Protein Function Prediction with Incomplete Annotations

Guoxian Yu, Huzefa Rangwala, Carlotta Domeniconi, Guoji Zhang and Zhiwen Yu, *Member, IEEE*

Abstract—Automated protein function prediction is one of the grand challenges in computational biology. Multi-label learning is widely used to predict functions of proteins. Most of multi-label learning methods make prediction for unlabeled proteins under the assumption that the labeled proteins are completely annotated, i.e., without any missing functions. However, in practice, we may have a subset of the ground-truth functions for a protein, and whether the protein has other functions is unknown. To predict protein functions with incomplete annotations, we propose a *Pro*tein Function Prediction method with *W*eak-label *L*earning (ProWL) and its variant ProWL-IF. Both ProWL and ProWL-IF can replenish the missing functions of proteins. In addition, ProWL-IF makes use of the knowledge that a protein cannot have certain functions, which can further boost the performance of protein function prediction. Our experimental results on protein-protein interaction networks and gene expression benchmarks validate the effectiveness of both ProWL and ProWL-IF.

Index Terms—Protein Function Prediction, Multi-label Learning, Incomplete Annotations

1 Introduction

Advanced biological techniques have generated various high-throughput proteomic data, i.e., protein-protein interaction networks and protein structures. However, the functions of these proteins, which is of great importance to the investigation of the life process, are not well studied. As such, predicting the biological functions of proteins is one of the fundamental issues in the post-genomic era [1], [2], [3]. The financial and time costs associated with biological experiments to annotate these proteins are quite demanding. The availability of various proteomic data and function annotation approaches allows for automatic protein function prediction, which can often guide the follow-up biological hypothesis and experiments. For these reasons, it is critical to develop computational methods to predict protein functions.

The recent availability of various proteomic data have led to the development of various computational methods for inferring protein functions. Several of these approaches take advantage of kernel functions to capture the similarity between gene expression sequences and employ kernel-based classifiers to predict protein functions [4], [5]. Some methods use PPI and graph- (or network-) based classifiers to predict the functions of proteins [1], [6], [7], [8], [9]. Several approaches predict protein functions by using heterogeneous data sources (including amino acid sequences and PPI) [2], [10], [11], [12].

Traditional protein function prediction models [5], [10] often neglect the fact that biological functions are correlated with each other [13]. Multi-label learning approaches [14], [15] use function correlation to boost the accuracy of protein function prediction [4], [14], [16] and can assign more than one function to a protein. Some protein function prediction methods incorporate the correlations between the functions (labels) to improve the multi-label prediction accuracy [7], [12], [16], [17], [18]. In particular, some approaches first train binary classifiers for each functional label and then utilize the hierarchical structure [19], [20], [21] prevalent within the underlying protein function databases (e.g., Function Catalogue [22] or Gene Ontology [23]). In this paper, we focus on protein function prediction using multi-label learning and function correlation.

All these methods predict the functions of proteins under the assumption that the functions of labeled proteins are complete, i.e. there are no missing labels. In contrast, in practice we just know a subset of the functions of a protein, and whether this protein has additional functions is unknown. Namely, these proteins have an incomplete annotation [2]. This kind of multi-label learning problem is referred to as the 'weak label' or 'incomplete class assignment' problem [24], [25]. Unlike, traditional multi-label learning methods [7], [16], [19], [18], which predict protein function under the assumption that the annotated functions of proteins in the training set are complete,

G. Yu is with the College of Computer and Information Science, Southwest University, Chongqing, 410075 China, and School of Computer Science and Engineering, South China University of Technology, Guangzhou, 510006 China, email: gxyu@swu.edu.cn

H. Rangwala and C. Domeniconi are with the Department of Computer Science, George Mason University, Fairfax, VA, 22030 USA, email: rangwala@cs.gmu.edu, carlotta@cs.gmu.edu

G. Zhang is with the School of Sciences, South China University of Technology, Guangzhou, 510640 China, email: magjzh@scut.edu.cn

Z. Yu is with the School of Computer Science and Engineering, South China University of Technology, Guangzhou, 510006 China, email: zhwyu@scut.edu.cn

Manuscript received 24 Mar. 2013; revised 16 Oct. 2013;

we develop a method, called *Pro*tein Function Prediction with *W*eak-label *L*earning (ProWL), which can replenish the missing functions on the incomplete annotated proteins in the training set, and also predict functions on the completely unlabeled proteins.

Sun *et al.* [24] and Bucak *et al.* [25] performed multi-label learning with weak labels by taking the currently specified labels of an instance as relevant labels, and all the unspecified labels (missing labels and irrelevant labels) of the instance as candidates for relevant labels. In practice, we may also know that a protein cannot have certain functions (hereinafter, we call these functions *irrelevant functions*). Previous weak-label learning approaches [24], [25] ignore this prior knowledge, which can often boost the performance of protein function prediction. To take advantage of these irrelevant functions, we propose a variation of ProWL, called *Pro*tein Function Prediction with *W*eak-label *L*earning and Knowledge of *I*rrelevant *F*unction (ProWL-IF). ProWL-IF can not only make use of the relevant functions of a protein, but also of the irrelevant ones to replenish the missing functions of a protein.

This work presented here is an extension of our earlier paper [26]. In particular, the additional contributions of this paper are as follows:

1) We provide motivations and an analysis of the proposed approaches.
2) We investigate the benefit of using the guilt by association rule and function correlation independently, along with an empirical study.
3) We compare the proposed methods against other related techniques, namely two multi-label weak-label learning methods and two multi-label protein function prediction approaches, using various metrics on public available protein datasets, and show their effectiveness.

The rest of the paper is organized as follows. In Section 2, we review related work on multi-label learning for predicting protein function and weak label learning approaches. In Section 3, we introduce ProWL and its variation ProWL-IF. Section 4 details the experimental protocol and Section 5 discusses the empirical results. In Section 6, we provide conclusions.

2 Related work

2.1 Graph-based Protein Function Prediction

Since our proposed approaches are graph- (or network-) based methods, we review some graph-based protein function prediction methods using PPI networks. Schwikowski *et al.* [27] determined the putative functions of a protein from the known function of its neighbors in PPI networks. Vazquez *et al.* [28] predicted protein functions by minimizing the number of protein-protein interactions among different functional categories and exploiting the global connectivity pattern of the protein network to predict protein function globally. Chua *et al.* [9] observed that indirectly interacting proteins also shared a few functions and extended the PPI network by setting different weights between level-1 and level-2 neighbors. Although these methods can assign more than one function to a protein by thresholding, they do not take into account the function correlation explicitly. For more information on network-based protein function prediction, one can refer to a comprehensive survey by Sharan *et al.* [1].

Proteins have multiple roles and functions. Each function can be viewed as a label. Thus, various multi-label learning approaches based on PPIs have been developed to automatically annotate proteins [14], [16], [17]. Pandey *et al.* [17] used Lin's measure [29] to compute the correlation between different functions (or GO terms) and incorporated the function correlation into a weighted k-nearest neighbor classifier to predict protein functions. Jiang *et al.* [18] proposed a product graph to incorporate pairwise function correlation in the label propagation framework. The adjacent matrix associated with this product graph is $(N \times K) \times (N \times K)$ (N is the number of proteins and K is the number of distinct functions). Given, the size of the product graph, it is computationally expensive to conduct label propagation on this graph. To overcome this limitation, Jiang [6] employed a bi-relation graph [30] and network propagation to predict protein functions. In the bi-relation graph, both proteins and functions are viewed as nodes, and three kinds of edges are defined, namely edges between proteins (exploiting the protein similarity), edges between functions (using function correlations) and edges between function nodes and protein nodes (function annotations). To avoid the risk of functions being overwritten (or missing) in the bi-relation graph, Yu *et al.* [12] proposed the directed bi-relation graph and applied a random walk with restart [31] on this graph to predict protein functions. Zhang *et al.* [16] used Jaccard coefficients to measure function correlations between different functions and then predicted protein function under a graph-based semi-supervised learning framework [32]. Wang *et al.* [7] used the Green function [33] to incorporate the function-function correlations based on the theory of reproducing kernel Hilbert space (RKHS), and proposed a method called Function-function Correlated Multi-Label (FCML) to infer protein functions. Bogdanov *et al.* [34] developed an approach that utilized the network structure for extracting features for predicting protein function. Mitrofanova *et al.* [35] took advantage of relationships between homologous proteins to connect networks of two (or more) different (but related) species for protein function prediction. Re *et al.* [36] developed an efficient ranking based prediction model using local and global learning strategies. Chi *et al.* [37] proposed an iterative protein function prediction method called Cosine Iterative Algorithm (CIA). CIA increases the number of predicted functions on unlabeled proteins iteratively. At each iteration, the most confident predicted functions on the unlabeled proteins are appended as relevant functions, and the pairwise similarities between training and testing proteins are updated using the functions belonging to these two sets of proteins. This updated similarity, together with the PPI network structure and the function correlation term, are used for predicting functions on the test proteins in the next iteration.

2.2 Weak-label Learning.

Prediction of the complete set of labels (i.e., predicting the missing labels), given partial or incomplete labels is defined as the *weak-label learning* problem. Most multi-label learning approaches focus on exploiting the label correlation to boost learning results, under the assumption that the given labels for the training instances are complete and accurate [14], [16]. However, in several cases, a multi-label instance often has only a subset of the ground-truth labels. In this scenario, given an annotated instance, it is unknown whether the annotations are complete or partial. Some approaches developed to replenish the missing (or noisy) labels in the single label case [38], and few methods are developed for multi-label learning scenarios.

Sun *et al.* [24] studied the weak-label learning problem in multi-label learning and proposed a method called WEak Label Learning (WELL). WELL considers the fact that classification boundary for each label should go across low density regions, and any given label will not be associated to the majority of instances. Based on these two assumptions, WELL solves this problem using convex optimization. In order to utilize the label correlation, WELL assumes that there is a group of low-rank based similarities, and the appropriate similarities between instances for different labels can be derived from these base similarities. However, WELL depends on quadratic programming to obtain the low-rank based similarities and to do the final prediction. Therefore, it has a large time complexity and computational load. This approach is only capable of replenishing the missing labels of partially labeled instances and can not be applied to a large number of proteins with a large number of functions. Buncak *et al.* [25] studied the incomplete class assignment problem for annotating images, and developed an approach called MLR-GR. MLR-GR optimizes the ranking errors and group Lasso loss in a convex optimization form. MLR-GL is useful for only predicting unlabeled multi-label instances using partially labeled instances. Qi *et al.* [39] used the Hierarchical Dirichlet Process to append missing labels for a set of images. In addition, Wang *et al.* [38] developed an approach for annotating weakly labeled facial images. However, this approach is a single-label (or multi-class) method and focuses on refining the noisy labeled images.

We develop a new weak-label learning algorithm for predicting multiple functions (or labels) of proteins by making use of the guilt by association rule [27] and function correlations. We refer to our approach as ProWL, *Pro*tein function prediction with *W*eak-label *L*earning. We extend ProWL to incorporate irrelevant functions (or labels) information of proteins and call the resulting approach ProWL-IF. Different from WELL and MLR-GL, the proposed ProWL and ProWL-IF can replenish the missing functions and make prediction on unlabeled proteins using partially labeled proteins. In addition, ProWL-IF makes additional use of irrelevant functions, which is rarely studied in previous weak label learning. Further, our empirical study shows that ProWL performs better than WELL and MLR-GL in both these two tasks.

3 Problem Formulation

We study the weak-label problem in protein function prediction for two tasks as illustrated in Figure 1. In these figures, each row denotes the function annotation for a protein, and each column corresponds to a function label. Fig. 1(a) depicts the complete annotated proteins, with 1 and 0 representing function annotations ($f1$ - $f4$) on the six proteins $p1$ - $p6$. In Fig. 1(b), 1 represents the known relevant functions, ? in the *color boxes* denote the missing functions and will be set to 0s, all the 0s serve as candidates for being predicted as relevant, i.e., ProWL may change a 0 to 1. In Fig. 1(c), 1 and -1 represent the relevant (1) and irrelevant (-1) known functions, ? in the color boxes denote the missing functions, which are set to 0s. The goal of ProWL-IF is to predict the missing functions (0) as relevant (1) or irrelevant (-1), respectively.

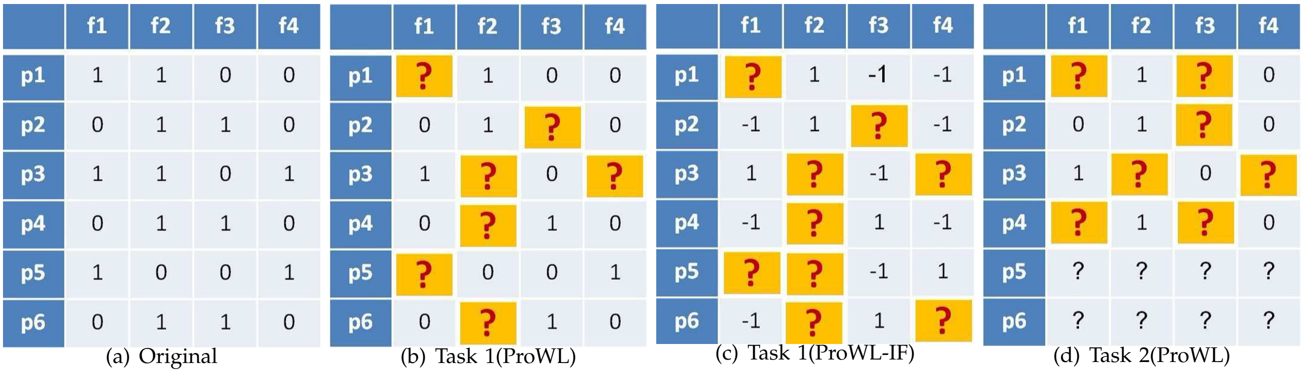
In Task 2 (c.f. Fig. 1(d)), the definition of 1 and 0 are the same as in Fig. 1(b). However, the target of ProWL is to use the incomplete annotated proteins ($p1$ - $p4$) to predict the functions of proteins $p5$ and $p6$, which are completely unannotated.

3.1 Protein Function Prediction with Weak-label Learning

It is important to make use of function correlations when annotating proteins [18], [16]. Given n proteins, let K be the number of distinct functions across all proteins. Let $Y = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n]$ be the original label set, with $y_{ik} = 1$ if protein i has the k-th function, and $y_{ik} = 0$ otherwise. At first, we can define a function correlation matrix $C' \in R^{K \times K}$ based on cosine similarity as follows:

$$C'_{st} = \frac{\mathbf{Y}_{.s}^T \mathbf{Y}_{.t}}{\|\mathbf{Y}_{.s}\| \|\mathbf{Y}_{.t}\|} \tag{1}$$

where C'_{st} is the function correlation between functions s and t, and $\mathbf{Y}_{.s}$ represents the s-th column of



	f1	f2	f3	f4
p1	1	1	0	0
p2	0	1	1	0
p3	1	1	0	1
p4	0	1	1	0
p5	1	0	0	1
p6	0	1	1	0

(a) Original

	f1	f2	f3	f4
p1	?	1	0	0
p2	0	1	?	0
p3	1	?	0	?
p4	0	?	1	0
p5	?	0	0	1
p6	0	?	1	0

(b) Task 1(ProWL)

	f1	f2	f3	f4
p1	?	1	-1	-1
p2	-1	1	?	-1
p3	1	?	-1	?
p4	-1	?	1	-1
p5	?	?	-1	1
p6	-1	?	1	?

(c) Task 1(ProWL-IF)

	f1	f2	f3	f4
p1	?	1	?	0
p2	0	1	?	0
p3	1	?	0	?
p4	?	1	?	0
p5	?	?	?	?
p6	?	?	?	?

(d) Task 2(ProWL)

Fig. 1. Task Overview: “1” stands for relevant function, “?” in a colored box stands for missing function and will be transformed to a “0”, “-1” stands for irrelevant function, $p5$ and $p6$ in the figure 1(d) are completely unlabeled.

Y. There are many other ways to define the function correlations, e.g., Jaccard coefficient [16] and Lin’s similarity [29]. Here, we use the cosine similarity for its simplicity and wide use [6], [7], [37]. Note that Eq. (1) can also be used with probabilistic function assignment. From Eq. (1), we can observe that if a large set of proteins share functions s and t but a small (or no) set of proteins share functions s and u, then $C^{'}_{st}$ will be greater than $C^{'}_{su}$. We normalize $C^{'}$ as follows:

$$C_{st} = \frac{C^{'}_{st}}{\sum_{k=1}^{K} C^{'}_{sk}} \tag{2}$$

Thus, C_{st} can be viewed as the likelihood that a protein has function t given that it is annotated with function s.

We now consider the case with incomplete annotation, and define the weighted loss function as the first part of our objective function as follows:

$$\Phi_1(\mathbf{f}) = \frac{1}{2}\sum_{i=1}^{n}\sum_{k=1}^{K} M_{ik}(f_{ik} - \tilde{y}_{ik})^2 \tag{3}$$

where $\tilde{Y} = [\tilde{\mathbf{y}}_1, \tilde{\mathbf{y}}_2, \ldots, \tilde{\mathbf{y}}_n]$ is the extended function set of n proteins, with $\tilde{Y} = YC$. $\tilde{y}_{ik}$ is the extended function assignments of protein i with respect to the k-th function, and f_{ik} is the predicted likelihood of protein i with respect to the k-th function. Our motivation in using $\tilde{Y}$ is to append the missing functions based on the labeled ones and function correlations. Specifically, suppose the currently confirmed functions Y_i for the i-th protein have a large correlation with the k-th function (which may be missing), then it is likely that this protein will also have function k. M_{ik} is the weight of protein i with respect to function k:

$$M_{ik} = \left\{ \begin{array}{l} 1, \quad y_{ik} = 1 \\ \mathbf{y}_i^T \mathbf{c}_{.k}, y_{ik} = 0 \end{array} \right. \tag{4}$$

where $\mathbf{c}_{.k}$ is the k-th column of C. As defined in Eq. (4), if the annotated functions of protein i have large correlations with function k, the weight M_{ik} will be large, since protein i is likely to also have function k.

A protein can have multiple functions, so the overlap between the function sets of two proteins can be used to measure their similarity, the more function they share, the more similar they are. This idea is also used in Chi *et al.* [37] and Wang *et al.* [7]. Thus we can use the function set of a protein to enrich its feature representation. We define the function induced graph W^f as:

$$W_{ij}^f = \frac{\mathbf{y}_i^T \mathbf{y}_j}{\|\mathbf{y}_i\|\|\mathbf{y}_j\|} \tag{5}$$

Note that an element in W^f describes the pairwise similarity between proteins induced from function annotations, whereas the element in C (in Eq. (2)) measures the pairwise correlation between functions.

The composite similarity matrix W between pairwise proteins can be defined as:

$$W = W^p + \gamma W^f \tag{6}$$

where W^p captures the *feature* (or biological) induced similarity between pairwise proteins. The matrix W^p can be set to the pairwise sequence similarities, frequency of interactions found in multiple PPI studies, or to a kernel matrix derived from PPI studies. γ is a parameter to balance the importance of the protein similarity graph W^p and the function induced similarity graph W^f, and it is often set as $\gamma = \sum_{i=1,j=1}^{N,N} W_{ij}^p / \sum_{i=1,j=1}^{N,N} W_{ij}^f$. Our empirical study shows that label propagation on W can achieve better performance than on sole W^p or W^f.

Proteins with similar amino acid sequences tend to have similar functions, and the ‘guilt by association’ rule [27] assumes that interacting proteins are more likely to share similar functions. To make use of this knowledge, as in label propagation [40], we incorporate a smoothness term within our objective function:

$$\begin{aligned} \Phi_2(\mathbf{f}) &= \frac{1}{2}\sum_{i,j=1}^{n} \| \frac{\mathbf{f}_i}{\sqrt{D_{ii}}} - \frac{\mathbf{f}_j}{\sqrt{D_{jj}}} \|^2 W_{ij} \\ &= tr(F^T(I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}})F) \\ &= tr(F^T L F) \end{aligned} \tag{7}$$

where $F = [\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_n]$, and D is a diagonal matrix with $D_{ii} = \sum_{j=1}^{n} W_{ij}$. I is an $n \times n$ identity matrix, $L = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$, and $tr(\cdot)$ is the matrix trace operation. By minimizing Eq. (7), the function annotations can be propagated from labeled proteins to unlabeled proteins.

Our objective function to be minimized is provided by:

$$\begin{aligned} \Phi(F) &= \frac{1}{2} \sum_{i=1}^{n} \sum_{k=1}^{K} M_{ik} (f_{ik} - \tilde{y}_{ik})^2 \\ &\quad + \alpha tr(F^T L F) + \beta \|F^T F\| \\ &= \frac{1}{2} \|M \circ (F - \tilde{Y})^T (F - \tilde{Y})\| \\ &\quad + \alpha tr(F^T L F) + \beta \|F^T F\| \end{aligned} \tag{8}$$

where $\circ$ denotes element-wise multiplication (also called Hadamard product). The third term is to control the sparsity of F, since each function is associated with a small number of proteins. α and β are parameters to balance the importance of the second and third terms, respectively. The motivation to minimize Eq. (8) is that we want to seek the prediction that is not only smooth and sparse, but can also append the missing annotations for proteins.

Eq. (8) coherently uses the first and second term to replenish (or predict) the missing functions for proteins in the training (or testing) set. Particularly, the first term uses $\tilde{y}_{ik}$ and M_{ik} to replenish the missing functions of partially annotated proteins in the training set, and the second term propagates the function annotations between proteins in the training set, it also can replenish the missing functions in some extent. For example, if training protein i has function k missing, all its interacting proteins annotated with function k, then this protein is likely to be annotated with function k. In addition, the second term can propagate the function annotations (including the replenished ones) on the training proteins to testing proteins.

Taking the derivative of Eq. (8) with respect to F, we have:

$$\frac{\partial \Phi(F)}{\partial F} = M \circ (F - \tilde{Y}) + \alpha L F + \beta I F \tag{9}$$

Eq. (9) can be divided into K problems and for the k-th problem it can be solved as:

$$(\tilde{M}_{\cdot k} + \alpha L + \beta I) \mathbf{f}_{\cdot k} = \mathbf{p}_k \tag{10}$$

where

$$\tilde{M}_{\cdot k} = diag(\mathbf{M}_{\cdot k}), \mathbf{p}_k = \mathbf{M}_{\cdot k} \circ \tilde{\mathbf{Y}}_{\cdot k} \tag{11}$$

$diag(\cdot)$ is the vector diagonalization operation. Instead of computing the inverse of $(\tilde{M}_{\cdot k} + \alpha L + \beta I)$, Eq. (10) can be solved with various existing fast iterative solvers [41]. We use the Conjugate Gradient (CG) solver, which is guaranteed to terminate in n steps. The most time-consuming step at each iteration of CG is a matrix vector product. The time complexity is proportional to the number of non-zero elements in $\tilde{M}_{\cdot k} + \alpha L + \beta I$. Since $\tilde{M}_{\cdot k}$, L and I are sparse, positive definite, and with $O(n)$ non-zero elements, Eq. (10) can be solved efficiently. In our experimental evaluation, we find that CG terminates in fewer than 30 iterations. The ProWL algorithm is described in **Algorithm 1**.

Algorithm 1 ProWL: *Pro*tein Function Prediction with *W*eak-label *L*earning

```
Input:
    Weight matrix $W^p$, incomplete annotations $Y = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n]$, $\alpha$, $\beta$
Output:
    Predicted likelihood score vectors $\{\mathbf{f}_i\}_{i=1}^{n}$
Compute $C$ using Eq. (2) and $W^f$ using Eq. (5).
Compute $W$ using $W^f$ and $W^p$, and $L = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$.
Set $\tilde{Y} = YC$ and initialize $M$ using Eq. (4).
for $k = 1$ to $K$ do
    Set $\tilde{M}_{\cdot k}$ and $\mathbf{p}_k$ using Eq. (11).
    Solve $\mathbf{f}_{\cdot k}$ using Eq. (10)
end for
return $F = [\mathbf{f}_{\cdot 1}, \mathbf{f}_{\cdot 2}, \dots, \mathbf{f}_{\cdot K}]^T$.
```

3.2 Protein Function Prediction with Weak-label Learning and Knowledge of Irrelevant Functions

In practice, we may know that some functions are *not* associated with specific proteins. However, all the aforementioned multi-label learning methods with weak labels [25], [39], [24] do not take into consideration this knowledge. Here, we introduce ProWL-IF, a variation of ProWL, which takes advantage of the annotated relevant and irrelevant functions, in addition to missing functions.

In this setting, we have a partially annotated function set $Z = [\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_n]$, with $z_{ik} = 1$ if protein i has the k-th function, $z_{ik} = -1$ if protein i does not have this function, and $z_{ik} = 0$ if it's unknown whether the protein has the function, i.e. the corresponding function is missing. At first, we transform Z into $\bar{Z} = [\bar{\mathbf{z}}_1, \bar{\mathbf{z}}_2, \dots, \bar{\mathbf{z}}_n]$, where $\bar{\mathbf{z}}_i = \frac{\mathbf{z}_i + abs(\mathbf{z}_i)}{2}$, and $abs(\mathbf{z}_i)$ computes the absolute values of $\mathbf{z}_i$, with each element corresponds to one entry of $\mathbf{z}_i$. This transformation of Z is intuitive but yet reasonable because correctly predicting relevant function is more desirable than irrelevant functions, and a protein often has more irrelevant functions than relevant functions. In the future we will investigate other possible and effective ways for transforming Z. Next, we define the correlation between functions s and t based on $\bar{Z}$ as follows:

$$\tilde{C}_{st} = \frac{\bar{\mathbf{Z}}_{\cdot s}^T \bar{\mathbf{Z}}_{\cdot t}}{\|\bar{\mathbf{Z}}_{\cdot s}\| \|\bar{\mathbf{Z}}_{\cdot t}\|} \tag{12}$$

where $\bar{\mathbf{Z}}_{\cdot s}$ is the s-th column of $\bar{Z}$. Similar to ProWL, we normalize $\tilde{C}$ into C as in Eq. (2) and define the function induced graph W^f as in Eq. (7) based on $\bar{Z}$.

Similar to Eq. (3), the weighted loss function of ProWL-IF is defined as:

$$\Psi_1(\mathbf{f}) = \frac{1}{2}\sum_{i=1}^{n}\sum_{k=1}^{K} M_{ik}^{'}(f_{ik} - \tilde{z}_{ik})^2 \tag{13}$$

where $\tilde{Z} = [\tilde{\mathbf{z}}_1, \tilde{\mathbf{z}}_2, \ldots, \tilde{\mathbf{z}}_n]$ is the extended label set of proteins. For the i-th protein with respect to the k-th function, $\tilde{z}_{ik}$ is specified as:

$$\tilde{z}_{ik} = \begin{cases} z_{ik}, & z_{ik} = 1 \ or \ z_{ik} = -1 \\ \bar{\mathbf{z}}_i^T \mathbf{c}_{.k}, & z_{ik} = 0 \end{cases} \tag{14}$$

where $\mathbf{c}_{.k}$ is the k-th column of the correlation matrix C, and $M_{ik}^{'}$ is the weight of protein i with respect to the k-th function:

$$M_{ik}^{'} = \begin{cases} 1, & z_{ik} = 1 \ or \ z_{ik} = -1 \\ \bar{\mathbf{z}}_i^T \mathbf{c}_{.k}, & z_{ik} = 0 \end{cases} \tag{15}$$

Eq. (13) is similar to Eq. (3), but in Eq. (13) $\tilde{Z} \in [-1, 1]$ and in Eq. (3) $\tilde{Y} \in [0, 1]$. In addition, Eq. (13) does not consider the irrelevant functions as candidates of missing functions, whereas Eq. (3) does. Therefore, ProWL-IF has the advantage of properly capturing the domain information.

The objective of ProWL-IF is to minimize the following function:

$$\begin{aligned} \Psi(F) &= \frac{1}{2}\sum_{i=1}^{n}\sum_{k=1}^{K} M'_{ik}(f_{ik} - \tilde{z}_{ik})^2 \\ &+ \alpha tr(F^T L F) + \beta \|(F + 1_{n\times K})^T (F + 1_{n\times K})\| \\ &= \frac{1}{2}\|M^{'} \circ (F - \tilde{Z})^T (F - \tilde{Z})\| + \alpha tr(F^T L F) \\ &+ \beta \|(F + 1_{n\times K})^T (F + 1_{n\times K})\| \end{aligned} \tag{16}$$

$1_{n\times K}$ is an $n \times K$ matrix with all entries equal to 1. The third term controls the complexity and sparsity of F, since each protein has a large proportion of irrelevant functions (denoted by -1) and a small proportion of relevant functions (denoted by 1). α and β are scalar parameters to balance the importance of the smoothness and sparsity terms, respectively.

Taking the derivative of $\Psi(F)$ with respect to F, we have:

$$\frac{\partial \Psi(F)}{\partial F} = M^{'} \circ (F - \tilde{Z}) + \alpha L F + \beta I_{n\times n}(F + 1_{n\times K}) \tag{17}$$

where $I_{n\times n}$ is an $n \times n$ identity matrix. Similar to Eq. (9), Eq. (17) can be divided into K problems and will be solved as:

$$(\tilde{M}_{.k}^{'} + \alpha L + \beta I_{n\times n})\mathbf{f}_{.k} = \mathbf{q}_k \tag{18}$$

where

$$\tilde{M}_{.k}^{'} = diag(\mathbf{M}_{.k}^{'}), \mathbf{q}_k = \mathbf{M}_{.k}^{'} \circ \tilde{\mathbf{Z}}_{.k} - \beta I_{n\times n}\mathbf{1}_{n\times 1} \tag{19}$$

Eq. (18) can be efficiently solved in the same way as Eq. (10), and the learning procedure for ProWL-IF is similar to that of ProWL (**Algorithm 1**).

4 Experimental Setup

4.1 Datasets

We evaluate the proposed methods by testing their performance on the tasks of replenishing missing functions and predicting protein functions on four benchmarks. The first dataset (**PPI17**) is a PPI network of *Saccharomyces Cerevisiae* extracted from the BioGrid [1] with PubMed ID 17200106 [2]. Its largest connected component contains 1002 proteins annotated according to FunCat[3] [7], across 33 functions. The functions in FunCat are organized in a tree structure. We use the most informative functions as defined in [18], [16]. Informative functions are the ones that have at least 30 proteins as members, and within the tree structure these functions do not have a particular descendant node with more than 30 proteins. The second dataset, *Saccharomyces Cerevisiae* PPIs (**ScPPI**), was downloaded from BioGrid (2011-12-25). After the preprocessing and filtering, it contains 3041 proteins annotated with 86 informative functions. The weight matrix W^p of ScPPI is specified by the number of common PubMed IDs, where 0 implies no interaction between two proteins and $q > 0$ implies the interaction is supported by q distinct publications. The third dataset (**HumanPPI**) was extracted from heterogeneous data sources of human protein-protein interactions benchmarks[4] [11]. We use its largest connected component, which includes 2950 proteins annotated according to the Gene Ontology [23]. Similar to [11], we use the functions that have at least 30 annotated proteins. The fourth dataset (**Yeast**) was used in WELL[5] [24] and includes 1500 proteins annotated with 14 functions. We specify the weight matrix W^p in the same way as it was done for WELL. The weight matrices W^p of **PPI17** and **HumanPPI** were specified by the providers. We do not specifically handle hierarchical structure or the transitive closer among functional annotations. For PPI17, ScPPI and Yeast datasets, we considered functional annotations at a *flat* level. For the HumanPPI dataset, the functional annotations are organized in hierarchical structure (Gene Ontology). However, for this study we do not utilize the hierarchical or transitive structure prevalent within the underlying data. The statistics of the processed datasets are listed in Table 1.

To simulate the incomplete annotation scenario, we assume that the annotations on the currently labeled proteins are complete and mask the ground truth (or relevant) functions of proteins. For example, if a protein has 4 functions (labels), we can change 2 functions from 1 to 0. As a result, it becomes unknown whether these masked functions belong to the protein

[1] http://thebiogrid.org/

[2] http://www.ncbi.nlm.nih.gov/pubmed/17200106

[3] http://mips.helmholtz-muenchen.de/proj/funcatDB/

[4] http://morrislab.med.utoronto.ca/∼sara/SW

[5] http://lamda.nju.edu.cn/files/WELL.rar

TABLE 1
Statistics of datasets (Avg±Std means average number of functions for each protein and its standard deviation)

Dataset	#Proteins	#Functions	Avg±Std
PPI17	1002	33	2.00 ± 1.37
ScPPI	3041	86	1.94 ± 1.60
HumanPPI	2950	200	6.86 ± 3.77
Yeast	1500	14	4.23 ± 1.58

or not. In this case, the incomplete function (IF) ratio is $2/4 = 50\%$. Explicit irrelevant functional annotations are quite rare in both the Gene Ontology and in the FunCat database. As such, to simulate the incomplete annotation settings in ProWL-IF, we assume the currently available functional annotations of a protein as relevant functions and the other unspecified functional annotations of this protein as irrelevant functions. We mask both relevant functions (1s) and irrelevant functions (-1s) as missing functions (0s). To keep consistency, the IF ratio is kept the same as in ProWL (i.e. we mask the same IF ratio of relevant functions). In addition, we also mask an equal number of -1s as 0s.

4.2 Comparing Methods and Evaluation Metrics

We compare our methods with (i) WELL [24], (ii) MLR-GL [25], (iii) FCML [7], and (iv) CIA [37]. The first two approaches are weak-label learning methods, and the other two methods are recently developed protein function prediction algorithms using multi-label learning and PPI networks. WELL and MLR-GL need an input kernel matrix, and we substitute the kernel with the PPI matrices W^p, or compute W^p as done in WELL [24]. In fact, the weight of each interaction between proteins in the PPI datasets is no smaller than zero, thus a PPI matrix is as a semi-definite positive matrix. WELL was originally proposed for replenishing the missing functions. Here, we adopt WELL for Task 2 by including the unlabeled proteins in the input kernel matrix. MLR-GL was initially developed for predicting the functions of testing proteins using partially annotated proteins. We adopt MLR-GL for Task 1 by using all the proteins as training and testing proteins. Note, W^f in Eq. (5) used by ProWL, FCML and CIA is computed based on the incomplete (or partial) annotations on proteins, instead of the to-be predicted annotations F. Both ProWL, CIA and FCML make use of W^f (function induced graph) and W^p (PPI network) as inputs for protein function prediction. The parameters of WELL are specified as the authors reported [24]. For MLR-GL, we use the default parameters in the package provided by the authors[6]. For FCML, we set $\alpha = 0.01$; for CIA, we use the default setting as in the original paper. For ProWL and ProWL-IF, we set α and β to 0.01 and 0.001. We observed that the performance with respect to various metrics does not change as we vary α and β around the fixed values.

Various evaluation metrics have been developed for evaluating multi-label learning methods [14]. Here we use six evaluation metrics, namely, *MacroF1*, *MicroF1*, *AvgROC*, *RankingLoss*, *adapted AUC* [25] and *Coverage*. For maintaining consistency with other evaluation metrics, we report *1-RankingLoss*. Thus similar to other metrics (except *Coverage*), the higher the value of *1-RankingLoss*, the better the performance. Some of these metrics were also used to evaluate WELL [24] and MLR-GL [25]. The discussion on these metrics are described in the supplementary file.

5 Experimental Analysis

5.1 Performance on Replenishing Missing Functions

We performed experiments to assess the performance of the proposed methods in replenishing the missing functions. In these experiments, all the proteins within the datasets are used as training and testing data. To investigate the performance of different methods, we vary the IF ratio of each protein from 20% to 80%, with an interval of 20%. Some proteins in the PPI network do not have functions. To make use of the PPI network structure and keep the network connected, we do not remove them, but we evaluate the performance of replenishing missing functions on only the proteins with annotations. The experimental results (average of 20 independent runs and standard deviations) are reported in Tables 2-5 (more experimental results can be found in the supplementary file). We use pairwise t-test at 95% significant level to check the difference among these comparing methods and report the best performance in **boldface**. WELL formulation involves quadratic programming to compute the solution and HumanPPI has a large number of proteins and functions. Thus, WELL did not complete on our system with 4GB memory. *MacroF1*, *MicroF1*, *1-HammingLoss*, *Accuracy*, and *Completeness* require partitioning the predicted likelihood vector $\mathbf{f}_i$ into a binary indicative label vector. Here, we consider the functions corresponding to the largest s values of $\mathbf{f}_i$ as the relevant ones, and the remaining as irrelevant functions of protein i. s is determined by the number of ground-truth functions of the i-th protein. Note, irrespective of the comparing methods, s is the same for each comparing method, and the number of missing functions with respect to a protein is the same. In addition, this setting of s also helps us to count how many functions are correctly replenished. Given these reasons, we set s equal to the number of ground truth functions for each protein. For the Task 2 (See Subsection 5.4, prediction on completely unlabeled proteins), we adapt another setting of s, where s is

[6] http://www.cse.msu.edu/∼bucakser/

TABLE 2
Replenishing missing functions on PPI17

Metric	IF Ratio	ProWL	WELL	MLR-GL	FCML
MacroF1	20%	**97.84±0.18**	45.07±0.82	35.09±0.65	90.51±0.36
	40%	**89.24±0.33**	39.20±1.51	36.09±0.91	84.84±0.60
	60%	**76.70±1.02**	33.18±1.91	37.37±1.13	**77.02±1.12**
	80%	**69.30±1.08**	25.01±1.09	42.18±0.65	67.81±1.38
MicroF1	20%	**98.19±0.13**	55.88±0.36	36.80±0.47	90.46±0.27
	40%	**90.32±0.28**	50.94±0.53	38.49±0.99	85.75±0.51
	60%	**78.59±0.90**	44.97±0.86	39.49±0.66	**78.15±0.90**
	80%	**71.33±0.97**	38.55±0.78	45.64±0.77	**71.23±1.16**
AvgROC	20%	**99.73±0.05**	96.77±0.04	74.39±0.28	99.17±0.06
	40%	**98.30±0.17**	94.89±0.17	74.52±0.30	97.58±0.11
	60%	**94.97±0.36**	91.85±0.36	73.77±0.43	**94.98±0.29**
	80%	**91.78±0.41**	88.47±0.76	68.91±0.72	91.13±0.70
1-RankingLoss	20%	**99.86±0.02**	90.92±0.04	70.77±0.35	99.33±0.04
	40%	**98.87±0.08**	89.66±0.19	69.95±0.57	98.37±0.09
	60%	**96.45±0.21**	86.59±0.41	68.40±0.48	**96.42±0.23**
	80%	93.87±0.27	83.92±0.23	61.71±0.52	**94.48±0.25**
AUC	20%	**98.41±0.05**	90.15±0.04	72.13±0.37	97.80±0.08
	40%	**96.77±0.13**	87.74±0.12	71.80±0.37	96.19±0.14
	60%	**93.70±0.25**	83.74±0.45	71.43±0.36	**93.73±0.28**
	80%	89.35±0.38	79.60±0.23	67.59±0.48	**90.64±0.31**
Coverage ↓	20%	**1.34±0.03**	5.56±0.03	13.06±0.17	1.67±0.05
	40%	**2.30±0.09**	6.53±0.07	13.63±0.19	2.60±0.08
	60%	**3.89±0.17**	8.21±0.22	13.84±0.09	**3.88±0.14**
	80%	5.36±0.16	9.28±0.07	16.30±0.17	**4.97±0.16**
Overall	Win/Draw/Lose	14/7/3	0/0/24	0/0/24	3/7/14

TABLE 3
Replenishing missing functions on ScPPI

Metric	IF Ratio	ProWL	WELL	MLR-GL	FCML
MacroF1	20%	**95.76±0.19**	56.92±0.15	22.99±0.65	95.09±0.13
	40%	86.47±0.30	51.08±0.06	24.91±0.47	**87.89±0.23**
	60%	71.60±0.49	42.01±0.19	27.24±0.55	**77.23±0.42**
	80%	61.37±0.64	35.12±0.99	27.21±0.68	**66.31±0.49**
MicroF1	20%	**95.72±0.12**	61.87±0.22	22.66±0.35	94.81±0.12
	40%	86.78±0.19	55.94±0.15	24.55±0.42	**87.90±0.23**
	60%	71.12±0.30	46.06±0.23	26.88±0.47	**78.43±0.36**
	80%	59.44±0.44	38.76±0.10	26.26±0.53	**69.17±0.37**
AvgROC	20%	**99.77±0.03**	98.83±0.11	64.03±0.51	99.63±0.04
	40%	98.50±0.07	94.74±0.22	62.14±0.53	**98.61±0.08**
	60%	94.59±0.21	88.32±0.09	60.63±0.82	**96.53±0.12**
	80%	87.75±0.28	83.36±0.12	55.06±0.96	**93.27±0.14**
1-RankingLoss	20%	**99.80±0.01**	95.24±0.03	45.47±0.37	99.79±0.01
	40%	**98.97±0.03**	93.48±0.08	42.99±0.47	**98.97±0.07**
	60%	96.28±0.13	90.13±0.03	40.81±0.54	**96.65±0.09**
	80%	92.52±0.30	88.24±0.12	33.21±0.60	**93.99±0.14**
AUC	20%	**99.08±0.03**	93.95±0.05	56.21±0.35	99.01±0.03
	40%	**97.78±0.05**	91.16±0.16	54.39±0.47	**97.79±0.11**
	60%	94.69±0.15	87.54±0.02	53.57±0.67	**95.26±0.12**
	80%	89.01±0.43	84.60±0.19	49.38±0.70	**91.37±0.17**
Coverage ↓	20%	**2.10±0.05**	9.30±0.06	54.23±0.53	2.15±0.06
	40%	**4.51±0.11**	13.44±0.34	57.68±0.55	**4.51±0.21**
	60%	9.68±0.26	18.70±0.02	59.59±0.74	**9.02±0.24**
	80%	15.91±0.51	20.99±0.45	66.08±0.64	**14.10±0.27**
Overall	Win/Draw/Lose	6/3/15	0/0/24	0/0/24	15/3/6

equal to the average number of functions per protein in the dataset. To simulate the incomplete annotation scenario, we assume the annotated functions of protein i in the dataset as the ground-truth functions. The ground-truth functions include the masked functions and the partially annotated (or unmasked) functions.

From these Tables (2-5), we can observe that ProWL outperforms WELL and MLR-GL in replenishing missing functions of proteins in almost all the metrics across the four datasets. Both ProWL and FCML take advantage of the guilt by association rule and function correlation explicitly. ProWL achieves better performance than FCML on PPI17 and Yeast; ProWL and FCML have similar performance on ScPPI and HumanPPI. Overall, ProWL performs better than FCML. Taking *MacroF1* on Yeast, for example, ProWL on average is 4.41% better than WELL, 52.22% better than MLR-GL, 28.73% better than FCML. These results confirm the effectiveness of ProWL in replenishing the missing functions. The experimental results also

TABLE 4
Replenishing missing functions on HumanPPI

Metric	IF Ratio	ProWL	MLR-GL	FCML
MacroF1	20%	**96.39±0.17**	15.11±0.37	96.32±0.09
	40%	**93.03±0.25**	16.46±0.30	90.43±0.28
	60%	**82.94±0.53**	16.02±0.29	80.45±0.46
	80%	58.50±0.78	13.50±0.31	**60.18±0.52**
MicroF1	20%	**96.71±0.14**	15.03±0.31	96.44±0.08
	40%	**93.75±0.22**	16.64±0.26	91.01±0.21
	60%	**84.16±0.49**	16.44±0.29	81.86±0.37
	80%	58.59±0.83	14.35±0.28	**62.99±0.51**
AvgROC	20%	**99.80±0.02**	61.06±0.17	99.70±0.02
	40%	**99.29±0.05**	61.76±0.16	99.14±0.04
	60%	**97.90±0.10**	61.74±0.20	**97.89±0.07**
	80%	**93.56±0.24**	58.27±0.23	**93.56±0.26**
1-RankingLoss	20%	99.86±0.02	54.68±0.21	**99.89±0.01**
	40%	99.43±0.04	55.10±0.21	**99.52±0.03**
	60%	98.13±0.12	57.98±0.22	**98.47±0.06**
	80%	94.07±0.36	57.75±0.38	**95.39±0.18**
AUC	20%	**98.77±0.03**	54.06±0.19	**98.78±0.02**
	40%	98.09±0.06	54.69±0.17	**98.15±0.05**
	60%	96.42±0.13	55.71±0.13	**96.74±0.09**
	80%	90.17±0.35	55.32±0.27	**92.11±0.21**
Coverage ↓	20%	**4.63±0.23**	124.80±0.58	**4.45±0.19**
	40%	8.60±0.37	131.60±0.76	**7.64±0.27**
	60%	16.69±0.69	130.12±0.65	**13.99±0.39**
	80%	36.79±1.70	129.76±1.15	**29.84±0.83**
Overall	Win/Draw/Lose	8/4/12	0/0/24	12/4/8

TABLE 5
Replenishing missing functions on Yeast

Metric	IF Ratio	ProWL	WELL	MLR-GL	FCML
MacroF1	20%	**96.34±0.34**	84.73±0.50	40.54±0.34	88.96±0.16
	40%	**89.54±0.65**	77.22±0.98	41.72±0.51	75.37±0.16
	60%	**78.67±1.29**	65.15±0.97	42.27±0.93	60.32±0.55
	80%	**59.14±2.31**	56.38±0.59	40.12±0.39	45.48±0.51
MicroF1	20%	**97.28±0.13**	92.73±0.19	56.51±0.33	92.66±0.03
	40%	**92.34±0.32**	86.35±0.25	56.96±0.54	75.24±0.18
	60%	**83.93±0.76**	77.28±0.30	57.43±1.07	56.70±0.45
	80%	68.54±2.55	**71.27±0.27**	53.84±0.93	41.15±0.42
AvgROC	20%	**99.08±0.06**	96.04±0.28	52.20±0.47	98.02±0.12
	40%	**97.85±0.15**	88.01±0.37	54.50±0.30	93.97±0.21
	60%	**92.79±0.62**	78.34±0.40	56.78±0.45	86.66±0.59
	80%	**76.38±1.65**	70.43±0.45	56.37±0.50	75.31±0.68
1-RankingLoss	20%	**99.40±0.04**	97.30±0.06	69.38±0.25	96.30±0.06
	40%	**98.17±0.07**	94.71±0.10	70.61±0.51	84.19±0.09
	60%	**94.83±0.35**	89.63±0.16	72.44±0.46	67.68±0.32
	80%	85.42±1.47	**86.72±0.13**	71.04±0.79	52.73±0.40
AUC	20%	**96.34±0.04**	94.60±0.07	72.81±0.22	92.78±0.07
	40%	**95.10±0.07**	91.87±0.09	73.56±0.44	80.57±0.09
	60%	**92.06±0.29**	87.69±0.15	74.45±0.40	65.45±0.28
	80%	83.11±1.34	**84.69±0.11**	73.43±0.63	50.65±0.35
Coverage ↓	20%	**3.45±0.02**	4.06±0.03	8.49±0.05	4.96±0.03
	40%	**3.89±0.03**	4.91±0.02	8.51±0.07	8.59±0.03
	60%	**4.70±0.09**	5.95±0.04	8.42±0.07	10.59±0.06
	80%	6.58±0.25	**6.29±0.02**	8.38±0.10	11.19±0.09
Overall	Win/Draw/Lose	20/0/4	4/0/20	0/0/24	0/0/24

corroborate our motivation in combining guilty by association rule and function correlation.

Another observation is that the performance of ProWL, WELL and FCML downgrade as the IF ratio increases. As more relevant functions are masked, the function correlation measure becomes less reliable and the task becomes more difficult. Since ProWL exploits the function correlation matrix C to specify the weight matrix M, as more functions are missing, C becomes less accurate and M turns out to be less reliable. Thus, ProWL and FCML have similar performance when the IF ratios are large (i.e., 60%), and ProWL is sometimes outperformed by FCML when a very large portion of functions are missing (i.e., 80%). The performance of MLR-GL varies based on experiment. MLR-GL was originally developed for predicting completely unlabeled samples by making use of incomplete labeled training samples. Here, it is adapted to replenish missing functions. As the IF ratio increases, the number of missing functions

TABLE 6
Experimental results of ProWL-IF on replenishing missing functions. The better performance are shown in boldface (statistical significance is examined via pairwise t-test at 95% significant level).

Dataset	Method	MicroF1				1-RankingLoss				AUC			
		20%	40%	60%	80%	20%	40%	60%	80%	20%	40%	60%	80%
PPI17	ProWL	98.19	90.32	78.59	71.33	99.86	98.87	96.45	93.87	98.41	96.77	93.70	89.35
	ProWL-IF	**98.75**	**92.65**	**82.34**	**76.10**	**99.96**	**99.61**	**98.63**	**97.53**	**98.67**	**98.38**	**97.75**	**96.85**
ScPPI	ProWL	95.72	86.78	71.12	59.44	99.75	99.23	98.33	97.65	99.08	97.78	94.69	89.01
	ProWL-IF	**97.57**	**90.20**	**80.19**	**72.88**	**99.97**	**99.79**	**99.37**	**98.81**	**99.33**	**99.17**	**98.86**	**98.40**
HumanPPI	ProWL	**96.71**	**93.75**	**84.16**	58.59	99.86	99.43	98.13	94.07	98.77	98.09	96.42	90.17
	ProWL-IF	94.33	85.40	74.79	**65.52**	**99.97**	**99.81**	**99.49**	**99.06**	**98.98**	**98.77**	**98.39**	**97.84**
Yeast	ProWL	**97.28**	**92.34**	**83.93**	**68.54**	**99.40**	**98.17**	**94.83**	**85.42**	96.34	**95.10**	**92.06**	83.11
	ProWL-IF	96.20	85.64	74.88	65.20	99.32	95.73	87.48	75.88	**96.60**	94.61	90.30	84.04

rises and the number of functional classes with less than 30 annotations also ascends. From the results in Tables (2-5), when there is a large number of missing functions, ProWL often performs similar with FCML, and outperforms other approaches.

We conducted additional experiments on the four datasets to investigate the performance of ProWL-IF. In these experiments, we masked few of the relevant (+1) and irrelevant (-1) functions for a protein as 0s. The defintion of IF ratio for ProWL-IF is similar to the previous experiments set for ProWL. We mask the same number of irrelevant functions (-1) as the relevant functions (+1) for each protein. For example, if two relevant functions (+1) of a protein are masked as 0s, two irrelevant functions (-1) are also masked as 0s (if this protein has at least two irrelevant functions).

We repeat ProWL-IF 20 times and in each run, we randomly mask the relevant and irrelevant functions according to the fixed IF ratio. For brevity, we just report the average results with respect to *MacroF1*, *1-RankingLoss*, and *AUC* in Table 6. We can observe that ProWL-IF generally outperforms ProWL. Another observation is that, as the ratio of missing function increase, the downgrade trend of ProWL-IF is not so pronounced as for ProWL. The reason is that ProWL-IF makes use of both *relevant* functions and *irrelevant* functions as prior knowledge, whereas ProWL just takes advantage of *relevant* functions as prior knowledge. There is a contrary phenomenon on the Yeast dataset that the performance of ProWL is generally better than that of ProWL-IF. The possible reasons are two-fold: (i) in the Yeast data set the number of relevant functions is 6342 and the number of irrelevant functions is 14658, so the assumption that each protein has a large number of irrelevant functions is not feasible here, and (ii) ProWL-IF uses $\|(F + 1_{n\times K})^T(F + 1_{n\times K})\|$ to enforce the prediction toward irrelevant functions.

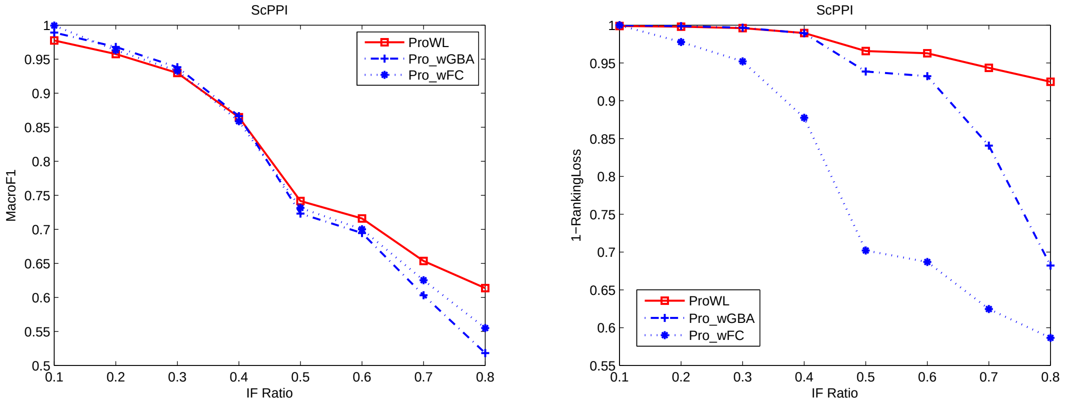
5.2 The Benefit of using Guilt By Association and Function Correlation

We also perform experiments to investigate the benefit of using guilt by association rule and function correlation. We introduce two variants of ProWL: (i)Pro_wGBA and (ii) Pro_wFC. Pro_wGBA corresponds to *Pro*tein function prediction using weak-label learning *w*ithout using *G*uilt *B*y *A*ssociation rule. Specifically, Pro_wGBA is based on Eq. (9) with $\alpha = 0$, that is Pro_wGBA just uses the currently incomplete annotation information and function correlation to replenish the missing functions. Pro_wFC corresponds to *Pro*tein function prediction using weak-label learning *w*ithout using *F*unction *C*orrelation. In Pro_wFC, Y is used in Eq. (10) instead of $\tilde{Y}$, M is set using annotated functions only and without considering the function correlation.

We vary the IF ratio from 10% to 80% at intervals of 10% and record the results of ProWL, Pro_wGBA and Pro_wFC. For brevity, in Figure 2 we just report the results with respect to MacroF1 and 1-RankingLoss on ScPPI. The results with respect to PPI17 are reported in the supplementary file. Pro_wGBA and Pro_wFC have similar performance as ProWL when a small number of functions are missing. This implies that guilt by association rule and function correlation can help replenish the missing functions to some extent. However, as the number of masked functions increases, the difference between ProWL, Pro_wGBA, and Pro_wFC increases. This can be attributed to the fact that ProWL takes advantage of both guilt by association and function correlation, whereas the two variants just make use of one or the other. These results demonstrate that both guilt by association rule and function correlation are important to replenish the missing functions, especially when a large portion of functions is missing.

5.3 The Benefit of using Biological Induced Graph and Function Induced Graph

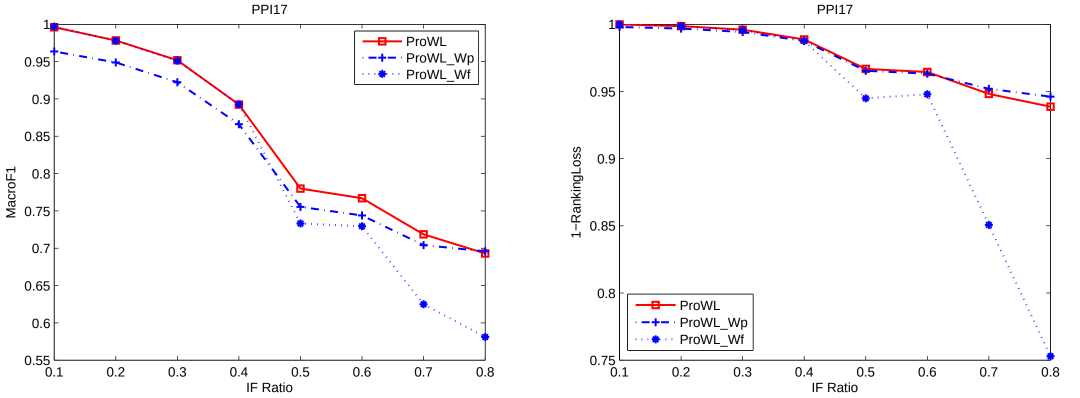
In this section, we conduct experiments to investigate the benefit in integrating biological induced graph and function induced graph. We introduce two variants of ProWL: (i) Pro_Wp and (ii) Pro_Wf. Pro_Wp stands for *Pro*tein function prediction using weak-label learning based on biological induced graph only (i.e., PPI networks, protein sequence induced pairwise similarity graph). Pro_Wf stands for *Pro*tein function prediction using weak-label learning based on function induced graph only.



(a) ScPPI (MacroF1)

(b) ScPPI (1-RankingLoss)

Fig. 2. The Benefit of using Guilt By Association and Function Correlation (Pro_wGBA means *Pro*tein function prediction *w*ithout *G*uilt *B*y *A*ssociation, Pro_wFC means Protein function prediction *w*ithout *F*unction *C*orrelation).



(a) PPI17 (MacroF1)

(b) PPI17 (1-RankingLoss)

Fig. 3. The Benefit of using Function Induced Graph and PPI Graph (ProWL_Wp means ProWL on PPI graph only, and ProWL_Wf means ProWL on *F*unction *i*nduced *g*raph only).

We vary the IF ratio from 10% to 80% and record the results of ProWL, ProWL_Wp and ProWL_Wf. For brevity, in Figure 3 we report only the results with respect to MacroF1 and 1-RankingLoss on PPI17. The results with respect to ScPPI are reported in the supplementary file. We can observe that ProWL often performs better than ProWL_Wg and ProWL_Wf for various IF ratios. The performance of Pro_Wf is poorer in comparison to ProWL and ProWL_Wp as the IF ratio increasing. As the number of missing functions increases, the pairwise similarity, induced from the overlapping functions between two proteins, becomes less reliable. For this reason, ProWL_Wf performs much worse than ProWL_Wp and ProWL, and ProWL_Wp performs slightly better than ProWL (see Figure 3(b)). ProWL often outperforms ProWL_Wp. This fact indicates that the function induced graph indeed reinforces the protein function prediction on PPI networks. ProWL_Wf sometimes gets better performance than ProWL when the IF ratio is small, but is outperformed by ProWL when a large portion of functions are missing. Overall, it is beneficial to make use of the composite function and biologically induced graph.

5.4 Performance on Task 2 (Completely Unlabeled Proteins)

We conduct experiments to evaluate the effectiveness of ProWL in predicting the function of completely unlabeled proteins using incomplete annotations on labeled proteins. We first divided each dataset into two subsets: (i) training set (accounting for 70% of the all proteins) with missing annotations and (ii) testing set (accounting for the remaining 30% of all the proteins) with no annotations (i.e., completely unannotated). We repeat these experiments 20 times. Each time, the dataset is randomly partitioned into training and testing datasets, and 50% functions of the labeled proteins are randomly masked in the training

TABLE 7
Prediction of Unlabeled Proteins with Incomplete Annotations on PPI17

Metric	ProWL	WELL	MLR-GL	FCML	CIA
MacroF1	**44.97±2.01**	16.91±2.09	32.61±1.96	38.19 ±2.01	37.70±2.20
MicroF1	**54.04±2.20**	33.84±1.39	35.47±1.49	49.71 ±2.55	42.59±1.86
1-RankingLoss	**86.74±1.38**	78.24±1.05	69.24±3.09	84.77 ±1.38	50.63±2.43
AUC	**86.45±1.19**	76.45±1.26	78.59±1.84	84.13 ±1.40	75.64±1.22
Coverage ↓	**7.15±0.54**	10.87±0.34	8.01±0.45	7.85±0.49	13.34±1.37

TABLE 8
Prediction of Unlabeled Proteins with Incomplete Annotations on ScPPI

Metric	ProWL	WELL	MLR-GL	FCML	CIA
MacroF1	**32.35±1.22**	5.35 ±0.35	25.09±1.34	29.86±1.09	25.68±0.94
MicroF1	**30.66±0.86**	19.40 ±1.58	22.72±0.43	27.91±0.73	19.79±0.53
1-RankingLoss	61.00±1.00	**71.84 ±1.40**	39.31±0.63	68.44±0.94	20.09±0.70
AUC	**79.13±0.55**	76.37 ±1.07	60.25±0.69	74.00±0.86	62.81±0.48
Coverage↓	**27.89±0.63**	30.63 ±1.13	51.71±2.15	37.61±1.16	49.05±0.82

TABLE 9
Prediction of Unlabeled Proteins with Incomplete Annotations on HumanPPI

Metric	ProWL	MLR-GL	FCML	CIA
MacroF1	**19.60±0.89**	11.85±0.96	14.99 ±0.72	10.34±0.58
MicroF1	**23.14±0.97**	12.92±0.93	17.46 ±0.98	13.08±0.67
1-RankingLoss	74.80±0.83	66.85±0.97	**75.23±0.93**	32.13±1.13
AUC	**77.35±0.58**	65.79±0.79	75.68 ±1.24	66.47±0.81
Coverage ↓	**71.04±1.46**	104.00±2.47	**70.01 ±2.20**	112.61±2.05

TABLE 10
Prediction of Unlabeled Proteins with Incomplete Annotations on Yeast

Metric	ProWL	WELL	MLR-GL	FCML	CIA
MacroF1	34.54±1.13	32.59±1.46	**41.47±0.60**	23.11±0.73	18.58±0.67
MicroF1	**63.15±1.37**	**62.67±1.31**	57.69±0.61	46.03 ±0.79	38.48±2.16
1-RankingLoss	**81.09±1.01**	**80.58±1.08**	76.83±0.76	60.14 ±1.44	48.19±2.54
AUC	**82.17±0.81**	**81.70±0.75**	78.63±0.77	64.01 ±1.06	70.45±1.77
Coverage ↓	**6.46±0.17**	**6.45±0.11**	7.38±0.13	10.58 ±0.08	8.63±0.75

set. The setting of missing functions for each protein is set as in the first set of experiments in Section 5.1, but s is determined as the average number of functions of all proteins. The experimental results (average of 20 independent runs) are reported in Tables 7- 10. The results with respect to *Accuracy* and *Completeness* are not included, since they were initially used to evaluate the performance of replenishing the missing functions. WELL on HumanPPI can not run to completion using 4GB of memory, so its results are not reported in Table 9.

From these tables (Table 7- Table 10), we can observe that ProWL achieves better performance than other comparing methods on various evaluation metrics. Taking the MacroF1 on PPI17 for example, ProWL on average is 165.94% better than WELL, 37.90% better than MLR-GL, 17.75% better than FCML, and 19.28% better than CIA. In the task of replenishing the missing functions, ProWL and FCML sometimes performs similarly to each other on ScPPI and HumanPPI. However, FCML always loses to ProWL in the task of predicting functions on completely unannotated proteins. ProWL takes into consideration the incomplete annotation in the training set (a more general case in real-world proteomic data), whereas FCML does not. CIA also takes advantage of function induced similarity and PPI network to predict protein functions, but it is always outperformed by ProWL. There are two possible reasons. Firstly, CIA does not consider the weights of interaction between two proteins. Secondly, CIA mainly depends on the function induced graph, when training proteins are only partially annotated, this graph becomes less reliable. MLR-GL predicts protein function under the assumption of partially annotated proteins, it is outperformed by ProWL. MLR-GL optimizes the ranking loss and group Lasso loss, whereas ProWL optimizes an objective function based on the function correlation and the guilt by association rule, which are more faithful to the characteristic of PPI data. For the same reasons, ProWL often outperforms WELL, which takes advantage of low density separation and low-rank based similarity to capture function correlation. All these results demonstrate the effectiveness of ProWL in predicting unlabeled proteins by considering the incomplete annotations on proteins.

5.5 Run Time Analysis

In Table 11, we also report the average run time for each of the methods (except CIA, which is only targeted at predicting function on unlabeled proteins) on the four datasets. The experiments are conducted on Windows 7 platform with Intel E8400 processor and 4GB memory. While ProWL has to solve Eq. (12) K times, its run time is ranked 2nd best amongst the four comparing methods. At the same time, it outperforms the other methods in Task 1 and Task 2. FCML infers the functions of a protein in one step, but it needs to compute the eigenvectors of the matrix associated with a PPI network. Eigendecomposition is computationally expensive, so it often takes more time than ProWL. MLR-GL solves the simplified Second Order Cone Programming (SOCP) [42] problem to solve the convex-concave optimization problem, which takes less time than the other methods. WELL uses eigendecomposition and convex optimization, so it takes much more time than the other comparing methods.

TABLE 11
Run time Analysis (seconds)

Dataset	ProWL	FCML	WELL	MLR-GL
PPI17	15.63	31.21	100.84	5.48
ScPPI	172.52	633.42	5783.49	5.04
HumanPPI	436.56	655.84	–	198.90
Yeast	27.45	85.33	97.80	85.22
Total	652.16	1405.80	5982.13	294.64

6 Conclusion

In this paper, we study the incomplete annotation problem in protein function prediction. We develop a method called ProWL. ProWL uses guilt by association rule and function correlation to replenish the missing functions of partially annotated proteins. It can also predict functions for completely unannotated proteins. To take advantage of irrelevant functions of proteins, we introduce a variant of ProWL, called ProWL-IF. Unlike traditional weak-label learning methods, which consider all the missing functions as candidates of relevant functions, ProWL-IF takes into account relevant, irrelevant, and missing functions of proteins. Our empirical study finds that the proposed methods perform better than other related methods. We will investigate a function correlation definition that can capture the correlation with a large ratio of missing functions. We also plan to search for a way to more efficient use of the irrelevant functions for ProWL-IF.

7 Acknowledgements

We are thankful to the anonymous reviewers and editors for their valuable comments. This paper is partially supported by grants from NSF IIS-0905117, NSF Career Award IIS-1252318, Natural Science Foundation of China (Project Nos. 61003174, 61101234, 61372138), Natural Science Foundation of Guangdong Province (S2012010009961), Specialized Research Fund for the Doctoral Program of Higher Education (20110172120027), Fundamental Research Funds for the Central Universities of China (Project Nos. XDJK2010B002, XDJK2013C026, XDJK2013C123), Doctoral Fund of Southwest University (Nos. SWU110063 and SWU113034) and China Scholarship Council (CSC).

References

[1] R. Sharan, I. Ulitsky, and R. Shamir, "Network-based prediction of protein function," *Molecular Systems Biology*, vol. 3, no. 1, 2007.

[2] L. Peña-Castillo, M. Tasan, C. Myers, H. Lee, T. Joshi, C. Zhang, Y. Guan, M. Leone, A. Pagnani, W. Kim *et al.*, "A critical assessment of mus musculus gene function prediction using integrated genomic evidence," *Genome Biology*, vol. 9, no. Suppl 1, p. S2, 2008.

[3] G. Pandey, V. Kumar, and M. Steinbach, "Computational approaches for protein function prediction," Department of Computer Science and Engineering, University of Minnesota, Twin Cities, Tech. Rep. TR 06-028, 2006.

[4] A. Elisseeff and J. Weston, "A kernel method for multi-labelled classification," in *Proceedings of Advances in Neural Information Processing Systems*, 2001, pp. 681–687.

[5] C. Leslie, E. Eskin, A. Cohen, J. Weston, and W. Noble, "Mismatch string kernels for discriminative protein classification," *Bioinformatics*, vol. 20, no. 4, pp. 467–476, 2004.

[6] J. Jiang, "Learning protein functions from bi-relational graph of proteins and function annotations," *Algorithms in Bioinformatics*, pp. 128–138, 2011.

[7] H. Wang, H. Huang, and C. Ding, "Function-function correlated multi-label protein function prediction over interaction networks," in *Proceedings of the 16th International Conference on Research in Computational Molecular Biology*, 2012, pp. 302–313.

[8] E. Becker, B. Robisson, C. Chapple, A. Guénoche, and C. Brun, "Multifunctional proteins revealed by overlapping clustering in protein interaction network," *Bioinformatics*, vol. 28, no. 1, pp. 84–90, 2012.

[9] H. Chua, W. Sung, and L. Wong, "Exploiting indirect neighbours and topological weight to predict protein function from protein–protein interactions," *Bioinformatics*, vol. 22, no. 13, pp. 1623–1630, 2006.

[10] K. Tsuda, H. Shin, and B. Schölkopf, "Fast protein classification with multiple networks," *Bioinformatics*, vol. 21, no. suppl 2, p. ii59, 2005.

[11] S. Mostafavi and Q. Morris, "Fast integration of heterogeneous data sources for predicting gene function with limited annotation," *Bioinformatics*, vol. 26, no. 14, pp. 1759–1765, 2010.

[12] G. Yu, C. Domeniconi, H. Rangwala, G. Zhang, and Z. Yu, "Transductive multi-label ensemble classification for protein function prediction," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2012, pp. 1077–1085.

[13] H. Mewes, D. Frishman, U. Güldener, G. Mannhaupt, K. Mayer, M. Mokrejs, B. Morgenstern, M. Münsterkötter, S. Rudd, and B. Weil, "Mips: a database for genomes and protein sequences," *Nucleic Acids Research*, vol. 30, no. 1, pp. 31–34, 2002.

[14] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Mining multi-label data," *Data Mining and Knowledge Discovery Handbook*, pp. 667–685, 2010.

[15] M. Zhang and Z. Zhou, "A review on multi-label learning algorithms," *IEEE Transactions on Knowledge and Data Engineering*.

[16] X. Zhang and D. Dai, "A framework for incorporating functional inter-relationships into protein function prediction algorithms," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 9, no. 3, pp. 740–753, 2012.

[17] G. Pandey, C. Myers, and V. Kumar, "Incorporating functional inter-relationships into protein function prediction algorithms," *BMC Bioinformatics*, vol. 10, no. 1, p. 142, 2009.

[18] J. Jiang and L. McQuay, "Predicting protein function by multi-label correlated semi-supervised learning," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 9, no. 4, pp. 1059–1069, 2012.

[19] Z. Barutcuoglu, R. Schapire, and O. Troyanskaya, "Hierarchical multi-label prediction of gene function," *Bioinformatics*, vol. 22, no. 7, pp. 830–836, 2006.

[20] G. Valentini, "True path rule hierarchical ensembles for genome-wide gene function prediction," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 8, no. 3, pp. 832–847, 2011.

[21] N. Cesa-Bianchi, M. Re, and G. Valentini, "Synergy of multi-label hierarchical ensembles, data fusion, and cost-sensitive methods for gene functional inference," *Machine Learning*, vol. 88, no. 1-2, pp. 209–241, 2012.

[22] A. Ruepp, A. Zollner, D. Maier, K. Albermann, J. Hani, M. Mokrejs, I. Tetko, U. Güldener, G. Mannhaupt, M. Münsterkötter *et al.*, "The funcat, a functional annotation scheme for systematic classification of proteins from whole genomes," *Nucleic Acids Research*, vol. 32, no. 18, pp. 5539–5545, 2004.

[23] M. Ashburner, C. Ball, J. Blake, D. Botstein, H. Butler, J. Cherry, A. Davis, K. Dolinski, S. Dwight, J. Eppig *et al.*, "Gene ontology: tool for the unification of biology," *Nature Genetics*, vol. 25, no. 1, p. 25, 2000.

[24] Y. Sun, Y. Zhang, and Z. Zhou, "Multi-label learning with weak label," in *Proceedings of 24th AAAI Conference on Artificial Intelligence*, 2010.

[25] S. Bucak, R. Jin, and A. Jain, "Multi-label learning with incomplete class assignments," in *Proceedings of 24th IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 2801–2808.

[26] G. Yu, G. Zhang, H. Rangwala, C. Domeniconi, and Z. Yu, "Protein function prediction using weak-label learning," in *Proceedings of the ACM Conference on Bioinformatics, Computational Biology and Biomedicine*, 2012, pp. 202–209.

[27] B. Schwikowski, P. Uetz, S. Fields *et al.*, "A network of protein-protein interactions in yeast," *Nature Biotechnology*, vol. 18, no. 12, pp. 1257–1261, 2000.

[28] A. Vazquez, A. Flammini, A. Maritan, and A. Vespignani, "Global protein function prediction from protein-protein interaction networks," *Nature Biotechnology*, vol. 21, no. 6, pp. 697–700, 2003.

[29] D. Lin, "An information-theoretic definition of similarity," in *Proceedings of the 15th International Conference on Machine Learning*, vol. 1, 1998, pp. 296–304.

[30] H. Wang, H. Huang, and C. Ding, "Image annotation using bi-relational graph of images and semantic labels," in *Proceedings of 24th IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 793–800.

[31] H. Tong, C. Faloutsos, and J. Pan, "Random walk with restart: fast solutions and applications," *Knowledge and Information Systems*, vol. 14, no. 3, pp. 327–346, 2008.

[32] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *Journal of Machine Learning Research*, vol. 7, pp. 2399–2434, 2006.

[33] C. Ding, H. Simon, R. Jin, and T. Li, "A learning framework using green's function and kernel regularization with application to recommender system," in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2007, pp. 260–269.

[34] P. Bogdanov and A. K. Singh, "Molecular function prediction using neighborhood features," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 7, no. 2, pp. 208–217, 2010.

[35] A. Mitrofanova, V. Pavlovic, and B. Mishra, "Prediction of protein functions with gene ontology and interspecies protein homology data," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 8, no. 3, pp. 775–784, 2011.

[36] M. Re, M. Mesiti, and G. Valentini, "A fast ranking algorithm for predicting gene functions in biomolecular networks," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 9, no. 6, pp. 1812–1818, 2012.

[37] X. Chi and J. Hou, "An iterative approach of protein function prediction," *BMC Bioinformatics*, vol. 12, no. 1, p. 437, 2011.

[38] D. Wang, S. Hoi, and Y. He, "Mining weakly labeled web facial images for search-based face annotation," in *Proceedings of the 34th ACM SIGIR Conference on Research and Development in Information*, 2011, pp. 535–544.

[39] Z. Qi, M. Yang, Z. Zhang, and Z. Zhang, "Mining partially annotated images," in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2011, pp. 1199–1207.

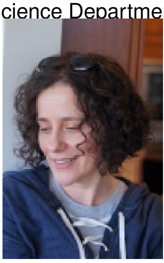
[40] D. Zhou, O. Bousquet, T. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Proceedings of Advances in Neural Information Processing Systems*, 2003, pp. 321–328.

[41] J. Nocedal and S. Wright, *Numerical Optimization*. Springer Verlag, 1999.

[42] F. Alizadeh and D. Goldfarb, "Second-order cone programming," *Mathematical Programming*, vol. 95, no. 1, pp. 3–51, 2003.

Guoxian Yu is an Associate Professor in the College of Computer and Information Science, Southwest University, Chongqing, China. He received B.Sc. degree in Software Engineering from Xi'an University of Technology, Xi'an, China in 2007, and Ph.D. in Computer Science from South China University of Technology, Guangzhou, China in 2013. He visited the Data Mining Lab in the George Mason University, VA, USA from 2011 to 2013. His current research interests include machine learning, data mining and bioinformatics.

Huzefa Rangwala is an Assistant Professor at the department of Computer Science, George Mason University, VA, USA. He received his Ph.D. in Computer Science from the University of Minnesota in 2008. His core research interests include bioinformatics, machine learning, and high performance computing. He is the recipient of the NSF Early Faculty Career Award in 2013, the 2013 Volgenau Outstanding Teaching Faculty Award, 2012 Computer Science Department Outstanding Teaching Faculty Award and 2011 Computer Science Department Outstanding Junior Researcher Award.



Carlotta Domeniconi is an Associate Professor in the Department of Computer Science at George Mason University, VA, USA. She received a B.Sc. degree in Computer Science from the University of Milan, Italy, in 1992, and a Ph.D. degree in Computer Science from the University of California, Riverside, in 2002. Her research interests include pattern recognition, machine learning, data mining, and feature relevance estimation. She has published extensively in premier data mining and machine learning conferences and journals. She is a recipient of an NSF CAREER Award.

Guoji Zhang is a Professor at the School of Sciences, South China University of Technology, Guangzhou, China. He received his B.Sc. degree in Computer Application and Ph.D. degree in Circuit and System from South China University of Technology, in 1977 and 1999, respectively. His research interests include computational intelligence, computational electromagnetic and cryptology, he has published over 50 research papers.

Zhiwen Yu is a Professor in the School of Computer Science and Engineering, South China University of Technology, Guangzhou, China. He received the B.Sc. and M.Phil. degrees from the Sun Yat-Sen University in China in 2001 and 2004 respectively, and the Ph.D. degree in Computer Science from the City University of Hong Kong, in 2008. His research interests include bioinformatics, machine learning, pattern recognition, intelligent computing and data mining. He has published more than 70 technical articles in referred journals and conference proceedings in the areas of bioinformatics, artificial intelligence, pattern recognition and multimedia.