Technical Report

Department of Computer Science
and Engineering
University of Minnesota
4-192 EECS Building
200 Union Street SE
Minneapolis, MN 55455-0159 USA

TR 08-001

Incorporating Functional Inter-relationships into Protein Function Prediction Algorithms

Gaurav Pandey, Chad L. Myers, and Vipin Kumar

January 7, 2008

Incorporating Functional Inter-relationships into Protein Function Prediction Algorithms

Gaurav Pandey
Dept of Comp Sc & Engg
Univ of Minnesota
Minneapolis, MN, USA
gaurav@cs.umn.edu

Chad L. Myers
Dept of Comp Sc & Engg
Univ of Minnesota
Minneapolis, MN, USA
cmyers@cs.umn.edu

Vipin Kumar
Dept of Comp Sc & Engg
Univ of Minnesota
Minneapolis, MN, USA
kumar@cs.umn.edu

ABSTRACT

Functional classification schemes that serve as the basis for annotation efforts in several organisms (e.g. the Gene Ontology) are often the source of gold standard information for computational efforts at supervised gene function prediction. While successful function prediction algorithms have been developed, few previous efforts have utilized more than the gene-to-function class labels provided by such knowledge bases. For instance, the Gene Ontology not only captures gene annotations to a set of functional classes, but it also arranges these classes in a DAG-based hierarchy that captures rich inter-relationships between different classes. These inter-relationships present both opportunities, such as the potential for additional training examples for small classes from larger related classes, and challenges, such as a harder to learn distinction between similar GO terms, for standard classification-based approaches. In this paper, we propose to enhance the performance of classification-based protein function prediction algorithms by addressing these issues, using the same inter-relationships between functional classes. Using a standard measure for evaluating the semantic similarity between nodes in an ontology, we quantify and incorporate these inter-relationships into the k-nearest neighbor classifier. We present experiments on several large genomic data sets, each of which is used for the modeling and prediction of different sets of over hundred classes from the GO Biological Process ontology. The results show that this incorporation produces more accurate predictions for a large number of the functional classes considered, and also that the classes benefitted most by this approach are those containing the fewest members. In addition, we show how our proposed framework can be used for integrating information from the entire GO hierarchy for improving the accuracy of predictions made over a set of base classes.

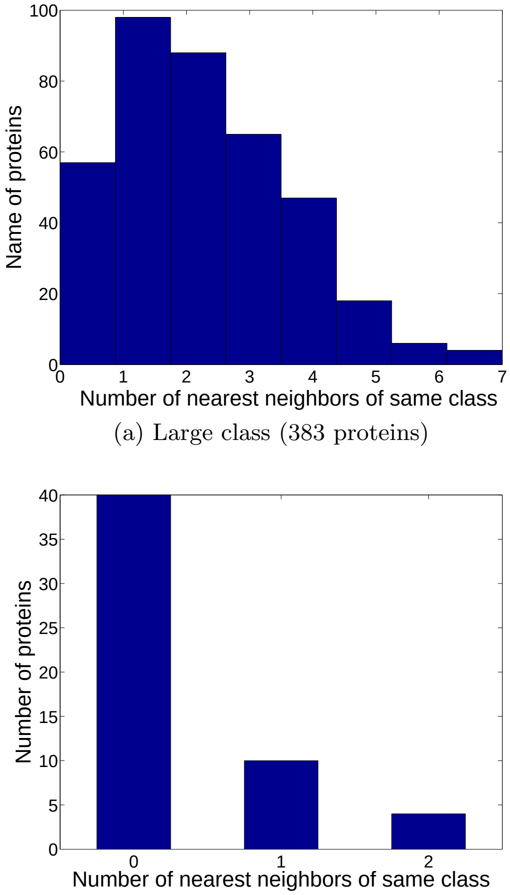
1. INTRODUCTION

A variety of recently available high throughput data sets, such as protein-protein interaction networks, microarray data and genome sequences, offer important insights into the mechanisms leading to the accomplishment of a protein's function. However, the complexity of analyzing these data sets manually has motivated the development of numerous computational approaches for predicting protein function [8, 26]. For a recent comprehensive survey on this topic, see Pandey *et al* (2006) [21].

One of the most popular methods used for predicting protein function from biological data is classification [30, 5, 15]. Under the traditional classification framework, each protein is represented by a set of features, such as the expression profile of its corresponding gene or the set of proteins it interacts with. Now, for each functional class, a model is constructed using the feature sets of the proteins annotated with this class. This model is then used to decide if an unannotated query protein should be annotated with this class. The key premise underlying this methodology for predicting protein function is that proteins belonging to the same functional class have ”similar” biological attributes.

Standard classification or predictive modeling techniques for function prediction rely on positive and negative examples from functional classification schemes, such as the Gene Ontology [1] or MIPS [24], and typically treat each functional class separately. However, this standard approach fails to capture one of the key properties of such classification schemes: most schemes not only provide annotations to functional classes, but also capture inter-relations between the functional classes. For example, the Gene Ontology (GO) is arranged as a directed acyclic graph in which the GO terms form a hierarchy capturing everything from relatively general functions (e.g. metabolism) to specific, biological roles (e.g. nucleotide excision repair). Such an organization of classes, particularly in the case of GO, poses two important challenges for predictive modeling techniques for function prediction. First, several studies [17, 2, 27] have concluded that proteins in inter-related, or ”similar” GO classes tend to have similar biological attributes. This limits the applicability of the key premise of classification-based function prediction discussed above, since distinguishing between such similar GO classes becomes hard.

The second important issue that arises is that often, an insufficient number of training examples are available for building models for functional classes, especially for classes deep in the hierarchy. This phenomenon is illustrated in Figure 1 for two classes, which had 383 and 54 member proteins in Mnaimneh *et al*'s gene expression data set [18], representing the a large class and one of the classes of median size respectively from the GO biological process ontology. These



(a) Large class (383 proteins)

(b) Smaller class (54 proteins)

Figure 1: Distribution of the number of nearest neighbors of the same class in a neighborhood of size 20 for two classes in Mnaimneh *et al*'s data set

histograms show, for each protein in these classes, how many "nearby" proteins belong to the same class. Neighborhood is restricted to the twenty proteins with the most similar expression profiles to the query protein, using correlation as the similarity measure. These plots show that for most of the proteins in both the large, as well as the much smaller class, only a limited number of similar proteins in the same class are available. For instance, for the large class, 243 of the 383 proteins have less than three similar proteins in the same class, while two is the maximum number of same class neighboring proteins for proteins in the smaller class. In fact, 40 of the 54 proteins in the smaller class have no proteins of the same class in their neighborhood. This illustrates the difficulty of building classification models for small functional classes that are located deep in the GO ontology. In summary, these challenges make it hard to build accurate classification models.

However, the availability of the same well-defined structure of relationships between functional classes in the form of Gene Ontology raises the following key question: "Can the performance classification algorithms for function prediction be improved by incorporating these inter-relationships into them?". In this paper, we propose to address this question using an approach shown visually in Figure 2. As illustrated by this figure, our approach uses evidence in "nearby" proteins belonging to similar classes to bolster the evidence for annotation of the query protein with the target class[1]. Using Lin's measure [16] for evaluating the semantic similarity between nodes in an ontology, we incorporate these inter-relationships into the k-nearest neighbor classifier [30]. We evaluate our algorithm on two large microarray datasets [18, 10], a recent protein interaction dataset [14] and a combination of interaction and microarray data sets, each of which is used for the modeling and prediction of over hundred classes from the GO Biological Process ontology. The results show that, compared to the base k-NN classifier, this incorporation produces more accurate predictions for many of the functional classes considered, and also that the classes benefitted most by this approach are those containing the fewest members. We also illustrate how the proposed framework can be used for integrating information in the entire GO Biological Process ontology to improve the accuracy of prediction over a set of target classes.

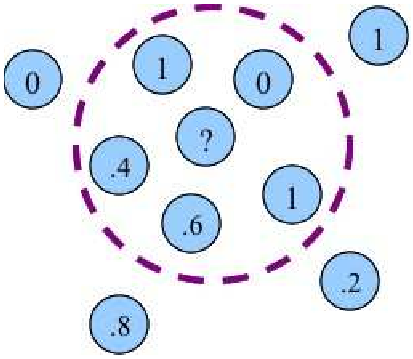


Figure 2: Overview of our approach. Solid circles represent data points and the values in them denote similarity of their class with the target class. The dashed circle represents the nearest neighborhood (derived from genomic data) of the query protein marked with a "?".

The rest of the paper is organized as follows. Section 2 discusses the motivation for incorporating the semantic similarity between functional classes for prediction algorithms further, and Section 3 discusses related work that has been done for incorporating functional relationships in GO into function prediction algorithms. Section 4 covers the necessary background for our approach, which is discussed in Section 5. Experimental results of the approach are presented in Section 6. Finally, we make conclusions and suggestions for future work in Section 7.

2. MOTIVATION FOR USING SEMANTIC SIMILARITY FOR FUNCTION PREDICTION

As has been mentioned before, there is ample evidence now that the similarity of biological characteristics of proteins implies the semantic similarity of their functional an-

[1] Since the rest of the discussion in this paper will be concerned with classification using GO, the terms (functional) *class*, *node* and *label* will be used interchangeably.

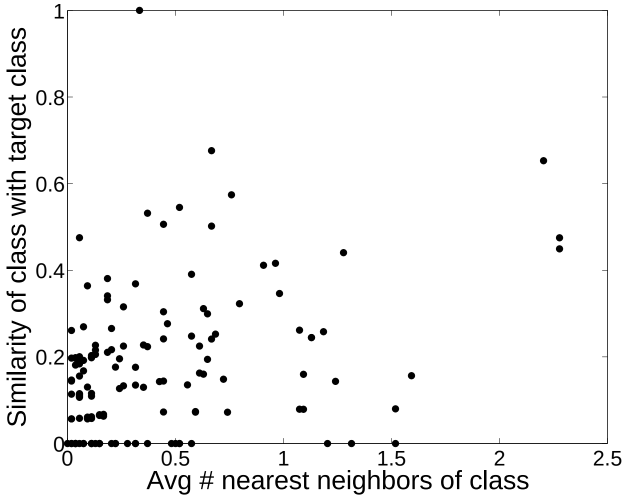


Figure 3: Distribution of nearest neighbors of similar classes for a class of medium size (54 proteins) in Mnaimneh *et al*'s data set

notations [17, 2, 27]. This result can be used to improve the prediction of protein function. Imagine a protein p, for which there is evidence that it performs function f. Now, it is possible that the proteins having expression profiles most similar to p may have labels that are similar to f, but not f itself. For instance, in Figure 2, p, which is the query protein shown by a ? symbol, has only two neighbors of the same target class, denoted by a semantic similarity of 1 to the target class. Due to this small number of similar proteins of the same class, the correct annotation for p may not be uncovered, using a direct neighborhood-based approach. This problem may become more severe in two circumstances, namely the class of interest being small, in which case there will not be sufficient number of similar proteins by definition, and in the case of noisy data, such as protein interaction networks and gene expression data, where the similarity between proteins may be affected by the noise in the data. Figure 1(a) shows this phenomenon for a large class in one of the data sets used in our study, for neighborhoods of size 20. This histogram shows that even for proteins annotated with this abundant label, the number of nearest neighbors carrying the same label are relatively few. For instance, the number of proteins having at most two neighbors carrying the same label is 243 respectively, which is over half of the class size. This behaviour is expected to be even more acute for smaller classes. For instance, Figure 1(b) shows this distribution for a relatively smaller class of size 54 and neighborhoods of size 20, and it can be seen that over 70% of the proteins in this class have no nearest neighbors of the same class. The distribution is similar for several other classes.

In these situations, it may be useful to consider the contributions from not only similar proteins having the label f, but also labels that are similar to f and are posessed by neighboring proteins. Figure 3 shows this phenomenon for the class discussed in Figure 1(b). Here, the semantic similarity of all the classes with the target class (calculated using Lin's measure [16] discussed Section 4) is plotted against the average number of nearest neighbors of the corresponding class in the nearest neighborhood of a protein of the target class. As can be seen from this scatter plot, even though the average frequency of the target class is very small (less than 0.5), there are several classes that are more abundant, and have a substantial semantic similarity with the target class (over 0.4 for several classes). This similarity can be used to enrich the available information in the neighborhood of proteins being tested for a target class. Our approach is based on this principle, and is discussed in detail in Section 5.

3. RELATED WORK

Recently, some approaches have been proposed to address the problem of incorporating inter-relationships between functional classes in GO into function prediction algorithms. These approaches can be categorized using the following two types of relationships between classes constituting the DAG-based functional hierarchies in GO:

- **Parent-child relationships**: The basic structure of the ontologies in GO is constructed from edges between parent and children terms. Some approaches have recently been proposed for incorporating these relationships in order to enforce the consistency required by these relationships, namely a gene annotated with a child node must be annotated with the parent node, into function prediction algorithms. Barutcuoglu *et al* [3] proposed a Bayesian network-based approach for this incorporation. In this work, they trained individual SVM classifiers on all the nodes of the hierarchy. Then, by constructing a Bayesian network using the structure of the ontology, the predictions of all the nodes were corrected iteratively in order to ensure consistency between parent-child annotations throughout the hierarchy, obtaining significant improvements over the individual classifiers. Carroll and Pavlovic [6] proposed a similar approach using probabilistic chain graphs for this problem. However, due to the limited evaluation experiments on small hierarchies, it is unclear how the performance of this approach would scale for a large set of classes from GO. Some other studies, such as Shahbaba and Neal (2006) [28], have also studied this problem, although their techniques are limited to tree-structured hierarchies.

- **Sibling and other distant relationships**: An effect of the structure of the ontologies in GO is the formation of sibling relationships between nodes that are children of the same parent. These relationships can be further generalized to extended family relationships, such as cousin and other relationships. King et al [13] approached the problem of incorporating these relationships into function prediction algorithms by predicting the functions of a protein using the patterns of annotations of other genes. More specifically, they construct models using the decision tree and Bayesian network methods for predicting the annotation of a certain gene with a given functional class, using the other GO term annotations as attributes. Tao et al [31] extended King et al's approach further by augmenting the prediction model with the semantic similarity between different classes. Here, they used Lin's similarity measure [16], also used in our study but using a different definition, to measure the inter-relationships between the functional classes in GO, and thus to measure the similarity between the sets of functional labels of two genes. This similarity measure is then used within the framework of a k-nearest neighbor classifier for predicting whether an unannotated gene belongs to a certain functional class or not. This study provided important evidence for the utility of semantic similarity as a method for measuring the degree of relationship between two classes, and using this measure for enhancing function prediction algorithms. However, since this technique uses the known annotations of a protein to predict its other potential annotations, it may be hard for it to predict functions for a gene or protein for which no annotations are known, since this gene will have no similarity to the other genes using this measure. This illustrates the utility of using external genomic data for predicting the functions of such poorly annotated proteins, since, in this approach, the similarity measurements between proteins are made on the basis of experimental data, the availability of which is not necessarily dependent on whether the protein is annotated or not. Our work takes this approach of augmenting biological data-based functional classification algorithms with inter-relationships between functional classes using measured using standard semantic similarity measures.

Incorporation of both these types of relationships are important for making use of the information available in the entire hierarchy. One of the advantages of the direct incorporation of functional relationships, which is the focus of the latter set of studies, is that it is possible to incorporate information from nodes farther away in the hierarchy, as compared to the hierarchical incorporation approaches, which only utilize the subgraph of the hierarchy corresponding to the set of target classes. Our work provides a framework for incorporating these distant functional inter-relationships into standard function prediction algorithms. Notably, this task is more challenging than the hierarchical incorporation problem due to two important reasons: (1) there are no widely accepted theoretical frameworks for a direct incorporation of relationships between labels into classification algorithms, while reasoning structures such as Bayesian networks and probabilistic chain graphs can be used for modeling hierarchical relationships, for which a clear structure is available in GO, and (2) there are a much larger number of relationships between labels to be considered than just the parent-child relationships between pairs of terms, which are relatively fewer. These factors make the incorporation of non-hierarchical relationships more challenging. Furthermore, as discussed above, we perform this augmentation using the biological characteristics of genes captured in high-throughput genomic data, thus addressing one of the limitations of King et al's [13] and Tao et al's [31] studies. This enables us to make predictions for poorly annotated genes, for which experimental data, such as their interactions and expression profiles, are available.

Our work is also related to the field of hierarchical and multi-label classification in machine learning and data mining [32]. Most of the work in this field has been done within the domains of text and image classification [9, 29, 25, 33]. However, this work has limited applicability to protein function prediction because (1) the input data is relatively easier to analyze than complex biological data, (2) the hierarchies are structured as trees rather than DAGs, which are considerably harder to incorporate due to the possibility of a child having multiple parents and (3) they consider no (or limited) interaction between the classes. Recently, some work has been done for incorporating relationships between labels into classification algorithms. In one such study, Kang *et al* [12] developed a k-NN model that takes into account the co-occurrence of the target labels. However, this co-occurrence measure may not be sufficient for our study, due to the presence of both large and small classes in the set of target classes. Also, this technique ignores the hierarchical arrangement of classes in functional classification schemes such as GO.

4. BACKGROUND

4.1 Semantic Similarity in an Ontology

In GO, nodes are connected to other nodes through parent-child edges, which imposes hierarchical inter-relationships between the nodes constituting an ontology. Also, the nodes contain member proteins that have been annotated with the corresponding functional class. Thus, it is possible to compute the similarity between two GO nodes, referred to as *semantic similarity*, on the basis of the hierarchy, the contents of the nodes, or a combination of both. Several information-theoretic semantic similarity measures have been developed for computing similarity between two concepts in a hierarchy, such as those by Lin [16], Resnik [23] and Jiang [11]. These measures evaluate the similarity of two nodes in terms of their proximity in the ontology, as well as their content. In particular, we use Lin's measure [16], defined in Equation 1.

$$linsim(l_1, l_2) = \frac{2 \times [\log p_{ms}(l_1, l_2)]}{\log p(l_1) + \log p(l_2)} \qquad (1)$$

Here, l_1 and l_2 are the labels (or nodes) between which similarity is being calculated, while $p(l)$ denotes the probability of a protein being annotated with label l, and is estimated from the available set of GO annotations for an organism. Also, $p_{ms}(l_1, l_2) = \underset{l \in S(l_1,l_2)}{min} p(l)$, where $S(l_1, l_2)$ is the set of common ancestors of l_1 and l_2. Thus, $p_{ms}(l_1, l_2)$ denotes the probability of the *minimum subsumer* of l_1 and l_2. Intuitively, Lin's measure measures the semantic similarity of l_1 and l_2 in terms of the contents of their minimum subsumer node in the ontology. Clearly, $linsim(l_1, l_2) = 1$ when $l_1 = l_2$, and $linsim(l_1, l_2) = 0$, when their minimum subsumer is the root of the ontology. An additional advantage of this measure is that they are bounded between $[0, 1]$. These fixed bounds of this measure are very useful for our implementation of the functional inter-relationship concept into prediction algorithms, as explained in Section 5.

An example of a label similarity matrix computed for the set of functional classes used in this study is shown in Figure 4(a).

4.2 k-Nearest Neighbor Classifier

One of the simplest classification algorithms is the k-nearest neighbor (k-NN) classifier [30], which is based on the principle of abundance of the target label in the neighborhood of the query example. We use a weighted variant of this classifier, similar to the direct k-NN classifier used by Kuramochi and Karypis [15], which counts the abundance of each label in the neighborhood of size k of a protein, weighted by the feature similarity of the neighboring proteins having the corresponding label. Thus, if the feature set of a protein p is denoted by $feature(p)$, then the likelihood score of a label l for a protein p is given by Equation 2.

$$likelihood(p,l) = \sum_{p' \in Nbd(p)} [sim(feature(p), feature(p')) \times I[l \in labels(p')]] \quad (2)$$

Here $sim(feature(p), feature(p'))$ denotes the similarity between the feature vectors describing proteins p and p', and I is an indicator function that returns 1 if l belongs to the set of labels p' is annotated with. Applying this formula for p for each label, and then repeating the calculation for all the proteins, produces a $|proteins| \times |labels|$ matrix, named LL_{basic}, of likelihood scores. The accuracy of this algorithm can then be evaluated using any threshold-free evaluation measure, which was chosen to be the area under the ROC curve (AUC score) [30] in our study.

We chose k-NN as the base classifier in our study since it is much simpler than other classification methods, such as SVM [30], and hence it is easier to incorporate additional factors into the model. Also, for the problem of protein function predictions, some authors have reported that with suitable parameter settings, k-NN produces comparable performance to SVM [15, 34, 35].

5. PROPOSED APPROACH

5.1 Modified classification algorithm

It can be observed from Equation 2 that k-NN is an additive model, *i.e.*, the likelihood scores are obtained by adding the contributions of all examples in the neighborhood of the test example. Thus, it is intuitively easy to incorporate contributions from examples annotated with similar labels. This is implemented using Equation 3.

$$likelihood(p,l) = \sum_{p' \in Nbd(p)} [sim(feature(p), feature(p')) \times (\sum_{l' \in labels(p')} LabelSim(l,l'))] \quad (3)$$

Equation 3 represents a direct extension of the model described in Equation 2, where, in addition to the label being tested (l), contributions are also taken from labels similar to l. The latter factor is incorporated into the model using the second term $\sum_{l' \in labels(p')} LabelSim(l,l')$, which denotes the sum of the similarities between the target label l and all the other labels l' posessed by p'. In fact, if LL_{basic} represents the $|proteins| \times |labels|$ likelihood matrix derived using the direct k-NN model, and $LabelSim$ is the matrix of pairwise label similarities, then the above equation can be written conveniently as follows, where $LL_{labelsim}$ contains the final likelihood scores.

$$LL_{labelsim} = LL_{basic} \times LabelSim \quad (4)$$

Equation 4 makes the implementation of our approach much easier. Also, it generalizes the implementation to other classifiers such as SVM, where the LL_{basic} matrix may simply contain the discriminant values produced by all SVM classifiers for all the classes, for each protein in the test set. However, care needs to be taken in applying this formula to all classifiers, since, unlike k-NN, they may not be additive in nature.

5.2 Filtering of label similarities

The label similarity matrix contains a value (however small or large) for each pair of labels. Many of these similarities, especially the smaller ones, are likely to be uninformative, since all functional classes are not expected to interact with all the others, particularly in a large diverse set of classes. In order to handle this issue, we used a heuristic approach for filtering the label similarity matrix. For each class, we determined a filtering threshold using a cross-validation procedure. This threshold was determined by running a grid search over the interval $[0, 1]$. For each such threshold t, all label similarities less than t were converted to 0. Then, a leave-one-out cross-validation procedure is run over the training set to determine the AUC score of the resulting label similarity-incorporated classifier. Finally, for each class, the threshold for which the highest AUC score is obtained, is chosen as the filtering threshold. A filtered version of the label similarity matrix shown in Figure 4(a) is shown in Figure 4(b).

6. EXPERIMENTAL EVALUATION

6.1 Data Sets

We used several high-throughput data sets for evaluating our approach. The first was Mnaimneh *et al*'s gene expression [18], which measures the expression of all *S. cerevisiae* (budding yeast) genes under a set of 215 titration experiments. The second was another large scale dataset known as the Rosetta gene expression compendium [10] prepared by subjecting yeast cells to a set of 300 diverse mutations and chemical treatments. Pearson's correlation coefficient, used commonly for measuring the similarity between the expression profiles of two genes [7], was used as the feature similarity function sim for these data sets. We also evaluated our approach on Krogan *et al*'s recently published data set of 7123 highly reliable physical interactions between proteins in yeast [14]. This data set was represented as an $n \times n$ adjacency matrix A, with the $A(i,j)$ cell containing the reliability of the interaction between proteins i and j, if any. We used the $h-confidence$ measure for measuring the similarity between the interaction profiles of two proteins in this matrix, which has been shown in a previous study [22] to handle the noise and incompleteness problems of protein interaction data robustly. Finally, we also considered a combined data set, which was prepared by combining the yeast protein interaction data in the BIOGRID database [4] with the two microarray datasets discussed above. This dataset was constructed by preparing the adjacency matrix for the BIOGRID interaction dataset, and concatenating the rows of this matrix with the gene expression profiles of the constituent genes. Also, any columns in the resultant data matrix that have less than two non-zero values are removed, since they do not contribute to the similarity computation. Finally, for this data set, we used the cosine similarity measure, since most of the data set is constituted by sparse interaction data.

Each of these data sets is used to construct classification models for a subset of 138 functional classes from the biological process ontology of GO, that have atleast 10 members in the corresponding data set. We chose these classes, since, using expert opinion, Myers *et al* [19] have estimated that the predictions made for these classes are likely to be testable in a wet lab and thus are of interest to biologists. Another

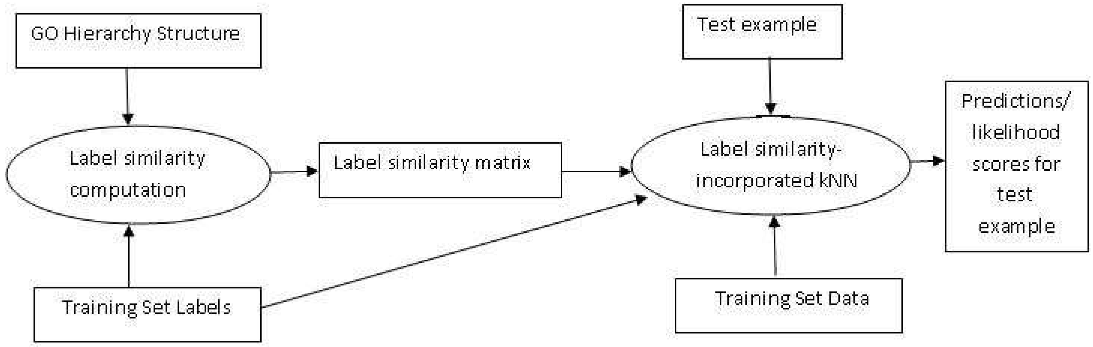
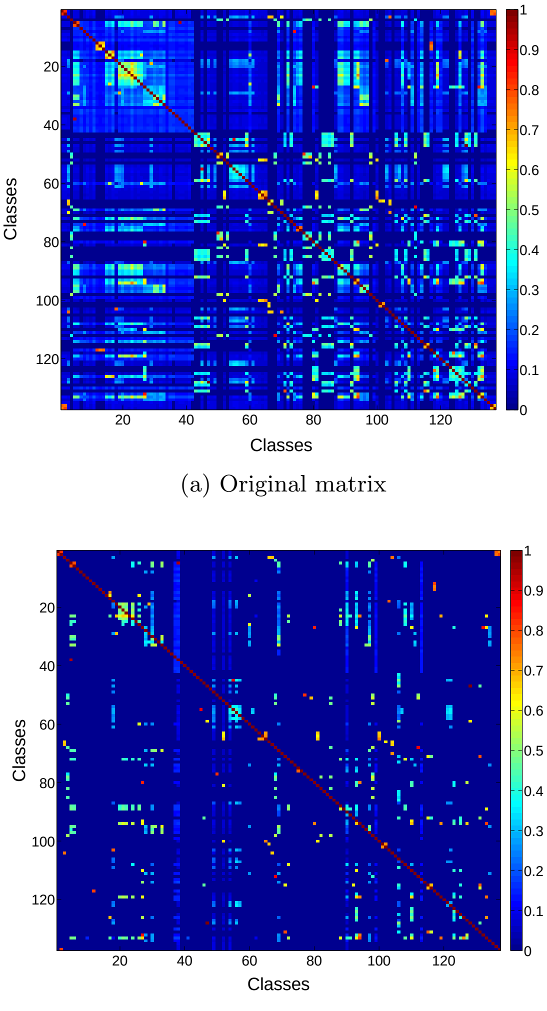

GO Hierarchy Structure
Label similarity computation
Label similarity matrix
Training Set Labels
Test example
Label similarity-incorporated kNN
Training Set Data
Predictions/ likelihood scores for test example

Figure 5: Overall methodology for classification using label similarity-incorporated k-NN



(a) Original matrix

(b) Filtered matrix

Figure 4: Original and filtered label similarity matrices generated using one of the training sets from Mnaimneh *et al*'s data set (best seen in color)

important reason for the choice of these classes is that no parent-child relationships exist between these classes, and thus it is difficult to use hierarchical relationship-based approaches for these classes. Also, these classes are spread throughout the ontology, and thus are suitable for illustrating the use of semantic similarity to improve predictions by incorporating information from several distant but related functional classes.

Table 1 shows the resultant number of proteins, features and classes used in each data set, as well as the value of k used in our classification studies (discussed below). Note that we limited the genes/proteins considered in each of these data sets to those annotated by at least one of the classes considered.

Dataset	# Proteins	# Features	# Classes	k
Mnaimneh	4062	215	137	20
Rosetta	3980	300	137	20
Krogan	2117	2117	108	5
Combined	3762	4277	136	10

Table 1: Details of data sets used for evaluation

6.2 Experimental Methodology

Our overall experimental methodology is shown in Figure 5. Below, we discuss the details of the individual components.

6.2.1 *Computation of label similarity matrix*

The first step of our experimental procedure is the construction of the similarity matrix between the labels, or the inter-relationship matrix between the corresponding classes, for each of the above data sets. For each set of labels, the original set of annotations are collected from the yeast GO annotations downloaded from the GO website in February 2000. Each of the annotations is then propagated up the biological process ontology, as per the parent-child relationships, and the label matrix corresponding to the classes used for each of the data sets is extracted from this complete annotation set. Finally, Equation 1 is applied to all pairs of labels in the original label set for each data set, in order to obtain the final label similarity matrix. Note that none of the test examples in a cross-validation setting is used for computing this matrix. An example of the matrix constructed from one training sets of Mnaimneh *et al*'s data set is shown in Figure 4(a).

6.2.2 *Classification and evaluation*

We followed a five-fold cross-validation procedure for our

Dataset	**Total # classes**	**# Classes improved**	**Average improvement over all classes**	**Maximum improvement**
Mnaimneh	137	74	0.0219 (3.57%)	0.1882 (39.92%)
Rosetta	137	47	0.0083 (1.33%)	0.2091 (38.66%)
Krogan	108	30	0.0045 (0.63%)	0.1982 (31.82%)
Combined	136	59	0.0079 (1.02%)	0.1129 (20.39%)

Table 2: Statistics about comparative performance of base k-NN classifiers and their functional similarity-incorporated versions

experiments. Here, for each protein p in the test set, Equation 3 is used to calculate the likelihood score for protein p for each class c. Repeating this process for each protein in each fold, using the other four folds as training sets, produces the global protein-label likelihood score matrix, which can then be evaluated by computing an AUC score for each label. Although the results reported in this section are based on five-fold cross validation, we obtained very similar results with other fold configurations also. Also, the values of k chosen for each data set, shown in the last column of Table 1, is chosen in accordance with the density or the sparsity of the corresponding data set. Thus, k is chosen to be high (20) for the dense microarray data sets, low (5) for the sparse protein interaction data set (Krogan), and an intermediate value (10) for the combined dataset constructed by combining both microarray and interaction data. However, we obtained similar results for other values of k.

An important intermediate step in our method is the filtering of the original label similarity matrix, which is implemented as explained in Section 5.2. The use of leave-one-out cross-validation for this procedure offers the important advantage of consistency of the filtering thresholds determined across different folds. Figure 4(b) shows a filtered version of the label similarity matrix shown in Figure 4(a). Clearly, the former is significantly sparser than the latter, and reflects the biological intuition that a functional class does not necessarily interact with all others.

6.3 Experimental Results

In this section, we compare the performance of the label similarity-incorporate classifier with the base k-NN classifier, and illustrate how the use of inter-relationships between classes can help improve the accuracy of predictions made over a set of target classes. Note that all the AUC scores presented in this section are obtained as the average of fifty five-fold cross validation runs of each classifier, unless otherwise stated.

6.3.1 Improvement of performance for a large set of classes

Table 2 lists specific comparative statistics about the AUC scores obtained for all the classes using the base k-NN classifiers and their label similarity-incorporated versions. As can be seen, a non-trivial improvement is observed in the average AUC score over all the classes for all the data sets, and the maximum improvement on one of the classes is usually very high. However, note that the incorporation of functional inter-relationships is not expected to produce the same degree of average improvement for all data sets, since for some of them, the base functional content may be high for most of the classes, and cases cited in Section 2 may not occur substantially, thus making it hard to improve their predictions using the label similarity approach. Also, it is hard to obtain substantial improvements over the large set of diverse classes used in our study.

We also examined the effect of our approach on the performance of classification for each class individually. Figure 6 shows the comparison of performance of individual base k-NN classifiers for each functional class, and their functional similarity-incorporated versions for Mnaimneh *et al*'s data set. More specifically, the AUCs of individual k-NN classifiers for each class is plotted on the x-axis, while those of the functional similarity-incorporated k-NNs are plotted on the y-axis. Thus, the points above the $y = x$ line indicate an improvement in the AUC score of the corresponding class, and vice versa. Using this interpretation, it is easy to see from this plot that the performance of a large fraction of the classes are improved by incorporating contributions from similar classes. Another encouraging aspect of this plot is that almost none of the classes suffers a major loss of prediction accuracy due to the incorporation of label-similarity, and in most cases, the difference can be accounted for by the effect of randomization in the cross-validation process. This implies that for those classes whose performance is invariant, the label similarity filtering process (discussed in Section 5.2) is able to infer that incorporating label similarity is not appropriate for these classes, and does not identify irrelevant relationships in the filtered label similarity matrix. This phenomenon is observed for the other data sets as well (data not shown due to lack of space).

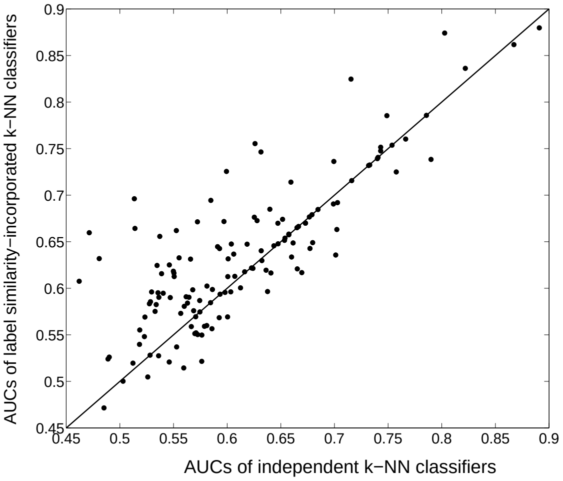


Figure 6: Comparison of the performance of hierarchy-based functional similarity-incorporated k-NN classifiers with individual k-NN classifiers for Mnaimneh *et al*'s data set

6.3.2 Improvement of performance for small classes

One of the primary motivations for the incorporation of label similarity into standard function prediction algorithms is to improve the prediction accuracy for data-poor classes, as discussed in Section 2. Our approach is expected to be useful for this task, as, in our model, the small classes can seek a contribution from classes of bigger sizes that have a high semantic similarity with them. To test this hypothesis, we selected classes of size at most 30 in all the data sets being used, and analyzed the results obtained using label similarity, as against those from basic classification. Table 3 provides detailed statistics about these results. Indeed, it can be seen from these results that the improvements, both in absolute terms and as a percentage of the average AUC score of the base classifiers, for these classes are significantly higher than the corresponding figures in Table 2. This shows that the label similarity-based classification approach is indeed able to help improve the accuracy of the predictions made over data-poor classes, for which it is hard to build very accurate base classifiers.

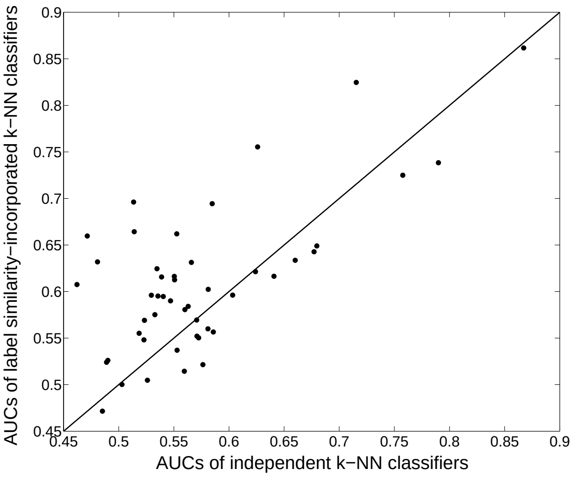


Figure 7: Comparison of the performance of hierarchy-based functional-similarity incorporated k-NN classifiers with individual k-NN classifiers for small classes in Mnaimneh *et al*'s data set

The class-by-class improvements for each small class in Mnaimneh *et al*'s data set is shown graphically in Figure 7. While the performance of some classes is invariant (close to the $y = x$ line), several classes show a large improvement in performance. In particular, we investigated the class GO:0051049 (regulation of transport), which has only 11 members in Mnaimneh *et al*'s data set, and shows the maximum improvement of almost 40%. In order to identify the classes that contributed to the improved performance of this class (besides itself), we identified the classes that had a non-zero semantic similarity with this class in the filtered label similarity matrix shown in Figure 4(b). Table 4 provides details of the eight classes so found.

GO Term	Definition	Size	Similarity
GO:0016192	Vesicle-mediated transport	328	0.4085
GO:0016458	Gene silencing	100	0.4067
GO:0016481	Negative regulation of transcription	163	0.4317
GO:0040029	Regulation of gene expression	100	0.4067
GO:0045184	Establishment of protein localization	273	0.4014
GO:0045941	Positive regulation of transcription	102	0.4143
GO:0051052	Regulation of DNA metabolic process	80	0.4058
GO:0051252	Regulation of RNA metabolism	383	0.4743

Table 4: Details of classes most similar to GO:0051049 (regulation of transport)

As can be observed from Table 4, all the classes contributing to the improvement of predictions made over this class are fairly large in size, and their high semantic similarity with the target class enables the label similarity-incorporated classifier to make use of the members of these classes to acquire more information about the data-poor class. Also, very interestingly, most of these classes are biologically related to the target class, since most of them are related to the processes of *transport* (vesicle-mediated transport and establishment of protein localization) and *regulation* (regulation of DNA metabolic process, regulation of RNA metabolism etc). Thus, it is useful to incorporate such relationships into the prediction process for small classes, such as the one discussed in this example.

This deeper analysis supports our hypothesis that the label similarity filtering process is able to capture the most meaningful relationships between functional classes in a given label similarity matrix, and the label similarity-incorporated classification process is able to utilize these relationships to improve the predictions over individual classes.

6.3.3 Incorporating information in the whole GO biological process ontology

Unlike the hierarchical incorporation approach discussed in Section 3, which focus on the subgraph of the functional hierarchy corresponding to the target classes, one of the advantages of the direct incorporation of relationships into the classification model is that relationships in the entire hierarchy can be incorporated into the classification model, while holding the set of target classes constant. This can be done by simply modifying the label similarity matrix to include the semantic similarities between the target classes and all the other classes in the hierarchy. Thus, instead of using an $n \times n$ matrix of similarities, one can use a $|l| \times |L|$ matrix, where $|l|$ is the number of target classes, and $|L|$ is the number of all the classes in the hierarchy. The rest of the approach, as shown in Figure 5, remains the same.

We tested this idea for Mnaimneh *et al*'s data set, using the GO biological process ontology as the source of all the functional inter-relationships. Here, the matrix of semantic similarities between the 137 target classes and all the 2395 non-empty terms in the biological process hierarchy is computed from each training set in the five-fold cross validation procedure. This matrix is then filtered using the training set itself as described in Section 5.2, and the used to modify the likelihood matrix produced by the base k-NN classifiers for each of the 137 classes. The results of this experiment,

Dataset	# Small classes	# Classes improved	Average improvement over all small classes	Maximum improvement
Mnaimneh	47	27	0.0358 (6.24%)	0.1882 (39.92%)
Rosetta	48	21	0.0225 (3.82%)	0.2091 (38.66%)
Krogan	40	14	0.0129 (1.89%)	0.1982 (31.82%)
Combined	48	28	0.0197 (2.72%)	0.1129 (20.39%)

Table 3: Statistics about comparative performance of base k-NN classifiers and their functional similarity-incorporated versions on small classes

	Total # classes	# Classes improved	Average improvement over all classes	Maximum improvement
All classes	137	71	0.0167 (2.65%)	0.2492 (53.93%)
Small classes	47	29	0.0363 (6.32%)	0.2492 (53.93%)

Table 5: Statistics about comparative performance of base k-NN classifiers and their functional similarity-incorporated versions using information in the whole GO biological process ontology

generated using ten rounds of five-fold cross validation, are summarized in Table 5, for all the classes and for the small classes.

These results are comparable to those obtained from Mnaimneh *et al*'s data set using only the target classes for identifying functional relationships. However, some results are improved when the whole hierarchy is used, namely the average and the maximum improvement over the small classes, showing once more that the small classes are able utilize the label similarity matrix more effectively. On the whole, these results indicate that while incorporating functional relationships across the whole hierarchy is a potentially useful method to make use of available information. However, it is important to carefully identify the relationships to be utilized due to the very large number (137 × 2395) of possible relationships, many of which are expected to be uninformative. This task may need a more sophisticated methodology than that used for only the target classes which had fewer (137 × 137) possible relationships.

In summary, these results show that the incorporation of direct relationships between functional classes constituting the GO functional hierarchies, measured using a suitable semantic similarity measure (Lin's measure [16] in this study) is a useful method for improving the accuracy of the predictions made over a set of target classes, particularly for classes with a small number of member genes.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we demonstrated the utility of idea of incorporating functional inter-relationships into protein function prediction algorithms, in order to improve the predictions made by them. We modeled these relationships using Lin's semantic similarity measure [16] and modified the commonly used k-nearest neighbor classification algorithm in order to seek contributions from other classes, weighted by their semantic similarity with the target class. Results on several large genomic data sets show that this approach is able to improve the results for a large majority of the classes considered. In particular, a bigger improvement is seen for smaller classes, which are otherwise harder to model and predict.

In future work, it will be useful to incorporate the concept of functional similarity into SVMs, which do not have the additive characteristic like k-nearest neighbor, and other function prediction algorithms, such as FunctionalFlow [20] for protein interaction networks, which are focused on specific types of data. Another important direction for our work will be to carefully analyze the relationships between a set of target classes with all the other classes in the hierarchy, in order to incorporate more information into the classifiers, while reducing the effect of spurious relationships. As noted in Section 3, it was noted that incorporating both parent-child and more distant relationships between classes into function prediction algorithms will be required for making optimal use of relationships constituting GO. For this, it will be useful to integrate our framework with the Bayesian network-based approach of Barutcuoglu *et al* [3] for enforcing parent-child consistency between the results of standard prediction algorithms. As an example of a possible methodology of integrating these approaches, distant functional relationships could be incorporated first using our technique, and then the resulting labels could be propagated hierarchically using the Bayesian network approach. Investigation of such schemes will be a topic of future research.

8. ACKNOWLEDGEMENTS

We are thankful to Zafer Barutcuoglu, Yves Lussier and Jianrong Li for several fruitful discussions, and for providing us with useful information and data for the project. We also thank Gang Fang for his technical help and discussions.

9. REFERENCES

[1] M. Ashburner et al. Gene Ontology: tool for the unification of biology. *Nature Genetics*, 25(1):25–29, 2000.

[2] F. Azuaje and O. Bodenreider. Incorporating ontology-driven similarity knowledge into functional genomics: An exploratory study. In *Proc. BIBE*, pages 317–324, 2004.

[3] Z. Barutcuoglu, R. E. Schapire, and O. G. Troyanskaya. Hierarchical multi-label prediction of gene function. *Bioinformatics*, 22(7):830–836, 2006.

[4] B.-J. Breitkreutz, C. Stark, and M. Tyers. The GRID: the General Repository for Interaction Datasets. *Genome Biology*, 4(3):R23, 2003.

[5] M. P. Brown, W. N. Grundy, D. Lin, N. Cristiniani, C. W. Sugnet, T. S. Furey, M. Ares, Jr., and

D. Haussler. Knowledge based analysis of microarray gene expression data by using support vector machines. *Proc Natl Acad Sci*, 97(1):262–267, 2000.

[6] S. Carroll and V. Pavlovic. Protein classification using probabilistic chain graphs and the gene ontology structure. *Bioinformatics*, 22(15):1871–1878, 2006.

[7] P. D'haeseleer. How does gene expression clustering work? *Nat Biotech*, 23:1499–1501, 2005.

[8] T. Gabaldon and M. A. Huynen. Prediction of protein function and pathways in the genome era. *Cell Mol Life Sci.*, 61(7–8):930–944, 2004.

[9] S. Godbole and S. Sarawagi. Discriminative methods for multi-labeled classification. In *Proceedings of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2004)*, pages 22–30, 2004.

[10] T. R. Hughes, M. J. Marton, A. R. Jones, C. J. Roberts, R. Stoughton, C. D. Armour, H. A. Bennett, E. Coffey, H. Dai, Y. D. He, M. J. Kidd, M. M. King, M. R. Meyer, D. Slade, P. Y. Lum, S. B. Stepaniants, D. D. Shoemaker, D. Gachotte, K. Chakraburtty, J. Simon, M. Bard, and S. H. Friend. Functional discovery via a compendium of expression profiles. *Cell*, 102(1):109–126, 2000.

[11] J. J. Jiang and D. W. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proc. Intl. Conf. Res. on Comp. Linguistics*, 1998.

[12] F. Kang, R. Jin, and R. Sukthankar. Correlated label propagation with application to multi-label learning. pages 1719–1726, 2006.

[13] O. D. King, R. E. Foulger, S. S. Dwight, J. V. White, and F. P. Roth. Predicting gene function from patterns of annotation. *Genome Research*, 13(5), 2003.

[14] N. J. Krogan, G. Cagney, H. Yu, G. Zhong, X. Guo, et al. Global landscape of protein complexes in the yeast saccharomyces cerevisiae. *Nature*, 440:637–643, 2006.

[15] M. Kuramochi and G. Karypis. Gene classification using expression profiles: A feasibility study. *Intl. J. on Artificial Intelligence Tools*, 14(4):641–660, 2005.

[16] D. Lin. An information-theoretic definition of similarity. In *Proc. International Conference on Machine Learning*, pages 296–304, 1998.

[17] P. W. Lord, R. D. Stevens, A. Brass, and C. A. Goble. Semantic similarity measures as tools for exploring the gene ontology. In *Pacific Symposium on Biocomputing*, pages 601–612, 2003.

[18] S. Mnaimneh et al. Exploration of essential gene functions via titratable promoter alleles. *Cell*, 118(1):31–44, 2004.

[19] C. L. Myers et al. Finding function: evaluation methods for functional genomic data. *BMC Genomics*, 7:187, 2006.

[20] E. Nabieva, K. Jim, A. Agarwal, B. Chazelle, and M. Singh. Whole-proteome prediction of protein function via graph-theoretic analysis of interaction maps. *Bioinformatics*, 21(Suppl. 1):i1–i9, 2005.

[21] G. Pandey, V. Kumar, and M. Steinbach. Computational approaches for protein function prediction: A survey. Technical Report 06-028, Department of Computer Science and Engineering, University of Minnesota, Twin Cities, 2006.

[22] G. Pandey, M. Steinbach, R. Gupta, T. Garg, and V. Kumar. Association analysis-based transformations for protein interaction networks: a function prediction case study. In *KDD '07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 540–549, 2007.

[23] P. Resnik. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *JAIR*, 11:95–130, 1999.

[24] A. Ruepp, A. Zollner, D. Maier, K. Albermann, J. Hani, M. Mokrejs, I. Tetko, U. Guldener, G. Mannhaupt, M. Munsterkotter, and H. W. Mewes. The FunCat, a functional annotation scheme for systematic classification of proteins from whole genomes. *Nucleic Acids Research*, 32(18):5539–5545, 2004.

[25] R. E. Schapire and Y. Singer. Boostexter: A boosting-based systemfor text categorization. *Mach. Learn.*, 39(2-3):135–168, 2000.

[26] A. S. N. Seshasayee and M. M. Babu. Contextual inference of protein function. In S. Subramaniam, editor, *Encyclopaedia of Genetics and Genomics and Proteomics and Bioinformatics*. John Wiley and Sons, 2005.

[27] J. L. Sevilla et al. Correlation between gene expression and go semantic similarity. *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, 2(4):330–338, 2005.

[28] B. Shahbaba and R. M. Neal. Gene function classification using bayesian models with hierarchy-based priors. *BMC Bioinformatics*, 7:448, 2006.

[29] X. Shen, M. Boutell, J. Luo, and C. Brown. Multi-label machine learning and its application to semantic scene classification. In *International Symposium on Electronic Imaging*, San Jose, CA, January 2004.

[30] P.-N. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining*. 2005.

[31] Y. Tao, L. Sam, J. Li, C. Friedman, and Y. A. Lussier. Information theory applied to the sparse gene ontology annotation network to predict novel gene function. *Bioinformatics*, 23(13):i529–i538, 2007.

[32] G. Tsoumakas and I. Katakis. Multi label classification: An overview. *International Journal of Data Warehousing and Mining*, 3(3):1–13, 2007.

[33] A. Veloso, M. J. Wagner, M. Goncalves, and M. Zaki. Multi-label lazy associative classification. In *Knowledge Discovery in Databases: PKDD 2007. 11th European Conference on Principles and Practice of Knowledge Discovery in Databases, volume LNAI 4702*, 2007.

[34] C. Wang and S. D. Scott. New kernels for protein structural motif discovery and function classification. In *Proc. 22nd Intl Conf on Machine Learning (ICML)*, pages 940–947, 2005.

[35] Z. Yao and W. L. Ruzzo. A regression-based k nearest neighbor algorithm for gene function prediction from heterogeneous data. *BMC Bioinformatics*, Suppl 1:S11, 2006.