9793	0.9769
TNFR	0.9225	0.9420	0.9556	0.9526	0.9629	0.9671	0.9845	0.9866
USP	0.9781	0.9798	0.9912	0.9895	0.9854	0.9875	0.9906	0.9895

Table 4: The maximum average AUC score over all σ values tested, for each LF, LN pair. The results given are averaged over 20 tests.

I also investigated the impact of different parameters for both semi-supervised and supervised methods. Table 4 and 5 show that, for both methods, generally with the increase of LF and LN, the AUC scores increase and the number of false negative decreases, indicating that performance improves. The impact of σ varies for different families. For most families, the performance is not very sensitive to σ. For some other families, the effect is more complex.

In the following sections, I use four families, one from each graphical type, to show the performance of both supervised and semi-supervised methods and the impact of these parameters in detail.

The minimum number of false positives after the first negative								
	Smallest tested LF				Largest tested LF			
	LN = 100		LN = 1000		LN = 100		LN = 1000	
	Super	Semi	Super	Semi	Super	Semi	Super	Semi
Clique, strong								
ACSL	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DVL	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Fox	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GATA	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SEMA	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Tbox	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
TRAF	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
WNT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Clique, weak								
Kinesin	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Laminin	1.77	1.69	1.97	2.05	0.10	0.10	0.37	0.09
Myosin	0.00	0.50	0.00	0.15	0.00	0.00	0.00	0.00
Notch	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Connected								
ADAM	0.33	0.00	0.13	0.00	0.00	0.00	0.00	0.00
FGF	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Not Connected								
Kinase	117.00	135.97	126.48	137.17	87.17	100.73	79.00	118.90
PDE	2.90	2.50	2.47	2.33	0.53	0.60	0.40	0.55
TNFR	5.13	3.85	5.43	4.70	1.23	1.15	0.50	0.85
USP	1.00	0.95	0.80	0.75	0.23	0.20	0.27	0.25

Table 5: The minimum average number of false positives over all σ values tested, for each LF, LN pair. The results given are averaged over 20 tests.

5.1 Fox

The Fox family is highly conserved. All family members are significantly similar to each other, *i.e.*, form a clique. In addition, several Fox proteins have weak raw similarity to some nonfamily members. For both semi-supervised and supervised methods, the choice of LF, LN, and σ does not affect the performance: perfect performance is obtained for all parameters tested. Fig. 3 shows three *rankplots*, where all sequences are ordered based on the predicted f(x) value, for the semi-supervised method for one test sample with LF = 3, LN = 100, under different σ. In all three plots, family members have greater $f(x)$ values than nonfamily members, and a clear cutoff can separate them.

The performance can be explained by the graph structure. In this study, the edge weights are calculated using a smoothing function (Equation 1). Because family-family pairs have much greater raw similarity scores than those of family-nonfamily pairs, when σ values range from 0.1

to 10.0, the edge weights for family-family pairs are significantly greater than those of family-nonfamily pairs (Fig. 2), and family members form a clique. In this case, family classification is comparatively easy. Not too much information is needed from labeled data, which explains why the performance is stable for different values of LF and LN. Note, if σ is extremely small or large, *e.g.*, $\sigma > 10$ or $\sigma < 0.05$, the nice graph property no longer hold. For example, if σ is extremely small, the edge weights between some family-family pairs will be too weak. In that case, the family no longer forms a clique. In contrast, if σ is too large, the edge weights between family-family pairs and family-nonfamily pairs will not be distinguishable. Under these two extreme cases, the performance of classification will not be perfect (data not shown here).

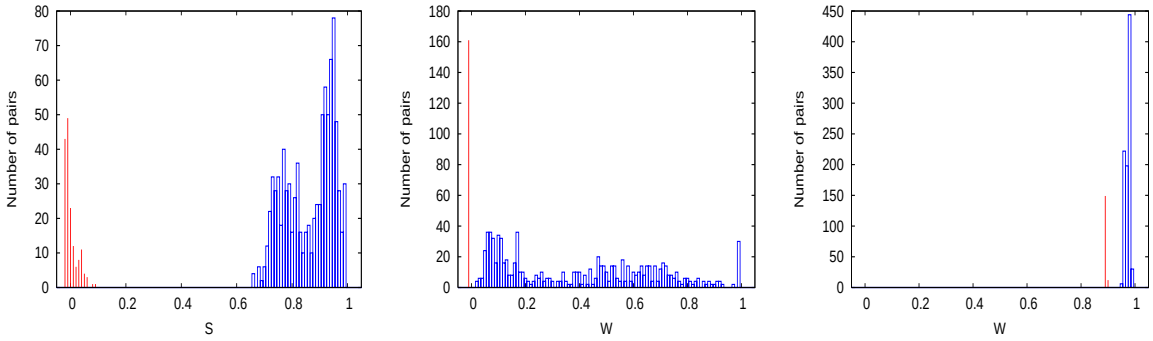

Figure 2: In Fox, the histogram of (A) original similarity scores, (B) edge weights when $\sigma = 0.1$, and (C) edge weights when $\sigma = 10$, respectively. Blue color represents the edges between family-family and red color represent the edge weights between family-nonfamily pairs.

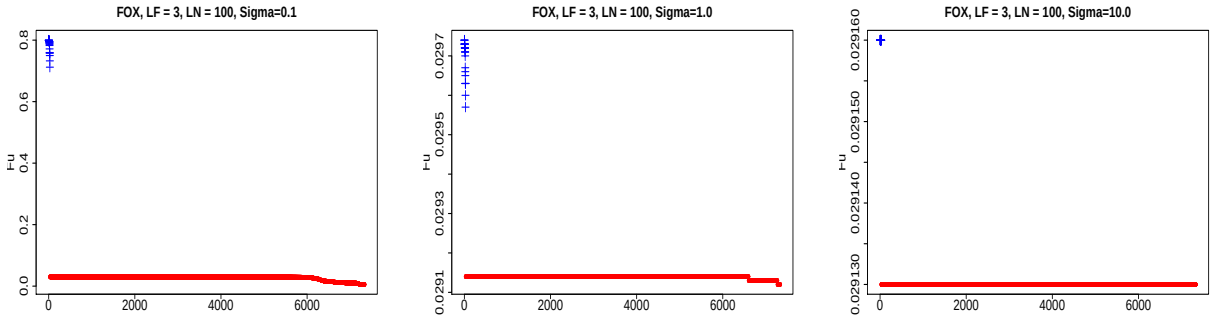

Figure 3: The rank plots for Fox with LF = 3, LN = 100 when (A) $\sigma = 0.1$ (B)$\sigma = 1.0$ and (C) $\sigma = 10$. Blue points represent family members and red points represent nonfamily members.

5.2 Notch

Similar to the Fox family, the Notch family is highly conserved and all family members form a clique. Unlike the Fox family, Notch family members have strong similarity to some sequences

outside the family. A large number of family-nonfamily pairs have similarity scores greater than 0.6.

The performance of the semi-supervised and supervised methods in Notch is also similar to that of Fox. For all pairs of LF, LN tested, perfect performance is obtained for both semi-supervised and supervised methods. However, the range of optimal σ values for the semi-supervised method ($0.05 \leq \sigma \leq 1.0$) is smaller that of the Fox ($0.05 \leq \sigma \leq 10.0$). When $\sigma = 10$, the performance for the semi-supervised method is not perfect.

The different performance of the semi-supervised method in Notch and Fox can be explained by the large raw similarity scores between Notch sequences and sequences outside the family. As shown in Fig. 4, when $\sigma = 10$, the weights for family-family pairs and family-nonfamily pairs are quite similar. Because of label propagation in the semi-supervised method, the impact of nonfamily members is substantial when the weight between family-nonfamily pairs increases, leading to a misclassification. For FOX, this situation happens only when $\sigma > 10$.

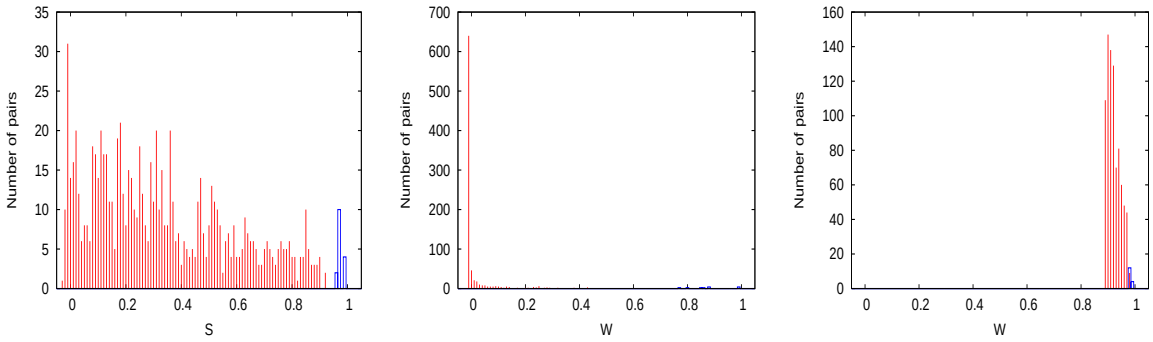

Figure 4: In Notch, the histogram of (A) original similarity scores, (B) edge weights when $\sigma = 0.1$, and (C) edge weights when $\sigma = 10$, respectively. Blue color represents the edges between family-family and red color represent the edge weights between family-nonfamily pairs.

5.3 ADAM

The ADAM family forms a single connected component, but does not form a clique. The family size is 26. However, one ADAM sequence is significant similar to only eight other ADAM sequences.

As shown in Fig. 5, the semi-supervised classification method always has perfect performance for all pairs of LF, LN tested. In contrast, the supervised method does not have perfect performance when the number of labeled family members is less than seven (27% of the family size).

The reason that the semi-supervised method outperforms the supervised method when LF is small is that the semi-supervised method exploits graph structure information, while the supervised method does not. For the supervised method, perfect classification requires that every unlabeled family member to be adjacent to at least one labeled family member. When LF is small, some family members, such as the sequence which is similar to only 8 out of 26 ADAM sequences, may not be adjacent to any labeled family members. However, when LF increases,

the probability that a family sequence is not adjacent to any labeled family members decreases. This explains why the supervised method has perfect performance when LF is large. For the graph-based semi-supervised method, since labeled data can affect other data points through label propagation, the direct connection between unlabeled family member and labeled family members is not required. Therefore, the semi-supervised method has perfect performance when LF is small.

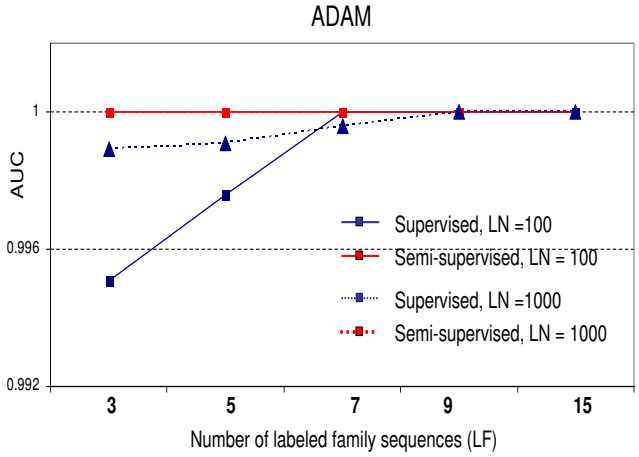

Figure 5: In ADAM, the maximum average AUC scores for each LF, LN pair over all σ values tested. The results are given averaged over 20 tests

5.4 TNFR

TNFR is characterised by low sequence similarity within the family. TNFR is a family that does not form a single connected component. In our test set, there are 24 TNFR proteins. The largest component contains 23 TNFR sequences and 5594 sequences outside the family. Among these 23 TNFR sequences, 20 TNFR sequences directly connect to other TNFR sequences. Three proteins indirectly linked to these 20 proteins through nonfamily proteins. The remaining protein is in a different component.

Neither the semi-supervised method nor the supervised method has perfect performance. Figs. 6 and 7 show that for all values of LF and LN = 100, the semi-supervised method has better performance than the supervised method, as indicated by the larger AUC score and fewer of false negatives.

6 Discussion

In this paper, I applied Zhu and Lafferty's graph-based semi-supervised learning algorithm [56] to protein family classification. To my knowledge, this is the first application of the algorithm to a problem in molecular biology. A different semi-supervised method was previously applied to domain classification, one related but different biological problem [50].

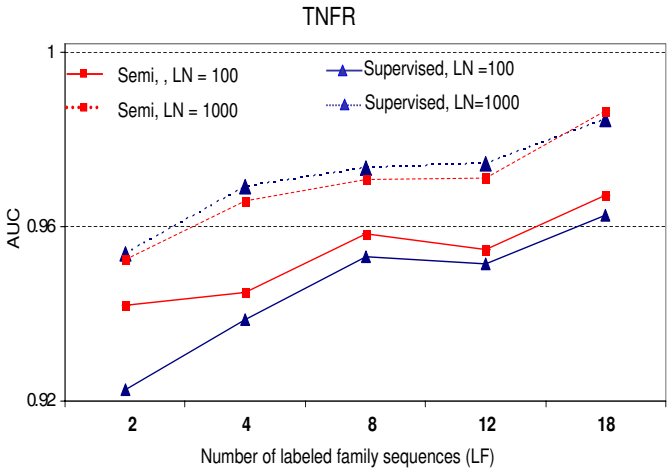

Figure 6: In TNFR, the maximum average AUC scores for each LF, LN pair over all σ values tested. The results are given averaged over 20 tests

My empirical study shows that the semi-supervised learning method is effective in protein family classification. The performance of the semi-supervised learning algorithm is better than the corresponding supervised method when the number of labeled data is small. These results suggest that the semi-supervised learning algorithm is a promising approach for gene annotation and selecting target genes for experimental study.

The performance of the method was tested on a dataset of 18 well-studied families. Of these, 12 form cliques. Both supervised and semi-supervised learning performed well on these families. The semi-supervised method outperforms the supervised method for all families that do not form a clique except FGF where they have equal performance. This is due to the label propagation properties of the semi-supervised learning algorithm. The semi-supervised method performs very well on families that form a single connected component, less well on families that span more than one connected component.

The semi-supervised method is most promising for families that do not form cliques. Cliques represent two thirds of my test set. However, it may be that cliques are over represented in my test set because these families are well studied: it is easy to identify families that are highly conserved and therefore form a clique. It is possible that a much smaller fraction of families in the protein universe form cliques.

The challenging problem for future work on semi-supervised protein family classification is how to identify sequences that are not in the same connected component. There are two promising approaches to this problem: we can either build the similarity graph using different similarity scores that promote great connectivity or enhance the label propagation properties of the graph-based semi-supervised method.

This study also contributes to expand applications of the semi-supervised method. The graph-based semi-supervised learning method has been successfully applied in different areas, including face recognition and text recognition. However, the protein family classification is substantially different from previous applications in two aspects. First, the graph structure of the protein

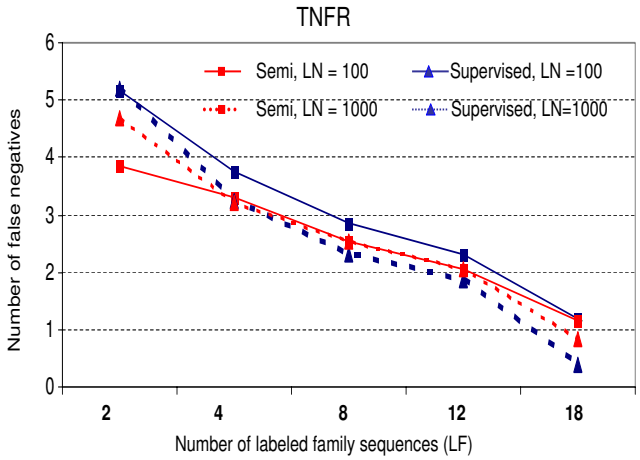

Figure 7: In TNFR, the minimum average number of false positives for each LF, LN pair over all σ values tested. The results are given averaged over 20 tests

sequence similarity graph is quite different from the abstract model in previous applications. As shown in Fig 2 [56], the abstract model in previous applications seem to have long tails, however, for protein family classification, the graph is more like islands in the sea. Since graph-based semi-supervised method is based on exploiting graph structures, it is valuable to investigate the performance of the semi-supervised method in different types of graphs. Second, in this paper, semi-supervised learning method is applied to an *atypical* binary classification problem. In protein family classification, all proteins are classified into two groups, one group is composed of family members, while the other group is composed of the remaining proteins. It is notable that the second group is not homogeneous, *i.e.*, all members in the second group do not own some common features other than not the same family as the studies group. It is interesting to know that the semi-supervised method can also work for this type of problem. The effectiveness of the semi-supervised method in protein family classification encourage us to further exploit the application of the semi-supervised methods.

Acknowledgments

I would like to thank S. H. Bryant and L. Y. Geer for providing domain architecture data, my advisors John Lafferty and Dannie Durand for helpful discussions and insight, and all people in the Durand Lab, especially Robbie Sedgewick for their help.

References

[1] S. Altschul. Evaluating the statistical significance of multiple distinct local alignments. In Subai, editor, *Theoretical and Computational Methods in Genome Research*, pages 1–14. 1997.

[2] A. Bairoch, R. Apweiler, C. Wu, W. Barker, B. Boeckmann, S. Ferro, E. Gasteiger, H. Huang, R. Lopez, M. Magrane, M. Martin, D. Natale, C. O'Donovan, N. Redaschi, and L. Yeh. The universal protein resource (UniProt). *Nucleic Acids Res.*, 33:D154–9, 2005.

[3] A. Blum and S. Chawla. Learning from labeled and unlabeled data using graph mincuts. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 19–26, San Francisco CA USA, 2001. ICML, Morgan Kaufmann Publishers Inc.

[4] A. Blum, J. Lafferty, M. Rwebangira, and R. Reddy. Semi-supervised learning using randomized mincuts. In *Proceedings of the Twenty-first International Conference on Machine Learning*, page 13, New York, NY, USA, 2004. ICML, ACM Press.

[5] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, pages 92–100, New York, NY, USA, 1998. COLT' 98, ACM Press.

[6] S. Cheek, H. Zhang, and N. Grishin. Sequence and structure classification of kinases. *J Mol Biol*, 320(4):855–881, Jul 2002.

[7] R. Coleman, T. Lewin, C. Van Horn, and M. Gonzalez-Baro. Do long-chain acyl-CoA synthetases regulate fatty acid entry into synthetic versus degradative pathways? *J Nutr*, 132(8):2123–2126, Aug 2002.

[8] R. Dara, S. Kremer, and D. Stacey. Clustering unlabeled data with SOMs improves classification of labeled real-world data. In *Proceedings of the 2002 International Joint Conference on Neural Networks*, volume 3, pages 2237–2242, May 2002.

[9] E. Degerman, P. Belfrage, and V. Manganiello. Structure, localization, and regulation of cGMP-inhibited phosphodiesterase (PDE3). *J Biol Chem*, 272(11):6823–6826, Mar 1997.

[10] J. Engel. Laminins and other strange proteins. *Biochemistry*, 31(44):10643–10651, Feb 1992.

[11] A. Enright, K. Kunin, and C. Ouzounis. Protein families and TRIBES in genome sequence space. *Nucleic Acids Res*, 31(15):4632–4638, Aug 2003.

[12] A. Enright and C. Ouzounis. GeneRAGE: a robust algorithm for sequence clustering and domain detection. *Bioinformatics*, 16(5):451–457, May 2000.

[13] B. Foth, M. Goedecke, and D. Soldati. New insights into myosin evolution and classification. *Proc Natl Acad Sci U S A*, 103(10):3681–3686, Mar 2006.

[14] A. Fujino, N. Ueda, and K. Saito. A hybrid generative/discriminative approach to semi-supervised classifier design. In Manuela M. Veloso and Subbarao Kambhampati, editors,

Proceedings, The Twentieth National Conference on Artificial Intelligence and the Seventeenth Innovative Applications of Artificial Intelligence Conference, pages 764–769. AAAI Press AAAI Press / The MIT Press, July 2005.

[15] H. Goodson and S. Dawson. Multiplying myosins. *Proc Natl Acad Sci U S A*, 103(10):3498–3499, Mar 2006.

[16] S. Hanks. Genomic analysis of the eukaryotic protein kinase superfamily: a perspective. *Genome Biol*, 4(5):111, 2003.

[17] H. Hutter, B. Vogel, J. Plenefisch, C. Norris, R. Proenca, J. Spieth, C. Guo, S. Mastwal, X. Zhu, J. Scheel, and E. Hedgecock. Conservation and novelty in the evolution of cell adhesion and extracellular matrix genes. *Science*, 287(5455):989–994, Feb 2000.

[18] J. Inoue, T. Ishida, N. Tsukamoto, N. Kobayashi, A. Naito, S. Azuma, and T. Yamamoto. Tumor necrosis factor receptor-associated factor (TRAF) family: adapter proteins that mediate cytokine signaling. *Exp. Cell Res.*, 254(1):14–24, 2000.

[19] N. Iwabe and T. Miyata. Kinesin-related genes from diplomonad, sponge, amphioxus, and cyclostomes: divergence pattern of kinesin family and evolution of giardial membrane-bounded organella. *Mol Biol Evol*, 19(9):1524–1533, Sep 2002.

[20] T. Joachims. Transductive inference for text classification using support vector machines. In Ivan Bratko and Saso Dzeroski, editors, *Proceedings of 16th International Conference on Machine Learning*, pages 200–209, San Francisco, US, 1999. ICML99, Morgan Kaufmann Publishers.

[21] K. Kaestner, W. Knochel, and D. Martinez. Unified nomenclature for the winged helix/forkhead transcription factors. *Genes Dev.*, 14(2):142–146, Jan 2000.

[22] J. Kim, K. Park, S. Chung, O. Bang, and C. Chung. Deubiquitinating enzymes as cellular regulators. *J Biochem (Tokyo)*, 134(1):9–18, Jul 2003.

[23] R. Kortschak, R. Tamme, and M. Lardelli. Evolutionary analysis of vertebrate Notch genes. *Dev Genes Evol*, 211(7):350–354, Jul 2001.

[24] A. Krause, S. Haas, E. Coward, and M. Vingron. Systers, genenest, splicenest: exploring sequence space from genome to protein. *Nucleic Acids. Res.*, 30:299–300, 2002.

[25] C. Lawrence et al. A standardized kinesin nomenclature. *J Cell Biol*, 67(1):19–22, 2004.

[26] R. Locksley, N. Killeen, and M. Lenardo. The TNF and TNF receptor superfamilies: integrating mammalian biology. *Cell*, 104(4):487–501, Feb 2001.

[27] C. Logan and R. Nusse. The Wnt signaling pathway in development and disease. *Annu Rev Cell Dev Biol*, 20:781–810, 2004.

[28] J. Lowry and W. Atchley. Molecular evolution of the GATA family of transcription factors: conservation within the DNA-binding domain. *J Mol Evol*, 50(2):103–115, Feb 2000.

[29] D. MacEwan. TNF ligands and receptors—a matter of life and death. *Br. J. Pharmacol.*, 135(4):855–875, Feb 2002.

[30] E. Maine, J. Lissemore, and W. Starmer. A phylogenetic analysis of vertebrate and invertebrate Notch-related genes. *Mol Phylogenet Evol*, 4(2):139–149, Jun 1995.

[31] D. Mashek et al. Revised nomenclature for the mammalian long-chain acyl-CoA synthetase gene family. *J Lipid Res*, 45(10):1958–61, Oct 2004.

[32] F. Mazet, J. Yu, D. Liberles, L. Holland, and S. Shimeld. Phylogenetic relationships of the Fox (Forkhead) gene family in the bilateria. *Gene*, 316(Oct 16):79–89, 2003.

[33] H. Miki, M. Setou, and N. Hirokawa. Kinesin superfamily proteins (KIFs) in the mouse transcriptome. *Genome Res*, 13(6B):1455–1465, Jun 2003.

[34] A. Nicholson, S. Malik, J. Logsdon Jr., and E. Van Meir. Functional evolution of ADAMTS genes: evidence from analyses of phylogeny and gene organization. *BMC Evol Biol*, 5(1):11, 2005.

[35] K. Nigam, A. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using EM. *Mach. Learn.*, 39(2-3):103–134, 2000.

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

[37] R. Patient and J. McGhee. The GATA family (vertebrates and invertebrates). *Curr Opin Genet Dev*, 12(4):416–22, Aug 2002.

[38] J. Raper. Semaphorins and their receptors in vertebrates and invertebrates. *Curr Opin Neurobiol*, 10(1):88–94, Feb 2000.

[39] T. Richards and T. Cavalier-Smith. Myosin domain evolution and the primary divergence of eukaryotes. *Nature*, 436(7054):1113–1118, Aug 2005.

[40] D. Robinson, Y. Wu, and S. Lin. The protein tyrosine kinase family of the human genome. *Oncogene*, 19(49):5548–5557, Nov 2000.

[41] O. Sasson, A. Vaaknin, H. Fleischer, E. Portugaly, Y. Bilu, N. Linial, and M. Linial. ProtoNet: hierarchical classification of the protein space. *Nucleic Acids Res*, 31(1):348–352, Jan 2003.

[42] M. Schubert, L. Holland, N. Holland, and D. Jacobs. A phylogenetic tree of the Wnt genes based on all available full-length sequences, including five from the cephalochordate amphioxus. *Mol Biol Evol*, 17(12):1896–903, Dec 2000.

[43] L. Sheldahl, D. Slusarski, P. Pandur, J. Miller, M. Kuhl, and R. Moon. Dishevelled activates Ca2+ flux, PKC, and CamKII in vertebrate embryos. *J Cell Biol*, 161(4):769–77, May 2003.

[44] S. Shiu and W. Li. Origins, lineage-specific expansions, and multiple losses of tyrosine kinases in eukaryotes. *Mol Biol Evol*, 21(5):828–840, 2004.

[45] N. Song. *Homology identification for multidomain proteins*. PhD thesis, Carnegie Mellon University, 2007.

[46] A. Stone, M. Kroeger, and Q. Sang. Structure-function analysis of the ADAM family of disintegrin-like and metalloproteinase-containing proteins (review). *J Protein Chem*, 18(4):447–465, May 1999.

[47] J. Swets. The relative operating characteristic in psychology. *Science*, 182(4116):990–1000, 1973.

[48] M. Szummer and T. Jaakkola. Information regularization with partially labeled data. *Adv Neural Inf Process Syst*, 15, 2002.

[49] J. Westin and M. Lardelli. Three novel Notch genes in zebrafish: implications for vertebrate Notch gene evolution and function. *Dev Genes Evol*, 207(1):51–63, May 1997.

[50] J. Weston, C. Leslie, D. Zhou, and W. Noble. Semi-supervised protein classification using cluster kernels. *Adv Neural Inf Process Syst*, (595-602), 2004.

[51] K. Wharton. Runnin' with the Dvl: proteins that associate with Dsh/Dvl and their significance to Wnt signal transduction. *Dev Biol*, 253(1):1–17, Jan 2003.

[52] S. Wing. Deubiquitinating enzymes—the importance of driving in reverse along the ubiquitin-proteasome pathway. *Int J Biochem Cell Biol*, 35(5):590–605, May 2003.

[53] T. Wolfsberg and J. White. ADAMs in fertilization and development. *Dev Biol*, 180(2):389–401, Dec 1996.

[54] U. Yazdani and J. Terman. The semaphorins. *Genome Biol*, 7(3):211, 2006.

[55] G. Yona, N. Linial, and M. Linial. Protomap: automatic classification of protein sequences and hierarchy of protein families. *Nucleic Acids Res*, 28(1):49–55, Jan 2000.

[56] X. Zhu, Z. Ghahramani, and J. Lafferty. Semi-Supervised learning using Gaussian fields and harmonic functions. In *The Twentieth International Conference on Machine Learning*, pages 912–919. ICML03, ACM Press, 2003.