January 2016

PROTEIN FUNCTION, DIVERISTY AND FUNCTIONAL INTERPLAY

Ishita Kamal Khan
Purdue University

Graduate School Form
30 Updated 12/26/2015

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By ISHITA KHAN

Entitled

PROTEIN FUNCTION, DIVERSITY AND FUNCTIONAL INTERPLAY

For the degree of Doctor of Philosophy

Is approved by the final examining committee:

DAISUKE KIHARA Chair	ALEX POTHEN
KIHONG PARK	JENNIFER NEVILLE
ROBERT D. SKEEL	

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University's "Policy of Integrity in Research" and the use of copyright material.

Approved by Major Professor(s): DAISUKE KIHARA

Approved by: SUNIL PRABHAKAR/WILLIAM J. GORMAN 8/3/2016

Head of the Departmental Graduate Program Date

PROTEIN FUNCTION, DIVERISTY AND FUNCTIONAL INTERPLAY

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Ishita K Khan

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

December 2016

Purdue University

West Lafayette, Indiana

To my father, who has been my best friend and my role model since I have known life, my mother, who would have been the happiest person in the planet to see me achieve this, and to my husband who has been devotedly by my side in every single step of the way.

ACKNOWLEDGEMENTS

First and foremost, I am sincerely thankful to my advisor Dr. Daisuke Kihara for being a very inspirational mentor and a friend throughout my PhD path. I consider myself extremely fortunate to have him as my teacher who have shown me how a simple path taken with patience, hard work and diligence can find its way despite the obstacles. I would also like to thank Dr. Alex Pothen, Dr. Kihong Park, Dr. Jennifer Neville, and Dr. Robert Skeel for their guidance, encouragement and suggestions. I appreciate the friendship and research contributions from all the past and present members of Kihara Lab and would like to take this opportunity to thank Meghana, Juan, Xuejiao, Yi, Xiaolei, Qing, Lenna, Xusi, Ziyun, Lyman, Charles, Mengmeng, Lillian, Tiange, Woonghee, Genki, Kim and others.

Last but not least, I am thankful to my family- my sister Shaulee, Mithun, Rumpa, brother Ishan, Sadat, Rahy, auntie Kishwar & uncle Sunnah for their bold existence and relentless love, my sweetheart little nephew and niece Shayor and Shuhita, as I have missed out on their childhood being far away from home, and my in-laws for being very supportive in the process. Without you all, I would not be in the place where I am today. *Thank you* is a mere word to say to you all for your unconditional love, loyalty and friendship.

TABLE OF CONTENTS

Page

Page

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

LIST OF ABBREVIATIONS

AFP	Automatic Function Prediction
BLAST	Basic Local Alignment and Search Tool
BP	Biological Process domain of Gene Ontology
CRF	Conditional Random Field
CAFA	Critical Assessment of Function Annotations
CC	Cellular Component domain of Gene Ontology
ESG	Extended Similarity Group algorithm
FAM	Function Association Matrix
GFP	Group Function Prediction
GO	Gene Ontology
KEGG	Kyoto Encyclopedia of Genes and Genomes
MP	Moonlighting Proteins
MF	Molecular Function domain of Gene Ontology
PFP	Protein Function Prediction algorithm

ABSTRACT

Khan, Ishita K. Ph.D., Purdue University, December 2016. Protein Function, Diversity, and Functional Interplay. Major Professor: Daisuke Kihara.

Functional annotations of novel or unknown proteins is one of the central problems in post-genomics bioinformatics research. With the vast expansion of genomic and proteomic data and technologies over the last decade, development of automated function prediction (AFP) methods for large-scale identification of protein function has become imperative in many aspects. In this research, we address two important divergences from the "one protein – one function" concept on which all existing AFP methods are developed:

1. One protein with multiple independent functions – Moonlighting Proteins: Moonlighting proteins perform more than one independent cellular function within one polypeptide chain. Recent biological experiments have been discovering such multi-functional proteins at a steady pace. Our work on moonlighting proteins can be divided into two logical parts: *1a.* Development of a computational framework for comprehensive genome-scale characterization of moonlighting proteins based on functional and context-based information. Our work identifies characteristic features of moonlighting proteins in both cases where current databases have functional annotations of the diverse functions of such proteins and cases where functional annotations do not exist. *1b.* Development of

automated prediction models of moonlighting proteins. We take two different approaches for our model development: using functional and context based features in a machine learning framework, and using text-based features, learned through text-mining algorithms.

2. Group of proteins sharing a common function: On a regular basis, biological experiments reveal sets of proteins involved in disease/disorder/cellular phenomena without sufficient explanation of the functional mechanisms of these group activities. Intuitively, proteins interact in a cell physically, through gene expression or genetic interaction to perform a common function that so often ends up causing a disease/disorder. To understand the functional nature of a set of proteins, it is often important to understand the functionalities in which they are involved in as a group, rather than understanding the detailed functional characteristics of the individual proteins. In this research, we develop a conditional random field (CRF)-based framework that predicts the function of the "protein groups", based on group neighborhood of their interaction network, and iteratively updates the function annotation of the unknown group members such that it reflects the protein's group activity.

For the protein function prediction research domain, it is vital to keep pace with existing AFP methods by improving the prediction accuracy, updating the models and making the methods available to the bioinformatics community. The final part of this research copes with the AFP problem in three aspects: improvement, database update and web-server development of two existing methods: PFP and ESG, and participation in a community-wide challenge for the AFP methods called CAFA (Critical Assessment of Function Annotation) and benchmarking the performances.

CHAPTER 1. INTRODUCTION

1.1 Background

Elucidating the biological function of proteins is vital to understanding the molecular mechanism of life, hence stands as a fundamental problem in diverse branches of biology and bioinformatics. As the amount of protein sequence and interaction data grows at an exponential rate, performing biological experiments to find functions of all the genes becomes an insurmountable task. At one end, large-scale experimental approaches give only non-specific information about the function of the protein, whereas in the other end small-scale experiments provide more direct evidence but are costly and labor intensive. Figure 1.1 shows the growth of sequence and structure databases well-known in bioinformatics research domain. Striking growth of databases such as GenBank [1] and KEGG [2] is evident from the plot, as number of DNA sequences rise from ~10^3 to ~10^8 in GenBank between years 1983-2014, and number of gene entries rise from 10^5~10^7 within years 1998-2016 in the KEGG database.

Consequently, bioinformatics approaches have been long sought as solutions that bridge the gap between the pace of whole-genome sequencing and revealing functional insights for the newly sequenced genes. Computational function prediction methods are also useful for analyzing protein function on a proteomic scale, such as interpreting high-throughput experiments including gene expression and protein-protein interaction

data, since these methods can be applied to a large number of proteins in a short time. As sequencing the whole genome of organisms becomes routine in experimental laboratories due to the rapid advancement of sequencing technologies, computational gene function prediction methods have become increasingly important.

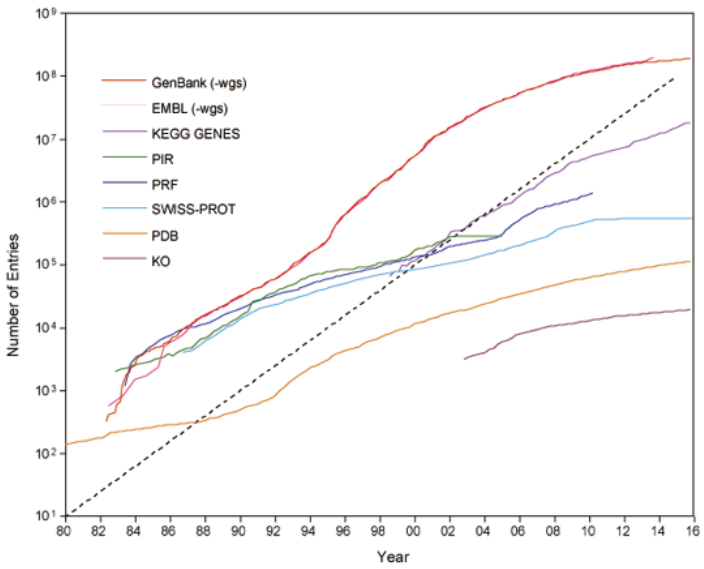

Figure 1.1 Growth of sequence and 3D structure databases

Yearly release information of KEGG data was obtained from GenomeNet (http://www.kanehisa.jp/en/db_growth.html)

1.2 Protein function prediction methods

The history of computational protein function prediction goes back to a very early stage of bioinformatics, when algorithms of sequence alignments and sequence database searches covered the major research problems in this area. From an evolutionary point of view, genes evolved from the same ancestor commonly retain sequence and functional similarity. Since protein sequence determines the tertiary structure of the protein, conventionally researchers have used protein sequence or structural similarity to transfer func-

tion information between proteins. Since structure-based methods rely on the availability of known structures of proteins, data that is quite scarce in the enormous genomic landscape, more often than not, the only available information on a functionally un-annotated protein is its sequence. Conventional homology-based function prediction methods can be summarized into three main categories: sequence-to-sequence comparison methods such as SSSEARCH [3], FASTA[4] and BLAST [5] extract functional annotations from top hit sequences which have a significant similarity score with the query. The second category of homology-based methods are profile-to-sequence comparison method such as PSI-BLAST[6], that iteratively construct a profile (multiple sequence alignment, MSA) with a target and retrieved sequences and uses it for the search in next iteration. Profiles can also be pre-computed for sequences in a database, and a target sequence is matched against them. This approach formulates the third category of sequence-based function annotation methods – sequence-to-profile comparison methods such as and BLOCKS [7], ProDom [8], PRINTS [9], Pfam [10] and InterPro [11].

Aside from the conventional homology-based function prediction methods, several advanced methods were developed that extract function information thoroughly from sequence database search results by making use of sequence-based features. Some of these methods have used machine learning tools such as Support Vector Machine (SVM) or Artificial Neural Network (ANN) as the backbone of their function prediction scheme. These methods include PFP [12,13], ESG [14], GOtcha [15], GOPET [16], OntoBlast [17], GOFigure [18], and ConFunc [19].

The homology driven function annotation methods have some shortcomings. There are cases where sequence similarity does not directly imply functional similarity (e.g.

gene duplication/paralogous genes). Also, homology driven annotation transfer leads to the percolation of miss-annotations in databases. Moreover, sequence data do not provide information on the biological context of protein functions. Such context driven function prediction can be performed using large-scale data on interactions (e.g. physical, genetic, co-expression) which are commonly represented as networks, with nodes representing proteins and edges representing the detected interactions.

Network based approaches were classified into two categories in a review by Sharan et. al. [41]: direct methods predict the functions of a protein from the known functions of its neighbors/interacting protein in the network. Module-based/indirect methods first identify function modules in the network and subsequently assign enriched function in the module to their un-annotated components. On the other hand, SIFTER [20], Flower-Power [21], and Orthostrapper [22] employ phylogenetic trees to transfer functions to target genes in the evolutionary context. There are other function prediction methods considering co-expression patterns of genes [23-27], 3D structures of proteins [28-36] as well as interacting proteins in large-scale protein-protein interaction networks [37-42].

1.3 Vocabulary for function prediction

For managing computational protein function prediction there is a need to trans-form the descriptive biological knowledge into a controlled and well-defined vocabulary. The Gene Ontology (GO) Consortium [43] of collaborating databases has developed a structured controlled vocabulary to describe gene function and currently serves as the dominant approach for machine-legible functional annotation. GO describes three aspects of gene product function: *molecular function*, *biological process* and *cellular location*.

Biological process (BP) terms indicate pathways and larger processes made up of the activities of multiple gene products. Examples of biological processes are *carbohydrate metabolism (GO:0003677), regulation of transcription (GO:0045449)*. Molecular functions (MF) represent activities carried out at molecular level by proteins or complexes, for example, *catalytic activity (GO:0003824)* or *DNA binding (GO:0003677)*. Cellular component (CC) indicates to which anatomical part of the cell the protein belongs to, for example, *ribosome (GO:0005840)* or *nucleus (GO:0005634)*. Thus each GO term has a category and an identifier in the format GO:xxxxxxx associated with it, along with a term definition to explain the meaning of the term. Each of the BP, MF and CC ontology is represented as a directed acyclic graph (DAG) where terms are represented as nodes in the graph and are arranged from general to specific. By standardizing an annotation and defining the relationships between terms using a graph, annotations may be computationally processed.

1.4 One protein multiple functions ─ Moonlighting protein

Automated protein function prediction methods are based on the concept of one protein involved in one function; hence conventionally AFP methods are based on sequence or structure homology. As the major focus of my research, I address two possible divergences from the “one protein – one function” concept for the first time that has inevitable impact on cellular processes: the first is the aspect of one protein having multiple functions, or moonlighting proteins, and the next is the aspect of group of proteins performing one function, described in the next subsection.

As the number of functionally characterized proteins increases, it has been observed that there are proteins involved in more than one function [44-46]. These proteins were described as “moonlighting” proteins [44]. Moonlighting proteins (MP) perform more than one independent cellular function within one polypeptide chain. Recent biological experiments have been discovering such multi-functional proteins at a steady pace. However, existing computational methods for automated function prediction (AFP) problem are aimed at identifying one, not multiple function of proteins; hence development of bioinformatics approaches for automatic identification of MPs has inevitable impact and novelty. Our work on moonlighting proteins can be divided into three logical parts:

1a. [47-49]: Development of a computational framework for comprehensive genome-scale characterization of moonlighting proteins based on functional and context-based information. Based on current knowledge of experimentally identified MPs, our work identifies characteristic features of MPs in both cases where current databases have functional annotations of the diverse functions of such proteins and cases when functional annotations do not exist. Different context-based protein association are explored for characterizing MPs apart from direct GO based results, such as protein-protein interaction (PPI), phylogenetic profile association, gene expression profile correlation, genetic interaction, protein’s structural features etc.

1b. [50]: Development of an automated prediction model of moonlighting proteins based on functional and context based features established in 1a. Our model applies machine learning classifiers to perform MP prediction on the diverse feature space. The model also addresses the missing feature problem commonly found in interaction networks, and imputes the features missing in protein databases through a iterative learning

algorithm. We show that we can identify MPs with very high accuracy when the functional annotations of the protein exist in the databases. More importantly, we show that our model can identify such proteins with high to moderate accuracy when functional annotations are absent in the database using network-based features and with incorporating missing feature prediction.

1c. As computational approaches for studying MPs are starting to emerge in the bioinformatics community, different facets of proteins: from sequence based properties, gene ontology (GO) to protein-protein interaction (PPI) have been considered. However, textual information associated to proteins have never been applied before to the automated identification of MPs. In the last part of my MP based work, we propose a novel method that extracts text information of proteins from scientific literature and applies text-mining techniques to provide automated MP prediction based on protein's textual features. Our developed model achieves high accuracy of MP prediction using different text-based features and shows that significant fraction of different genomes are predicted as MPs with sufficient high specificity over known MPs.

1.5 Function prediction of protein groups

The second part of this research addresses yet another divergence from the one-protein-one function paradigm. Proteins work together to achieve a common function in a cell. More often than not, biological experiments reveal sets of proteins involved in a disease/disorder, co-expressed together, or phylogenetically correlated together without sufficient explanation of the functional mechanisms of these group activities. Consequently, the computational challenge of correctly annotating protein's function and explaining the

mechanisms through which multiple proteins interact in a cell toward a common phenomenon becomes ever more important. Intuitively, proteins interact in a cell physically, through gene expression or genetic interaction to commemorate a common function that so often ends up causing a disease/disorder. To understand the functional nature of a set of proteins, it is often important to understand the biological process/molecular function/cellular location the proteins are involved in as a *group*, rather than understanding the detailed functional characteristics of the individual proteins in the group. My research aims to develop a computational model that predicts functions of *protein groups* based on protein's interaction networks.

Existing computational AFP methods aims at identifying individual functions of proteins, and there is no existing model that can identify protein's group function. Here we propose a model that takes groups of proteins found to work together in certain biological experiment, disease, or pathway, maps them to several functional linkage networks and integrates them, and then uses an iterative clustering and graphical modeling based schema to find group functions of the input proteins. As a backbone to the function prediction model of protein group, we use an integration of a number of major protein interaction networks. We propose a conditional random field (CRF)-based framework that predicts function of the "protein groups" in the network based on group neighborhood, and iteratively updates the function annotation of the unknown group members such that it reflects the protein's group activity. The perspective of "group" function annotation to a set of proteins opens up novel possibilities in understanding the functional nature of complex cellular interactions of protein groups.

1.6 Update on AFP methods and CAFA challenge

[51-53]: An essential task in bioinformatics is to propose and develop new tools and new ideas. However, to support the biology community, it is equally important to maintain and update previously-developed software tools so that users can continue using them. For a prediction method, it is important that the prediction accuracy be improved over time so that it can keep pace with other existing methods of the same type. The last part of my research copes with the AFP problem in three aspects: A. database update and improvement of methods previously developed in our group- PFP[12,13] and ESG [14], B. development of a web-server for the methods, and C. participation in a community-wide challenge for the AFP methods called CAFA (critical assessment of function annotation. We also develop two ensemble methods that combine GO predictions from multiple AFP models. We report benchmark performances of our updated methods and also performances of our component and ensemble methods in CAFA [54].

CHAPTER 2. MOONLIGHTING PROTEINS

2.1 Background

The first divergence from the "one protein – one function" concept that I address in my research are *moonlighting proteins*. With the overwhelming growth of genome sequence data produced by rapidly advancing sequencing technologies, the challenge of correctly determining functions of encoded proteins becomes ever more evident. As the number of functionally characterized proteins increases, it has been observed that there are proteins involved in more than one function [44-46]. These proteins were described as "moonlighting" proteins first by Jeffery [44]. A moonlighting protein demonstrates multiple autonomous and usually unrelated functions. Diversity of dual functions of these proteins is in principle not a consequence of gene fusions, splice variants, multiple proteolytic fragments, homologous but non-identical proteins, or varying post-transcriptional modification.

The first and the most widely known example of moonlighting proteins was identified by Piatigorsky and Wistow [55] who showed that crystallins, structural proteins in the eye lens, also have enzymatic activity. Crystallins in several mammals, geckos, birds, and some other species, are eye lens proteins that retain their metabolic functions, including lactate dehydrogenase, arginosuccinate lyase, and α-enolase [56-59]. Many known moonlighting proteins were originally recognized as enzymes, but there are also others

that were known as receptors, channel proteins, chaperone proteins, ribosomal proteins, and scaffold proteins [44,60,61]. The secondary/moonlighting functions of these proteins include transcriptional regulation, receptor binding, apoptosis-related, and other regulatory functions. A variety of causes have been found for the moonlighting activities of these proteins [44], including locations inside and outside of cell (e.g. thymidine phosphorylase [62]), different locations within a cell (putA proline dehydrogenase [63]), ligand binding sites (E. coli aspartate receptor [64]), oligomerization states (glyceraldehyde-3-phosphate dehydrogenase [65]), differential expressions (neuropilin [66]), and ligand concentration (aconitase [67]).

As long as the additional functions do not interfere with the primary function, moonlighting functions can benefit a cell in several ways. Especially in prokaryotes, existence of multifunctional proteins aids in saving energy in cell growth and reproduction and makes their genomes more compact. Moonlighting proteins can also help in coordinating cellular activities in signaling pathways, transport, biosynthesis, and other functions [68]. It has been suggested that the presence of moonlighting proteins is under positive selection [44,61,69].

Recent papers [61,70] indicate that a number of moonlighting proteins in mammals play important roles in cellular activities and biochemical pathways that are involved in cancer and other diseases. Sriram et al. discussed how moonlighting functions may contribute to the complexity of metabolic disorders [71]. The positive selective pressure for developing moonlighting functions and the cell-level benefits given by moonlighting proteins suggest that the existence of moonlighting proteins in diverse genomes might be a common phenomenon.

2.2 Current computational analysis on MP

The functional diversity of moonlighting proteins pose a significant challenge to computational protein function annotation as current methods do not explicitly consider the possibility of dual functions for a protein. Conventional sequence-based functional annotation methods that are based on the concept of homology [6] or conserved motifs/domains [72-74] will have problems for identifying secondary functions because there are cases that a homolog of a moonlighting protein does not possess the secondary function [75] or has a different secondary function [67,76,77]. There are two studies that have investigated whether existing sequence-based function prediction methods can identify distinct dual functions of moonlighting proteins [49,78]. Gomez et al. compared eleven methods and reported that PSI-BLAST [6] performed relatively well in identifying moonlighting functions [78]. We have compared our function prediction tools, PFP and ESG [14], with PSI-BLAST and showed that PFP, which mines function information from weakly similar sequences, had the best performance in predicting two distinct functions of moonlighting proteins [49]. These two studies suggest that secondary functions may be found in distantly related sequences if not in close homologs; however, further investigation is needed because the studies are based on a limited dataset. Gomez et al. have also analysed protein-protein interactions (PPIs) of moonlighting proteins and showed that GO terms of secondary function are enriched in interacting proteins, although they concluded that predicting correct secondary function from a PPI network is not an easy task [79]. Becker et al. [80] analysed Human PPI network and developed a novel clustering method that can decompose a network into multiple overlapping clusters. They reported that proteins that belong to the overlapping clusters are more central in the

network compared to mono-clustered proteins and contain multiple domains; hence they are candidates for multitasking proteins. Studies also explore different aspects of moonlighting proteins using intrinsically disordered region, functional motif/domains and correlated mutations [81,82]. Currently, there are two publicly available online databases for multifunctional/moonlighting proteins[83,84]. Computational works on moonlighting proteins were recently summarized in a review article [47].

2.3 Performance evaluation of AFP methods on MP prediction

In this work, we have analyzed the ability of existing function prediction methods to correctly identify diverse functions of experimentally identified moonlighting proteins [69]. We have collected Gene Ontology (GO) term annotations of these proteins from the Uniprot database and manually classified these annotations into two distinct functions. Based on the GO annotations, we have examined the prediction performance of PSI-BLAST and two other major sequence based function prediction methods, the Protein Function Prediction (PFP) and the Extended Similarity Group (ESG) method.

Overall, PFP showed higher average recall than PSI-BLAST and ESG. ESG showed lower recall as compared with PFP and PSI-BLAST, although it has a higher precision. The results suggest that the functional diversity of the moonlighting proteins can be captured if weakly similar sequences are considered among a broad range of similar sequence sets.

2.3.1 Methods

In this section we briefly describe the three AFP methods we examined, PFP, ESG, and PSI-BLAST, for computational prediction of moonlighting proteins.

2.3.1.1 Protein Function Prediction (PFP) algorithm

The PFP algorithm uses PSI-BLAST to obtain sequences hits for a target sequence and predict GO function annotations. PFP computes the score to GO term f_a as follows:

$$s(f_a) = \sum_{i=1}^{N} \sum_{j=1}^{Nfunc(i)} \left(\left(-\log(E - value(i)) + b \right) P(f_a \mid f_j) \right), \qquad \text{(Eq. 2.1)}$$

where N is the number of sequence hits considered in the PSI-BLAST hits, $Nfunc(i)$ is the number of GO annotations for the sequence hit i, $E\_value(i)$ is the PSI-BLAST E_value for the sequence hit i, f_j is the j-th annotation of the sequence hit i, and constant b takes value $2\ (= log_{10}100)$ to keep the score positive when retrieved sequences up to E_value of 100 are used (so that $-log_{10}(100) + 2 = 0$, when E_value = 100). The conditional probabilities $P(f_a|f_j)$ is to consider co-occurrence of GO terms in single sequence annotation, which is computed as the ratio of number of proteins co-annotated with GO terms f_a and f_j as compared with genes annotated with the term f_j. To take into account the hierarchical structure of the GO, PFP transfers the raw score to the parental terms by computing the proportion of proteins annotated with f_a relative to all proteins that belong to the parental GO term in the database. The score of a GO term computed as the sum of the directly computed score by Eqn. 2.1 and the ones from the parental propagation is called the raw score.

2.3.1.2 Extended Similarity Group (ESG) algorithm

ESG recursively performs PSI-BLAST searches from sequence hits obtained from the initial search from the target sequence, thereby performing multi-level exploration of the sequence similarity space around the target protein. Each sequence hit in a search is assigned a weight that is computed as the proportion of the -*log*(*E_value*) of the sequence relative to the sum of -*log*(*E_value*) from all the sequence hits considered in the search of the same level and this weight is assigned for GO terms annotating the sequence hit. The weights for GO terms found in the second level search are computed in the same fashion. Ultimately the score for a GO term is computed as the total weight from the two levels of the searches. The score for each GO term ranges from 0 to 1.0.

2.3.1.3 PSI-BLAST algorithm

PSI-BLAST search is performed with a default setting with maximum of three iterations. Then the top hits with an E_value score better than 0.01 that have annotations is used for transferring annotation to the query sequence. The BLAST predictions were ranked according to –log(E_value)+2 for each of the prediction.

2.3.2 Results

We analyze the performances of PFP, ESG and PSI-BLAST in predicting the functional diversity of the moonlighting proteins. The 19 moonlighting proteins were taken from review article [69]. These proteins have two diverse and distinct functions. According to the verbal description of the two diverse functions of the proteins, we classi-

fied GO terms assigned to these proteins from Uniprot into four classes: Terms that belong to the major moonlighting function of the protein (Function 1); those which belong to the second moonlighting function (Function 2); terms which belong to both functions; and terms that do not belong to either of the functions.

The raw score of PFP predictions has a large range of values. Up to 1000 GO term predictions were sorted by their raw score and plotted at an interval of 10. ESG predictions have a score range of 0 to 1.0, and 100 cutoffs are used within this range. PSI-BLAST predictions are ranked by -log(E_value)+2, and 100 score cutoffs are used from 4 (E_value of 0.01) to 45 (E_value of 10-43). To compare the prediction performances of the methods, we computed precision and recall. Precision is defined as TP/(TP+FP) and recall is defined as TP/(TP+FN), where TP and FP denote true and false positive, respectively, and FN denote false negative. All predictions by the three methods are propagated to the root of the GO hierarchy, so are the true annotations for the proteins.

2.3.2.1 Average precision recall of PFP, ESG, and PSI-BLAST

In Figure 2.1, the performance of PFP, ESG, and PSI–BLAST in terms of the average precision and recall for all the GO terms of the 19 moonlighting proteins are shown. Figure 2.1 shows that ESG predictions perform significantly better than the other two methods in the recall range of 0.4 – 0.7. ESG has better precision than BLAST

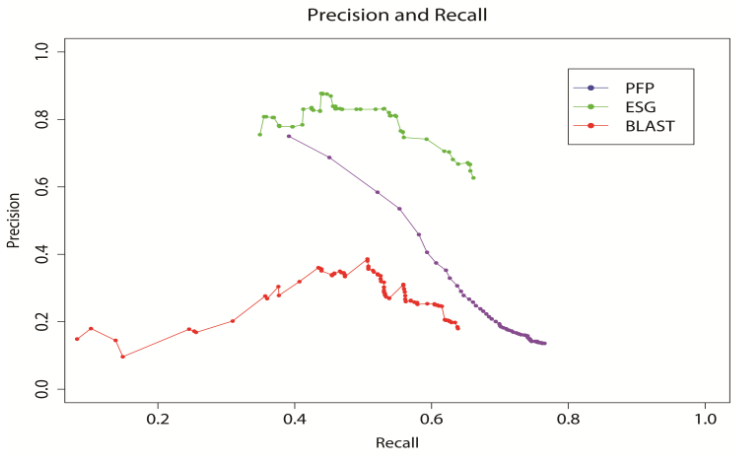

Figure 2.1 Precision recall of PFP, ESG and PSI– BLAST

within recall range of 0.37 – 0.66. PFP predictions ranked with raw score (Eq. 2.1 in Methods) reaches the highest recall. In Figure 2.2 we show the performance of the methods in terms of recall values of the methods at 100 cutoff scores (with all the GO annotations of the proteins considered). It is apparent from this plot that PFP showed higher recall than PSI-BLAST, and ESG. ESG has lowest recall within the cutoff range of 0.09-0.88.

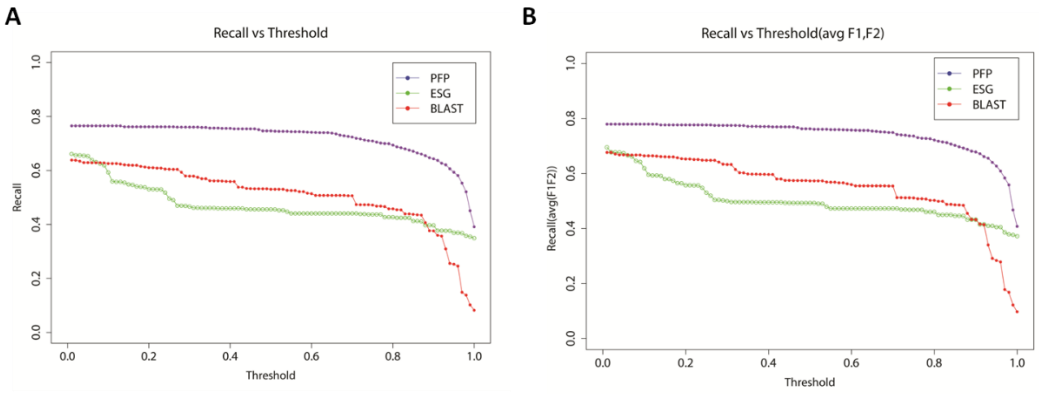

Figure 2.2 Recall of PFP, ESG and PSI–BLAST at each threshold

A, *Recall where all the GO annotations for proteins are considered.*

B, *Recall where only the GO annotations labeled as Function 1 or Function 2 for proteins are considered.*

In Figure 2.2B, the performance was evaluated where only the GO annotations for the two moonlighting functions (Function 1 and Function 2) are taken into account as the target annotations. The prediction performance for the moonlighting functions is essentially the same as those measured for the all GO term annotations (Fig. 2.2A).

2.3.2.2 Recall at individual proteins

Next In Figure 2.3, we plotted the recall for the three methods for each of the 19 moonlighting proteins separately. The cutoff of the prediction scores used are 0.5 for PFP, 0.35 for ESG, and E_value 0.01 for PSI-BLAST. The PFP cutoff of 0.5 will yield the maximum of 500 GO term predictions. The score cutoff value of 0.35 for ESG is an optimal cutoff score established in the previous work [14]. E_value 0.01 for PSI-BLAST is a standard cutoff used in general for homology search. We added the predictions of two more versions of PSI-BLAST, with BLOSUM45 and BLOSUM30 scoring matrices (BL+bls45 and BL+30 in Figure 2.3, respectively) to consider more divergent sequences in the homology search. PSI-BLAST uses BLOSUM62 as the default scoring matrix.

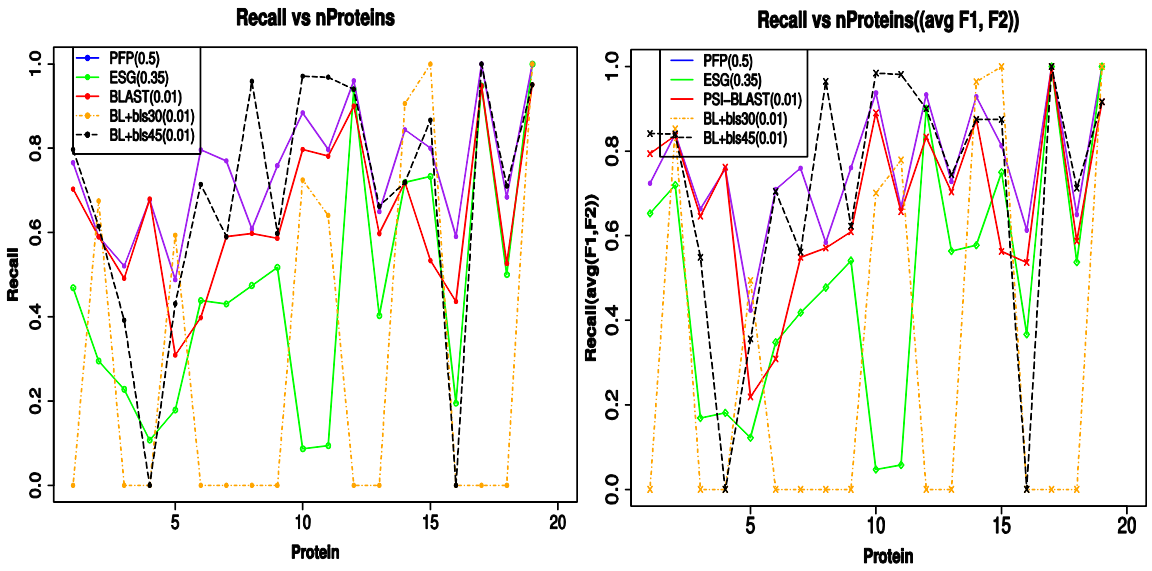

Figure 2.3 Recall of PFP, ESG, PSI–BLAST with different BLOSUM *matrix*

A, *Recall where all the GO annotations for proteins are considered.*

B, *Recall where only the GO annotations labeled as Function 1 or Function 2 for proteins are considered.*

When all the GO terms are considered (Fig. 2.3A), PFP showed higher recall than PSI-BLAST for almost all the cases (except for proteins 2 and 4, which are ties). ESG has similar recall of predictions as PSI–BLAST for proteins 14 and 17, slightly higher recall for proteins 6, 12 and 15 than PSI-BLAST, and a lower recall than PFP and PSI-BLAST for the rest of the proteins. PSI-BLAST with BLOSUM45 remains lower or equal in recall values than PFP for most of the cases except for 3 proteins where BL+bls45 wins over the others. BL+bls30 fails to predict any GO terms above E_value of 0.01 for many proteins. Overall, PFP shows the highest overall recall than ESG and PSI-BLAST with different scoring schemes. We see a similar performance pattern for the three methods when we consider only the GO terms belonging to moonlighting function 1 and function 2 of the proteins (Fig. 2.3B).

These results indicate that the PFP can find moonlighting GO terms that are missed by regular PSI-BLAST searches for quite a lot of cases. The strength of PFP is its coverage of a large number of sequences, by including weakly similar sequences into consideration for annotation transfer. On the other hand, ESG puts more weight on the consensus sequences that have strong similarity with the query protein among all the sequences that it encounters along multiple iterations. So although ESG provides a higher precision on the predictions among all three methods (Fig. 2.1), it fails to detect the functional variations in a number of cases. These results suggest that the functional diversity of the moonlighting proteins could be captured by using weakly similar sequences are considered among a broad range of similar sequences.

2.4 Genome-scale identification and characterization of MPs

Despite the potential abundance of moonlighting proteins in various genomes and their important roles in pathways and disease development, systematic studies of moonlighting proteins are still in their early stage for obtaining a comprehensive picture of proteins' moonlighting functions and also for developing computational methods for predicting moonlighting proteins. The limited number of known moonlighting proteins is mainly because secondary functions of proteins are usually found unexpectedly by experiments. To lay the foundation for studying moonlighting proteins, the current work is aimed at establishing a framework for systematically identifying moonlighting proteins in an organism using currently available function annotations and omics-scale data. This work consists of two logical parts. First, we examined Gene Ontology (GO) annotations [43,85] of known moonlighting proteins in the UniProt protein sequence database [86] to

see if functional diversity of moonlighting proteins is reflected in current GO annotations. Since the systematic study of moonlighting proteins is still in an early stage, most of the cases they are not explicitly labelled in the database as “moonlighting”, “dual function”, “multitasking”, or related words, which makes it difficult to collect and reuse existing knowledge of moonlighting proteins. We analyzed the GO terms assigned to each known moonlighting protein and found that the GO term semantic similarity score can clearly separate the GO terms of the diverse functions of these proteins. Encouraged by this result, we further analyzed the GO term annotations of protein genes in the *Escherichia coli* K-12 genome and found 33 novel moonlighting proteins by identifying genes with clear GO term separations. We confirmed in literature that the dual functions of the identified proteins had experimental evidence. Among our computationally identified moonlighting proteins, we later found that DegP was experimentally identified as a moonlighting protein with both protease and chaperone activity [87-89], which confirmed that our procedure was valid.

In the second part of this work, we investigated characteristics of moonlighting proteins in omics-scale data, namely, protein-protein interaction, gene expression, phylogenetic profile [90], and genetic interactions [91]. We decided to analyze these omics-scale data because moonlighting proteins’ distinct functions may display characteristic features in association patterns with other proteins. In analyzing protein-protein interactions, we found that moonlighting proteins interact with a higher number of distinct functional classes of proteins than non-moonlighting ones, which intuitively stems from the functional diversity of these proteins. We found a substantial number of moonlighting proteins in the PPI network of moonlighting proteins, suggesting moonlighting proteins

tend to interact with other moonlighting proteins. It is also notable that moonlighting proteins share their primary functions with the majority of interacting proteins. Similarly, a weak tendency was found that moonlighting proteins interact with proteins from more diverse functional classes in gene expression and phylogenetic profile networks. We have further examined structural features of proteins, i.e. ligand binding sites and disordered regions. We analysed disordered regions and found that a larger fraction of moonlighting proteins have intrinsically disordered regions than non-moonlighting proteins. Finally, although there are only a few moonlighting proteins whose tertiary structures were available, we found cases where the binding sites that correspond to distinct functions are located in separate regions of the proteins' tertiary structures.

2.4.1 Methods

2.4.1.1 Dataset of known MPs

We constructed three datasets of experimentally confirmed moonlighting proteins from two review articles [44,69] and papers we collected from the PubMed database. They are called the MPR1 [75] [69], MPR2 [92] [44], and MPR3 (16) set, respectively. In the parentheses is the number of moonlighting proteins in the each dataset. The MPR1 dataset was used in our previous study [49]. The three datasets are available at http://kiharalab.org/MoonlightingDatasets. The list of proteins in the MPR3 set is provided in Table A.3. In MPR1 and MPR2, we found four proteins (ATF2, PutA, neuropilin-I, and BirA) are multi-domain proteins. Although these four proteins are also listed as moonlighting proteins in MultitaskProtDB and MoonProt, we excluded them from the

dataset in all the results except for the bar graphs in Fig. 2.5 and Fig. 2.9 where these proteins are noted with asterisk (*). For each of the moonlighting proteins in the three datasets, GO term annotations in UniProt were classified into four classes by referring to textual description of the protein's function in literature: GO annotations that described the "primary" function of the protein (Function 1, F1), GO annotations that describe "secondary" function (Function 2, F2), GO annotations that correspond to both functions of the protein (usually general GO terms at a higher depth of the GO hierarchy), and lastly, GO annotations whose functional association to either of the two functions were unclear. In cases that the description of the secondary function of a moonlighting protein was absent or incomplete in UniProt, we annotated the protein with appropriate GO terms selected from the GO database.

2.4.1.2 Semantic similarity & funsim score

We used the relevance semantic similarity score (SS^{Rel}) [93] for computing functional similarity of a pair of GO terms, c_1 and c_2:

$$SS^{Rel}(c_1, c_2) = \max_{c \in S(c_1, c_2)} \left(\frac{2 \log p(c)}{\log p(c_1) + \log p(c_2)} (1 - p(c)) \right) \quad \text{(Eqn. 2.2)}$$

Here $p(c)$ is the probability of a GO term c, which is defined as the fraction of the occurrence of c in the GO Database [43,85]. The root of the ontology has a probability of 1.0. $s(c_1,c_2)$ is the set of common ancestors of the GO terms c_1 and c_2. The first term considers the relative depth of the common ancestor c to the depth of the two terms c_1 and c_2 while

the second term takes into account how rare it is to identify the common ancestor *c* by chance.

To quantify the functional similarity of two proteins, both of which are annotated with a set of GO terms, we used the funsim score [93]. The funsim score of two sets of terms, GO^A and GO^B of respective size of *N* and *M*, is calculated from an all-by-all similarity matrix s_{ij}.

$$s_{ij} = sim(GO_i^A, GO_j^B)_{\forall i \in \{1..N\}, \forall j \in \{1..M\}} \quad \text{(Eqn. 2.3)}$$

$sim(GO_i^A, GO_i^B)$ is the relevance similarity score for GO_i^A and GO_j^B. Since the relevance similarity score is defined only for GO pairs of the same category, a matrix is computed separately for the three categories, Biological Process (BP), Molecular Function (MF), and Cellular Component (CC). Then, the GOscore of the matrix of each GO category is computed as follows:

$$GO_{score} = \max\left(\frac{1}{N} \sum_{i=1}^{N} \max_{1 \le i \le M} s_{ij}, \frac{1}{M} \sum_{i=1}^{M} \max_{1 \le i \le N} s_{ij} \right) \quad \text{(Eqn. 2.4)}$$

GOscore will be any of the three category scores (MFscore, BPscore, CCscore). Finally the funsim score is computed as

$$funsim = \frac{1}{3}\left[\left(\frac{MFscore}{\max(MFscore)} \right)^2 + \left(\frac{BPscore}{\max(BPscore)} \right)^2 + \left(\frac{CCscore}{\max(CCscore)} \right)^2 \right]$$

(Eqn. 2.5)

where *max(GOscore)* = 1 (maximum possible GOscore) and the range of the funSim score is (0,1).

2.4.2 Results

2.4.2.1 Pairwise GO semantic similarity analysis

We investigated whether the distinct dual functions of moonlighting proteins were reflected in their GO term annotations. We used 58 experimentally confirmed moonlighting proteins in three datasets (see Materials and Methods). We classified the GO terms of these proteins into four classes: GO terms that belong to the "primary" function of the protein (Function 1, F1), terms that belong to the "secondary" function (Function 2, F2), terms that belong to both functions, and terms that do not belong to either of the functions. For each moonlighting protein, we computed the relevance semantic similarity score (SS^{Rel}, Eqn. 2.2) for three types of GO term pairs: pairs where both terms belong to either F1 or F2 and pairs that consist of one GO term from F1 and the other from F2. SS^{Rel} ranges from 0.0 to 1.0 with 0.0 for the least similarity and 1.0 for the highest similarity.

Figure 2.4 shows an example of the semantic similarity of GO pairs for aconitase in yeast (UniProt ID: P19414). This protein was initially identified as an enzyme in the tri-carboxylic acid (TCA) cycle, which catalyzes the isomerization of citrate to iso-citrate via cis-aconitate. The GO terms for F1 include TCA cycle (GO:0006099), propionate metabolic process (GO:0019541), glutamate biosynthetic process (GO:0006537), citrate metabolic process (GO:0006101), cytosol (GO:0005829), cytoplasm (GO:0005737), citrate hydro-lyase (GO:0052632), lyase activity (GO:0016829), iso-citrate hydro-lyase (GO:0052633) and aconitate hydratase activity (GO:0003994). The enzyme's secondary function (F2) was later found as a "role in mitochondrial DNA maintenance" [76], which

is annotated with GO terms including mitochondrial genome maintenance (GO:0000002), mitochondrial nucleoid (GO:0042645), single-stranded-DNA binding (GO:0003697), and double-stranded-DNA binding (GO:0003690). The GO terms that belong to both the primary and secondary functions (F3) are “mitochondrion” and “mitochondrial matrix” (GO:0005759). Figure 2.4A shows the SS^{Rel} score distribution of GO term pairs, those within F1 or F2 and pairs across F1 and F2 (F1F2 pairs). It is apparent that the SS^{Rel} scores for all the F1F2 pairs are very small, below 0.2. All four F2 pairs have large scores over 0.4. As for F1 pairs, 8 out of 27 have large scores over 0.4. We must note that 12 F1 pairs have a score of 0, which occurs when the lowest common ancestor for a GO term pair is at the root of the GO hierarchy. In the case of aconitase, the majority of the 0 scores for F1 pairs occurred between terms related to ion-sulfur cluster binding and aconitase hydrolase (Fig. 2.4B).

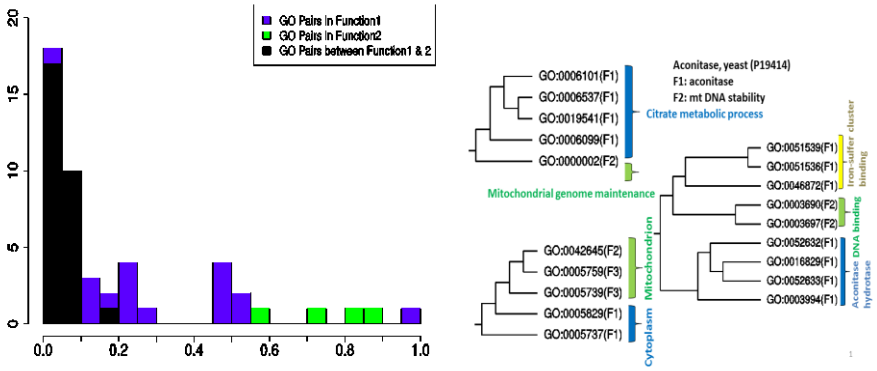

Figure 2.4 Semantic similarity distribution on MPs

The distribution of the relevance semantic similarity SSRel score of GO term pairs, aconitase, yeast (Uniprot ID : P19414).

***A,** SSRel distribution of GO pairs within the primary function (function 1), the secondary function (function 2), and pairs from function 1 and 2.*

***B**, Hierarchical clustering of GO terms in the three GO categories using pairwise SSRel scores.*

Figure 2.4B shows a hierarchical clustering of GO terms of aconitase based on SS^{Rel}. In all three GO categories, terms in F1 and F2 were clearly separated. In the Biological Process (BP) ontology, the only GO term in F2 is "mitochondrial genome maintenance" (GO:0000002), which is separated from the other F1 GO terms. In the Molecular Function (MF) ontology, the GO terms with F2 labels (ssDNA and dsDNA binding, GO:0003697 and GO:0003690, respectively) form a cluster that is separate from the F1 GO terms. Two separate clusters were formed for F1 terms in MF, "Iron-Sulfer cluster binding" GO terms (highlighted in yellow) and terms related to aconitase enzymatic activity. The former F1 cluster lies closer to the F2 cluster due to a common ancestral term "binding". In the Cellular Component (CC) ontology, the F2 term "mitochondrial nucleoid" (GO:0042645) is separate from F1 GO terms (related to cytoplasm) but clustered with two F3 terms.

Next, we show the mean SS^{Rel} score for GO pairs within F1 or F2 and across F1and F2 for all moonlighting proteins in the three datasets (Fig. 2.5). The mean SS^{Rel} scores for F1 pairs and F2 pairs are higher than those for across F1F2 pairs in 51 (87.9%) moonlighting proteins (MPR1-3 datasets). One exception of this trend is Protein 17 in MPR1 (Fig. 2.5A). This protein is aconitase of *Mycobacterium tuberculosis* (UniProt ID: O53166), which has "TCA cycle enzyme" as F1 and "iron-responsive protein" as F2. This protein switches between the two functions depending on the cellular iron levels, namely, binding of a 4Fe-4S cluster occurs as a part of the aconitase function whereas binding of a 3Fe-4S cluster triggers the secondary function [67]. Thus, the GO term for "4 iron, 4 sulfur cluster binding" (GO:0051539) was classified for F1 and "3 iron, 4 sul-

fur cluster binding" (GO:0051538) for F2, which resulted in a relatively high SS^{Rel} score of 0.698 for this F1F2 pair.

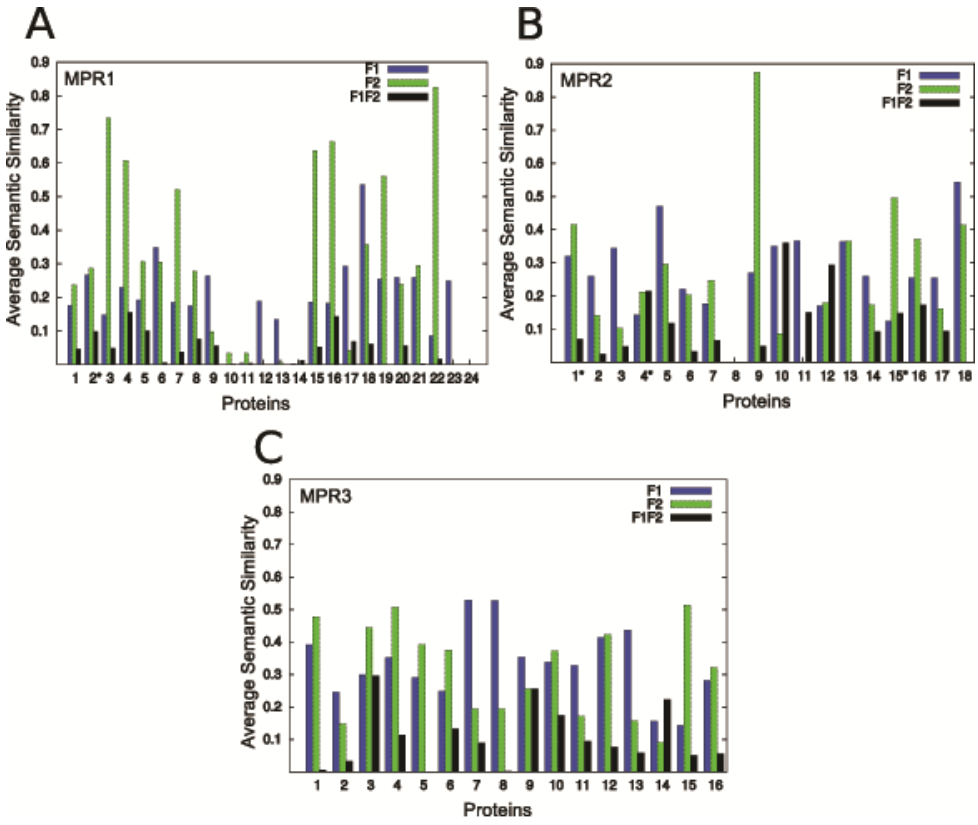

Figure 2.5 Average SS^{Rel} of GO term pairs for MPs

Average SSRel of GO pairs within function 1, function 2, and pairs from function 1 and 2 were computed separately.

A, *Moonlighting proteins in the MPR1 set. Protein 24 is presenilin in Physcomitrella patens (Uniprot ID: A9S846). This protein have one GO term each in F1 and F2 (F1 term GO:0004190, "aspartic type endopeptidase activity" and F2 term GO:0016021, "intergral to membrane"). The two GO terms are in different ontologies, MF and CC respectively, and thus the score are zero for F1 and F2 (because there is only one term) as well as F1-F2 (because similarity of GO terms in different categories cannot be considered).*

B, *the MPR2 set; and*

C, *the MPR3 set.*

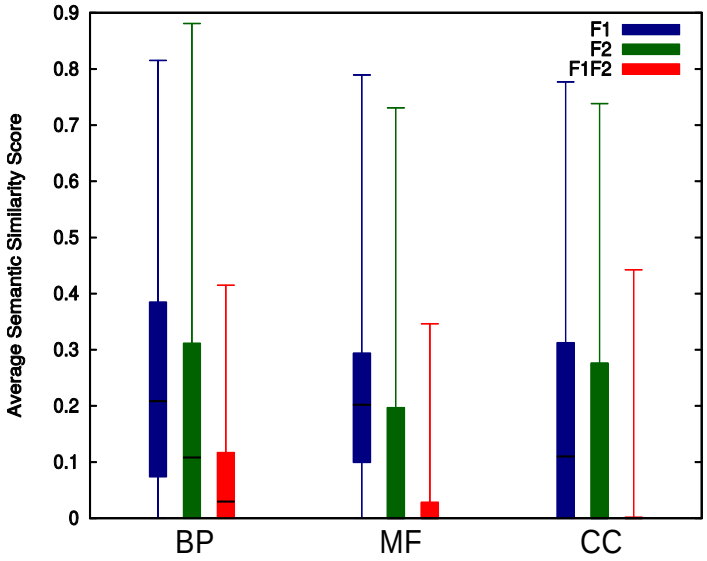

Figure 2.6 Average SS^{Rel} distribution of MP

*Box-and-whisker plots for average SSRel distribution of BP, MF, and CC GO pairs for the moonlighting proteins in the MPR1-3sets excluding proteins with * in Figure 2.5. The top and the bottom of a box show the first and third quartiles and the line in the middle of a box is the median. The two ends of whisker show the minimum and the maximum values.*

Figure 2.6 summarizes the distribution of the average SS^{Rel} score for F1, F2, and F1F2 GO pairs in the BP, MF, and CC ontologies for the proteins in MPR1-3. The Friedman test was performed to evaluate statistical significance of score difference between F1, F2, and F1F2 GO term pairs. It was shown that the F1F2 pairs have significantly smaller scores than F1 and F2 pairs in BP and CC (p-value < 0.05). As for MF, the score difference of F1F2 pairs from F1 pairs had a p-value below 0.05 but the p-value versus F2 pairs was a slightly larger value of 0.097.

2.4.2.2 Novel prediction in *Escherichia coli* genome

The previous section showed that GO terms of moonlighting proteins can be clustered into distinct functions using the SS^{Rel} score. In this section we identified potential

moonlighting proteins in the *Escherichia coli* K-12 genome by examining clusters of GO term annotations taken from UniProt. We used GO terms of the BP ontology because BP GO terms showed a clearer separation between F1 and F2 functions (Fig. 2.6).

Figure 2.6 shows clustering profiles of moonlighting proteins, where terms in BP and MF (Fig. 2.7A and B) GO were clustered using single linkage clustering at different SS^{Rel} cutoff values. A clustering profile provides a more thorough picture of GO term similarities than clustering using a single cutoff value. It can show how the number of clusters grows at different cutoff values. Using the profiles for moonlighting proteins in MPR1 (black), MPR2 (red), and MPR3 (green) as a reference, the following three criteria were used to identify potential moonlighting proteins in *E. coli*: 1) proteins that have at least eight GO terms in the UniProt annotation; 2) proteins that have at least two clusters in the clustering profile at a SS^{Rel} cutoff of 0.1; 3) proteins that have at least four clusters in the clustering profile at a 0.4 SS^{Rel}. 140 proteins were found to satisfy all of these three criteria. We have also identified potential non-moonlighting proteins by applying essentially the opposite criteria to above: 1) proteins that have at least eight GO terms in the UniProt annotation; 2) proteins that have at most one cluster at a SS^{Rel} of 0.1; 3) proteins that have at most one cluster at 0.4 SS^{Rel}. There were 150 proteins that satisfied these criteria for non-moonlighting proteins.

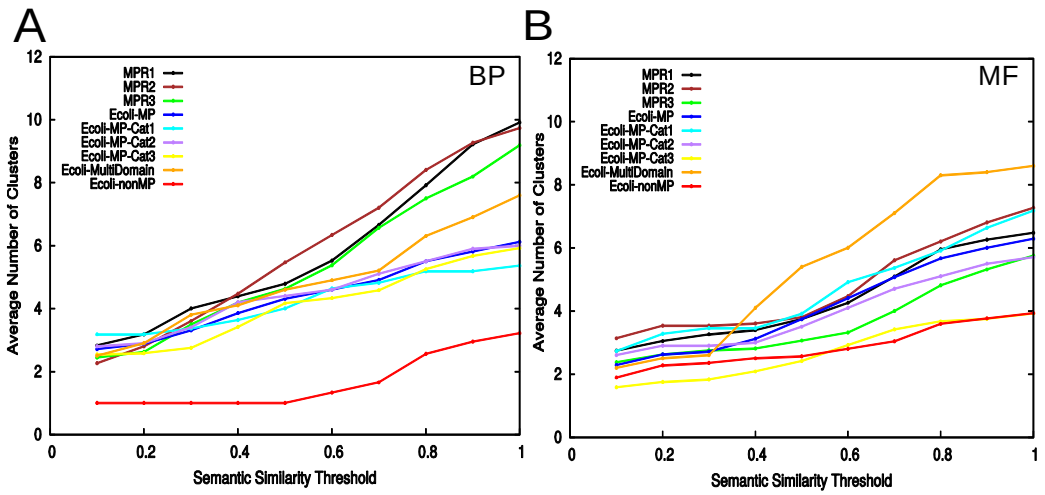

Figure 2.7 Clustering profiles of sets of MP and non-MPs

For each protein in a dataset, GO terms were clustered using various threshold values of SSRel and average number of GO term clusters were plotted. The datasets plotted were experimentally known moonlighting proteins (MPR1, 2, and 3) and identified moonlighting and non-moonlighting proteins in E. coli (Ecoli-MP and Ecoli-nonMP). E. coli moonlighting proteins were also plotted separately for each evidence category, 1 to 3 (Ecoli-PosMP-Cat1-3; see Materials and Methods) as well as multi-domain multi-function proteins.

A, *BP GO terms were considered.*

B, *MF GO terms were considered.*

For the 140 identified potential moonlighting proteins, we manually consulted original literature to determine the level of experimental support for annotated functions and whether diverse functions are directly related to each other. This literature check step has selected 43 proteins that have distinct dual functions. Subsequently, we used the Pfam database [72] to find domains in the 43 proteins in order to distinguish proteins whose multi-functionality originates from different domains. GO terms associated with each Pfam domain in a protein were compared with the primary and secondary functions of the protein. Finally, 33 proteins were selected as moonlighting proteins through this post-processing (Table A.1). The selected moonlighting proteins were further classified them into three categories. The first category is for moonlighting proteins that have clear

experimental evidence for two independent functions. The second category is proteins for which we found literature evidence of two diverse functions, but no evidence was found as to whether those two functions are independent or related. The third category is for "weak" moonlighting proteins for which the evidence for the secondary function was found from a large scale assay or a phenotypic experiment of mutants and the relationship between the primary and the newly found secondary function is not known. We would like to note that some of the moonlighting proteins classified into the second or the third category are so-called neomorphic moonlighting proteins [70], which exhibit the secondary function due to a mutation or conformational change.

Table A.2 lists ten multi-functional and multi-domain proteins that were excluded from by the Pfam domain search the final list of moonlighting proteins. These proteins happen to include five multi-reaction enzymes, which are enzymes that are generally listed as bi-functional or multi-functional proteins in UniProt and in literature. They perform multiple reactions with similar substrates in the same or different pathways. A multi-reaction enzyme is not included as a moonlighting protein in the original definition [69]. However, they are kept here along with the five other multi-domain proteins in Table A.2 because they were detected by the GO clustering criteria.

The identified 33 moonlighting proteins (Table A.1) and 10 multi-domain multi-function proteins (Table A.2) do not have many overlap with the MoonProt database [84] and MultitaskProtDB [83]. Only two (PepA and DegP) in Table A.1 and one (NadR) in Table A.2 were found in the two databases.

Among the 140 proteins that were identified by the GO clustering criteria, 97 (69.3 %) of them were discarded later by the literature survey. The discarded proteins sat-

isfied the three GO term clustering criteria but either a) the sufficient number of GO term clusters was due to a non-descriptive GO term at a high (general) level of the GO hierarchy such as “transport” or “biosynthesis”, which resulted in a small similarity scores with the other GO terms; or b) experimental evidence of GO terms were found in literature only for one of its functions but not the other. Proteins discarded by the latter reason may be confirmed as moonlighting proteins in the future when experimental evidence is made available.

Clustering profiles of the identified moonlighting and non-moonlighting proteins in *E. coli* are shown in Figure 2.7 in comparison with the MPR1-3 datasets. Three categories of moonlighting proteins as well as multi-domain multi-functional proteins were also separately plotted. Clearly, the number of GO term clusters for moonlighting proteins is higher than non-moonlighting proteins for both BP and MF. In the MF ontology, the multi-domain multi-functional proteins have a larger number of clusters than the rest for high cutoff values of over 0.4. The two-sample Kolmogorov-Smirnov (KS) test showed that the *E. coli* moonlighting proteins (Ecoli-PosMP in Fig. 2.7) and the MPR1-3 sets have significantly larger numbers of clusters than the *E. coli* non-moonlighting proteins (Ecoli-NegMP) at the three semantic similarity thresholds, 0.1, 0.5, and 1.0 for the BP ontology (Fig. 2.7A) (p-values < 0.05). As for the MF ontology, *E. coli* moonlighting proteins have significantly larger number of clusters than the *E. coli* non-moonlighting proteins at threshold 1.0, using a p-value cutoff of 0.05. The full results of the KS tests are provided in Table A.4.

It was noticed that known moonlighting proteins in the MPR1-3 sets have more GO annotations than the *E. coli* moonlighting proteins, which is a part of the reason why the MPR1-3 sets have more GO clusters (Fig. 2.7). The average number of BP GO annotations of the *E. coli* moonlighting proteins was 5.76 while the MPR1-3 proteins had 9.65 terms. The clustering profile analysis can identify new moonlighting proteins from their existing GO annotations in UniProt. However, a limitation is that candidate proteins need to be well annotated with a sufficient number of GO terms. Indeed only 29.1% of *E. coli* proteins have eight or more GO terms and were subject to this analysis. In the subsequent sections, we will explore different ways to identify potential moonlighting proteins that do not require GO annotations.

2.4.2.3 Protein-protein interaction network of MPs

From this section, we examine characteristic features of moonlighting proteins in large-scale omics data. We begin with the protein-protein interaction (PPI) network. Interacting proteins tend to share common function and thus a PPI network can be used as a valuable source for predicting protein function [94]. It was also shown that PPI networks are helpful in detecting additional novel function of well-known proteins [95]. We obtained physically interacting proteins from the STRING database [96].

First, we examined the number of interacting proteins of moonlighting and non-moonlighting proteins (Fig. 2.8A). In addition to the *E. coli* moonlighting and non-moonlighting proteins, histograms for the MPR1-3 sets are shown for comparison. Among the *E. coli* MP set, 11 proteins in the first category (those that have clear experi-

mental evidence of their dual functions) were also separately plotted to verify that the observed trend for the entire *E. coli* MP set was consistent with its most reliable subset. Overall MP and nonMP have similar distributions with the largest peak at 0-5 interacting proteins. A small peak at 20-25 interacting proteins was observed for *E. coli* MP. This peak consists of two proteins, pepA (P68767) and frdB (P0AC47).

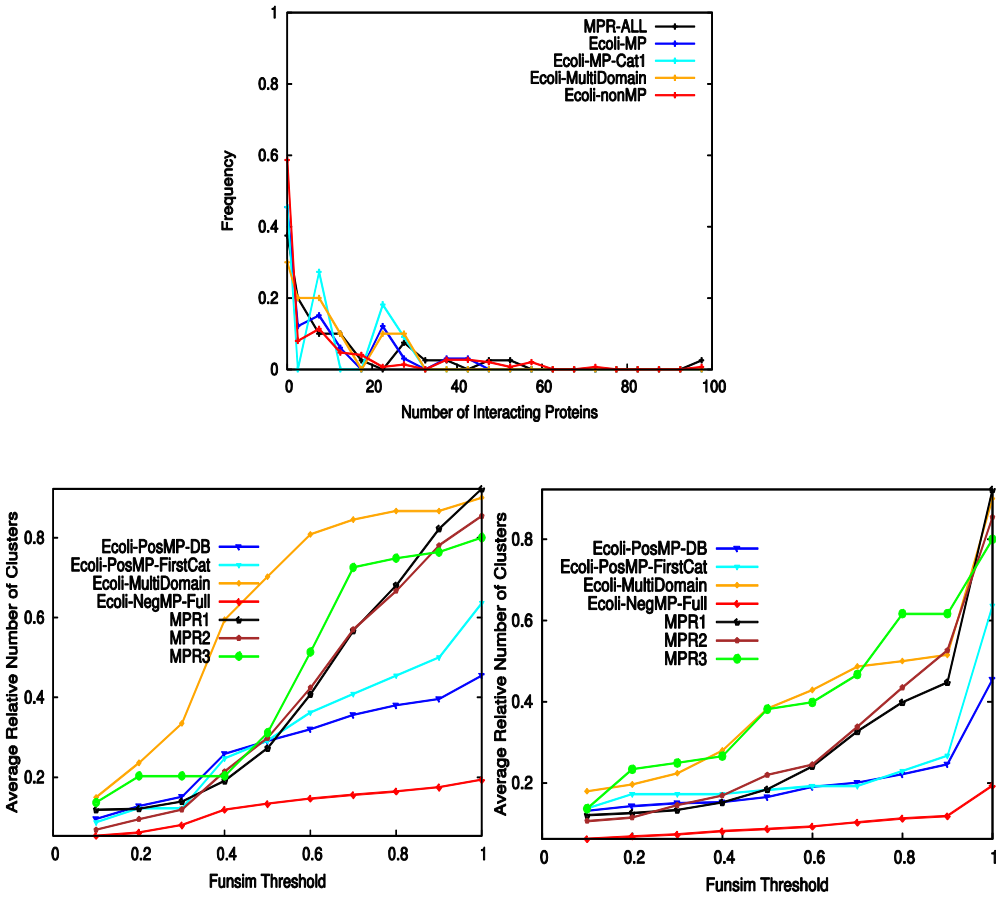

Figure 2.8 Interacting proteins of MP and non-MPs

Physically interacting proteins were obtained from the STRING database.

A, *Histogram of the number of interacting proteins.*

B, *average number of clusters of interacting proteins clustered using the funsim score (Eqn. 2.5).*

C, *Clustering was performed using the funsim score of BP terms only (Eqn. 2.5).*

Next, we checked the functional divergence of interacting proteins. Using the same datasets as Figure 2.8A, interacting proteins for each moonlighting or non-moonlighting proteins in the datasets are clustered based on their functional similarity using the funsim score (Eqn. 2.5). In Figure 2.8B, the average numbers of clusters per interacting protein at different threshold values are plotted. The funsim score of all three GO categories was used for Figure 2.8B while the funsim score with only BP (BP-funsim score) was used for Figure 2.8C. In the two clustering profiles (Figs. 2.8B & 2.8C) the non-MP set has consistently lower number of clusters as compared to moonlighting proteins. *E. coli* MPs and non-MPs show a clear contrast in the number of clusters with the former having over twice as many clusters as the latter. Consistent results were obtained when interacting proteins were selected from the STRING database using a score that combines different types of evidence including physical interactions, comparative genomics approaches, and gene expression (data not shown). A pairwise two-sample KS divergence test showed that the average number of clusters of the *E. coli* MP and nonMP sets is significantly different at the funsim-BP threshold values of 0.2, 0.6, and 0.8 and funsim threshold values 0.6 and 1.0 (Table A.4). To conclude, the results show that moonlighting proteins interact with proteins with more diverse functions than non-moonlighting ones.

We also investigated the extent to which the primary and secondary functions of a moonlighting protein are shared by its interacting proteins. For this analysis, we used 27 moonlighting proteins in the MPR1-3 sets that have interacting proteins because GO terms for their primary and secondary functions were manually classified. For each moonlighting protein in MPR1-3, we computed the functional similarity of its primary

function (F1) and its secondary function (F2) separately against GO term annotation of its interacting proteins. Functional similarity was quantified by the funsim score (Fig. 2.9A) and the BP-funsim score (Fig. 2.9B). To determine if an interacting protein was biased to either the F1 or F2 function, the score difference between F1 and F2 was computed.

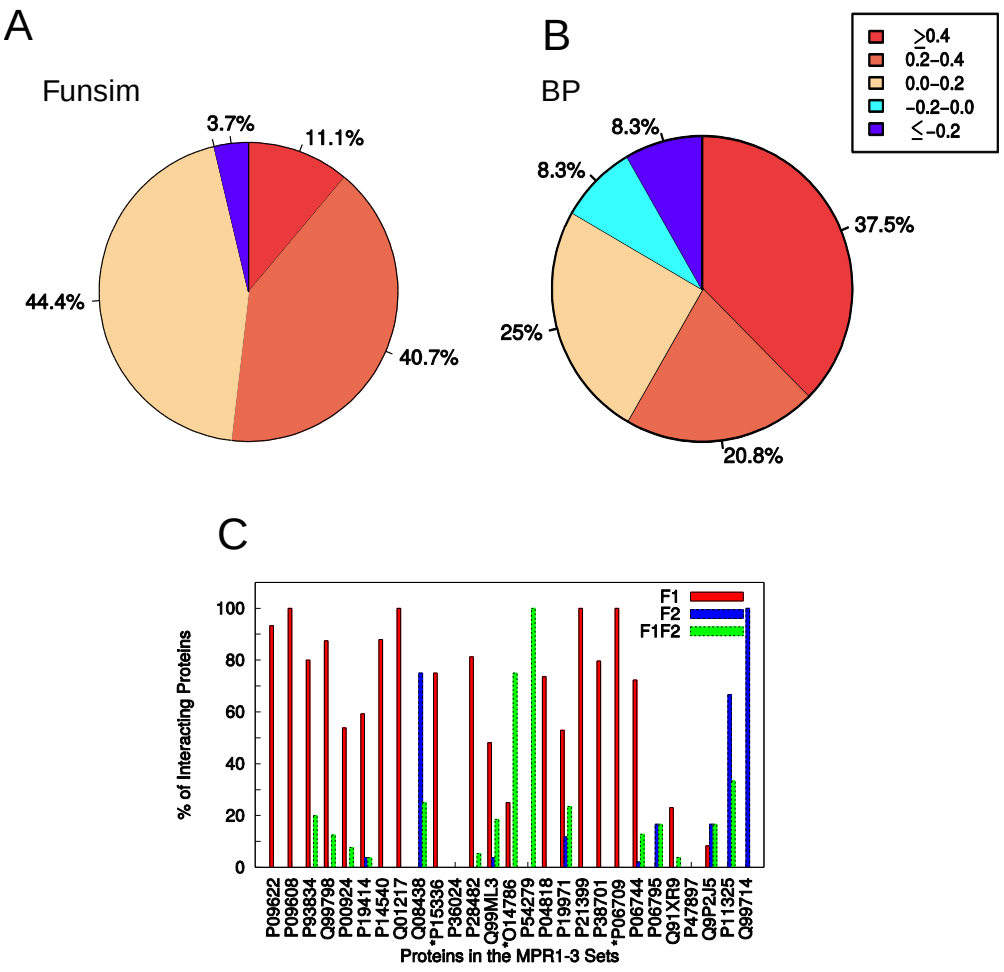

Figure 2.9 Function similarity analysis of MP's interacting partners

A, *The functional similarity score is computed between GO terms of the primary (F1) or the secondary (F2) functions of a moonlighting protein against the entire GO terms of its interacting protein and the score difference was computed.*

B, *The same type of chart as panel A, using the BP-funsim score.*

C, *Foreach moonlighting protein, percentages (%) of interacting proteins sharing F1, F2, or both functions of moonlighting proteins are shown.*

It is evident that the F1 function is dominant for the majority of the interacting proteins. When the funsim score was considered (Fig. 2.9A), 96.3% of the interacting proteins have functions closer to the F1 rather than the F2 function. The dominance of F1-oriented functions in interacting proteins is consistent in Figure 2.9B, where the BP-funsim score was considered.

Figure 2.9C provides results for individual moonlighting proteins. For a moonlighting protein, GO terms of its F1 and F2 functions were compared separately to the entire GO annotation of each interacting protein. If GO terms of an interacting protein have a BP-funsim score that is larger than the mean SS^{Rel} scores of BP terms in F1 or F2 of the moonlighting protein, the interacting protein was considered to share common F1 or F2 function, respectively, with the moonlighting protein. In the case that a moonlighting protein has very diverse F1 or F2 GO terms in itself with the mean SS^{Rel} score of 0, we used a BP-funsim score of 0.4 as a cutoff to determine if an interacting protein shares F1 or F2 function. Consistent with Figure 2.9A and 2.9B, the majority of interacting proteins have F1 function for 18 out of 27 the moonlighting proteins (66.7%) (red bars). On the other hand, only nine moonlighting proteins (33.3%) have interacting proteins of F2 functions (blue bars), and among them interacting proteins with F2 function are dominant for three (11.1%) moonlighting proteins.

There are interacting proteins of moonlighting proteins that have functional similarity with both F1 and F2 functions of moonlighting proteins (shown by green bars in Fig. 2.9C). Fifteen moonlighting proteins have in total of 30 interacting proteins with both F1 and F2 functions. We analyzed assigned GO terms of these interacting proteins

by referring to literature and found that 18 out of 30 of these proteins are also moonlighting proteins while three proteins are multi-domain proteins. This result indicates that moonlighting proteins tend to interact with moonlighting proteins; thus, novel moonlighting proteins may be identified by analyzing PPIs of moonlighting proteins.

We discuss two such cases. The first example is mismatch repair endonuclease PMS2 (P54279) in mouse, which also contributes to somatic hypermutation [97]. It has just one interacting protein, which is another DNA mismatch repair protein Mlh1 (Q9JK91) that is also involved in somatic hypermutation [98]. Thus, this is an example of two interacting moonlighting proteins that have the same primary and secondary functions.

The second example is mitogen activated protein kinase 1 (ERK2) (P28482) in human. This protein is MAP kinase and moonlights as a transcriptional repressor [99]. It has 187 interacting proteins in the PPI network, among which there are ten proteins with both F1 and F2 functions. One of the interacting partners is death-associated protein kinase 3 (DAPK3, UniProt: O43293), which enhances transcriptional activities of STAT3/P40763 by phosphorylating them. Besides the kinase function, DARPK3 is known to have multiple secondary functions, including involvement in apoptosis [88], roles in transcription (same as the secondary function of ERK2), regulation of cell polarity, contractile processes in non-muscle or smooth muscle cells, and cytokinesis [89]. Thus, in this example, among interacting moonlighting proteins that share both F1 and F2 functions one of them has more secondary functions.

2.4.2.4 Co-expressed protein network of MPs

Next, we investigated functions of co-expressed genes with moonlighting proteins in *E. coli*. The *E. coli* gene expression data were taken from the COLOMBOS database [100], which contains expression data of 4295 genes in 2369 contrasts. We calculated the Pearson correlation coefficient of expression levels of each pair of genes and selected pairs as co-expressed if the absolute value of the correlation coefficient ranked within the top 2% largest values among all the pairs. The number of co-expressed genes of moonlighting and non-moonlighting proteins do not have large difference, except for a peak observed at 65 for the moonlighting proteins (Fig. 2.10A), which consists of four moonlighting proteins (P77489, P0A8Q3, P0AC47, and P25516). Then, similar to the analysis in Figure 2.8B and 2.8C, we computed functional clustering profile for co-expressed genes of *E. coli* moonlighting proteins to see if co-expressed genes have functional divergence. The clustering profile using the funsim score (Fig. 2.10B) and the BP-funsim score (Fig. 2.10C) showed that the moonlighting proteins have a slightly larger average number of clusters of functionally similar proteins per co-expressed genes than that for non-moonlighting proteins, although this difference is not statistically significant (Table A.4). The same conclusion was obtained when we defined co-expressed genes as those which have over 0.4 of the correlation coefficient value (data not shown).

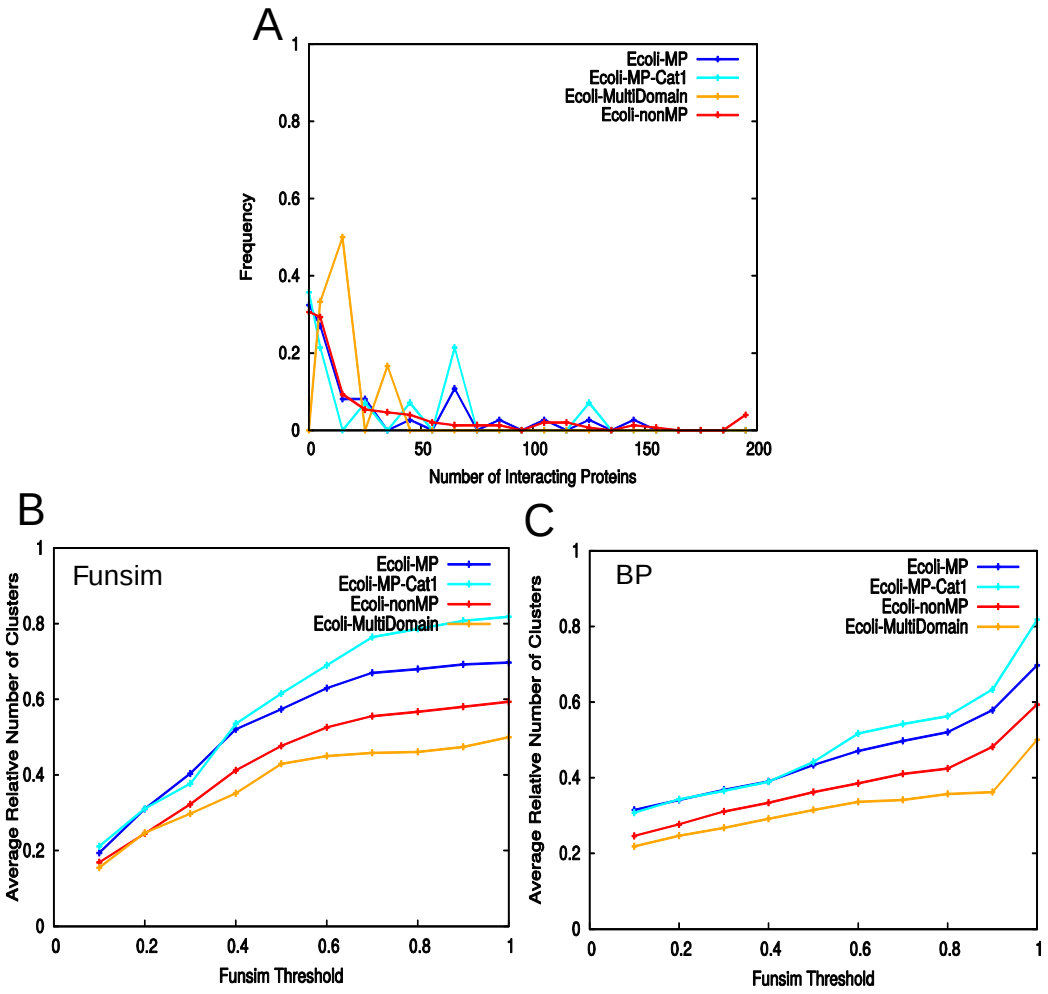

Figure 2.10 Gene expression profile analysis for MPs

Average number of clusters of interacting proteins relative to the number of proteins interacting by gene expression. Proteins considered to be interacting are the top 2% of proteins in the Gene Expression network of E. coli sorted in terms of the Pearson correlation coefficient.

A. *Histogram of number of interacting proteins.*

B, *Functional clustering using Funsim (BP, MF, CC) score thresholds between 0.1 and 1.0.*

C, *Functional clustering using Funsim (BP) score thresholds between 0.1 and 1.0.*

2.4.2.5 Phylogenetic co-evolution network of MPs

We further analyzed genes that have similar comparative genomic context to the moonlighting proteins [90]. Using the STRING database, for a protein of interest, we selected proteins as phylogenetically related if they were located in the neighbourhood of the target genes, were found to co-occur or co-absent, or were fused in multiple genomes. Concretely, genes that have a sufficient score (> 0.7 as recommended by STRING) at "neighborhood", "co-occurrence", or "gene-fusion" in the STRING database [96] were selected. It has been observed that phylogenetically proteins are functionally related in many cases [90]. Figure 2.11 shows the clustering profiles of phylogenetically related proteins of the moonlighting and non-moonlighting proteins.

A larger fraction of the non-moonlighting proteins have no phylogenetically related proteins as compared with the moonlighting ones (0 at the x-axis in Fig. 2.11A). The clustering profiles using the funsim score (Fig. 2.11B) and the BP-funsim score (Fig. 2.11C) show that the *E. coli* moonlighting proteins have slightly more functional clusters on average, i.e. more functional divergence in their phylogenetically related proteins, than their non-moonlighting counterparts. The p-value of this difference in the number of functional clusters was 0.08 at the score threshold of 0.8 in the funsim score (Fig. 2.11B) and larger than 0.05 for the BP-funsim score profile (Fig. 2.11C). Comparing with the MPR1-3 sets, on average MPR2 and MPR3 have a higher number of clusters than the *E. coli* moonlighting and proteins, while the MPR1 set has less functional divergence in their phylogenetically related proteins.

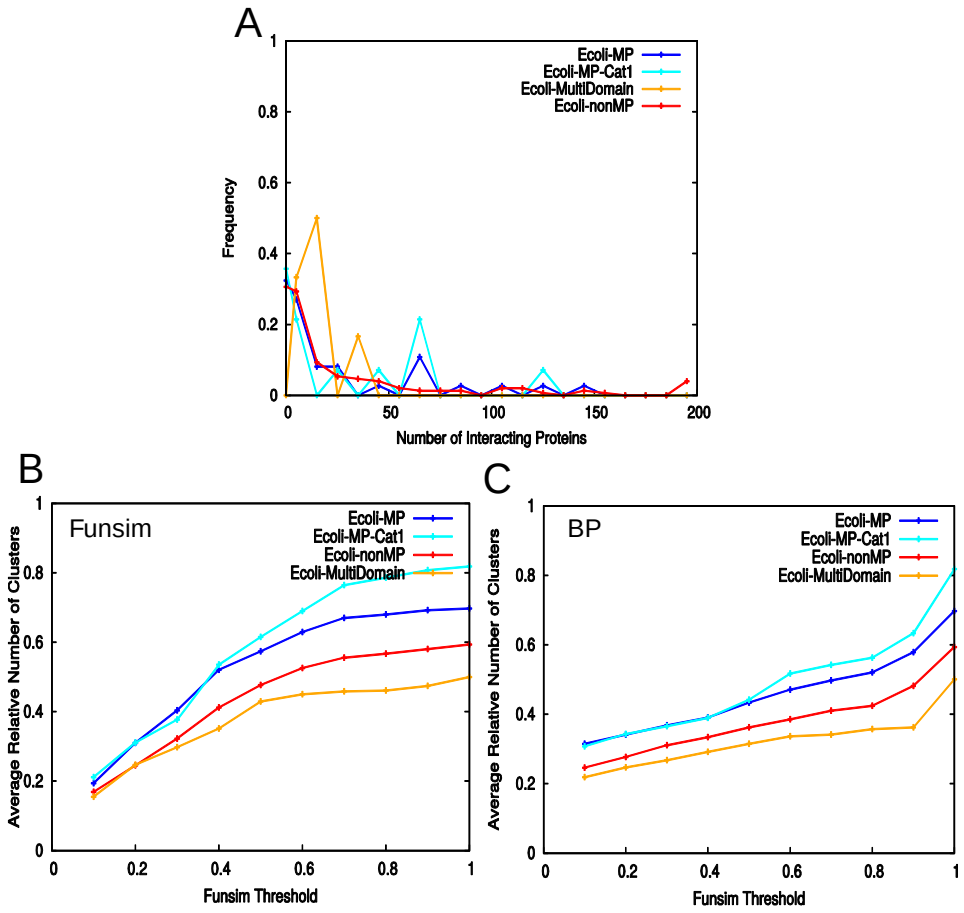

Figure 2.11 Phylogenetic profile analysis for MPs

Average number of clusters of phylogenetically related proteins relative to the number of phylogenetically related proteins. Phylogenetic related proteins are taken from the STRING database.

A, *The histogram of number of phylogenetically related proteins.*

B, *Functional clustering using Funsim (BP, MF, CC) score with thresholds between 0.1 and 1.0.*

C, *Functional clustering using Funsim (BP) score thresholds from 0.1 to 1.0.*

2.4.2.6 Genetic interaction network of MPs

The last omics data we analyzed were genetic interactions. A genetically interacting gene pair was identified by examining the growth curves of a single gene knockout mutant and a double gene knockout mutant. In general, genes in the same pathway tend

to show positive interaction and those in parallel pathways show negative or synthetic lethality [101]. Genetic interactions in *E. coli* were identified by Takeuchi et al. [102] using conjugation methods reported as GIANT-coli [103] and eSGA [104] with an improved quantitative measurement [105]. This dataset includes genetic interaction data for 215 genes against 3868 genes, which results in total of 813,560 gene combinations. Among them, 2009 pairs were identified as genetically interacting, which were defined as those have a correlation coefficient of over 0.2 in the maximum growth rate in time-series measurements [102]. The interacting gene pairs overlap with a small portion of the *E. coli* moonlighting and non-moonlighting proteins: 5 out of 33 moonlighting proteins, 3 out of 16 first category moonlighting proteins, and 5 out of 150 non-moonlighting proteins. Using these shared proteins, we performed the clustering profile analysis (Fig. 2.12).

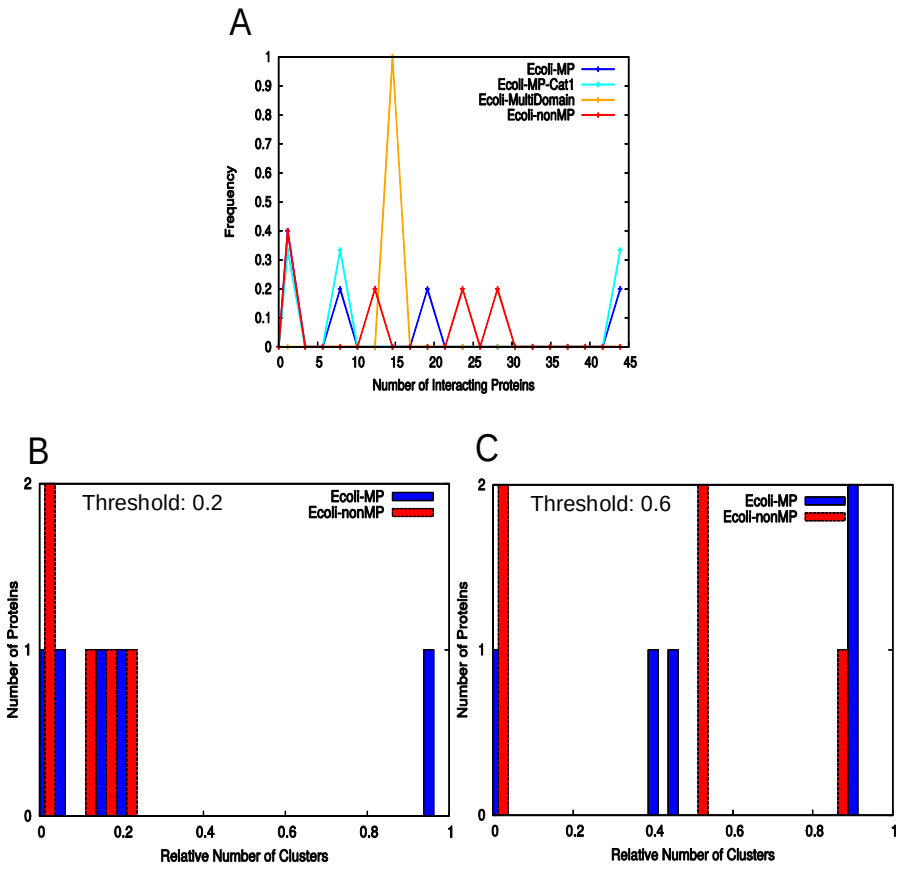

Figure 2.12 Genetic interaction network analysis for MPs

The number of interacting proteins in the genetic interaction network of E. coli.

A, *The number of interacting proteins selected with a Pearson correlation cutoff of 0.2. E. coli MP and non-MP, multi-domain multi-functional proteins, and the first category E. coli MPs are plotted.*

B, *The number of clusters of interacting proteins for individual E. coli moonlighting (blue) and non-moonlighting (red) proteins at BP-funsim threshold of 0.2.*

C, *The number of clusters of interacting proteins for individual E. coli moonlighting (blue) and non-moonlighting (red) proteins at BP-funsim threshold of 0.6.*

Moonlighting and non-moonlighting proteins do not seem to have difference in the number of genetic interactions (Fig. 2.12A) and the number of functional clusters (Fig. 2.12B & 2.12C), although the number of proteins available for the analysis was too small to make a firm conclusion. In terms of the number of genetic interactions (Fig. 2.12A), there is one moonlighting protein that has 43 genetic interactions. This protein is a subu-

nit of fumarate reductose flavoprotein in *E. coli* (P00363), which we classified as a first category moonlighting protein (Table A.1). The 43 interacting proteins belong to 30 different pathways. Panels B & C in Figure 2.12 show histograms of the number of functional clusters of genetically interacting proteins for the *E. coli* moonlighting and non-moonlighting proteins at the BP-funsim thresholds of 0.2 and 0.6. There is a moonlighting protein that interacts with two proteins with very different functions (the bar at x=1.0 in Fig. 2.12B). This protein is P23895, a third category/weak moonlighting protein identified to function as a multidrug transporter and in DNA damage response. It interacts with P77368 (UPF0098 family protein inferred by homology) and P75719 (endopeptidase that performs host cell lysis).

To summarize the omics data analyses, we observed a clear tendency for moonlighting proteins to have physical interactions with more diverse classes of proteins and most of these proteins share the primary function of the moonlighting protein with which they interact. Moreover, it was found that moonlighting proteins frequently physically interact with other moonlighting proteins. In terms of gene expression and phylogenetically related proteins, a weak trend was observed that on average moonlighting proteins interact with more functionally diverse proteins, although not all of the cases were statistically significant.

2.4.2.7 Structural properties of MPs

Now we turn our attention to structural properties of moonlighting proteins, namely intrinsically disordered regions and ligand binding sites. An intrinsically disor-

dered region in a protein lacks a well-defined tertiary structure in its native condition. Intrinsically disordered regions have been found to have important roles in protein function [106], often serving as binding sites for proteins. There are moonlighting proteins that can both activate and inhibit their binding partners in the same or overlapping binding regions which have been found to be disordered. These proteins can bind the same partner in different conformations or bind to completely different partners through the disordered binding regions [107]. Here, we examined the prevalence of disordered regions in the proteins in MPR1-3 and the *E. coli* moonlighting and non-moonlighting proteins. Disordered regions in the proteins were obtained from the D2P2 database [108].

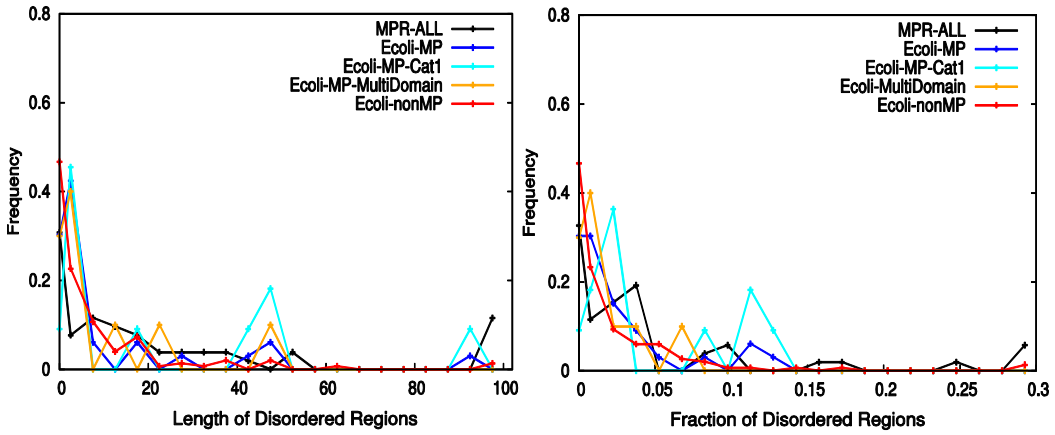

Figure 2.13 Disordered region of MP & non-MPs

Histograms of the disordered regions in moonlighting and non-moonlighting proteins. Five datasets are plotted: MPR1-3 (MPR-All), E. coli moonlighting proteins (Ecoli-MP), E. coli moonlighting proteins in the first category (Ecoli-MP-Cat1), multi-domain multi-functional proteins, and E. coli non-moonlighting proteins (Ecoli-nonMP).

A, *Length of the disordered regions;*

B, *Fraction of the length of disordered regions relative to the whole sequence length of the proteins.*

The total length of disordered regions and their fraction relative to the full length of a protein are shown in Figure 2.13. The distributions for moonlighting proteins and non-moonlighting proteins were overall similar, both having the peak at lower end within disordered region lengths 0 to 5. However, it is noteworthy that moonlighting proteins had a smaller fraction of proteins with no disordered regions (Fig. 2.13A) and more moonlighting proteins had a larger fraction of disordered regions (Fig. 2.13B). Moonlighting proteins had a small peak for disordered regions of 47 residues in length and slightly higher frequency for disordered regions of over 90 residues (Fig. 2.13A). The peak of the moonlighting proteins at 47 residue-long disordered regions (Fig. 2.13A) consists of four proteins, fumarate reductase (P00363), ribonuclease R (P21499) deferrochelatase (P31545), and GTPase ObgE (P42641). Moonlighting proteins with a large fraction of disordered region include anion exchange protein 3 (P48751) and phosphopantothenoylcysteine decarboxylase subunit VHS3 (Q08438) and subunit S1S2 (P36024). Anion exchange protein 3 does not have known physical interactions with other proteins while the two subunits of phosphopantothenoylcysteine decarboxylase have eight physical interactions in the PPI network.

Finally, we discuss ligand binding sites in the tertiary structures of moonlighting proteins that are related to either of their primary or secondary functions. Such examples are limited since the tertiary structures of the proteins must be available for the analysis and multiple bound ligands need to be involved in the functions. Sixteen proteins in the MPR1-3 sets have their tertiary structures available in PDB [109,110]. Among them, we found six structures that have two ligands that bind to physically different locations. We

discuss two cases below, because the other four are multi-domain proteins (Fig. 2.14). These two proteins to be discussed are one-domain proteins according to Pfam.

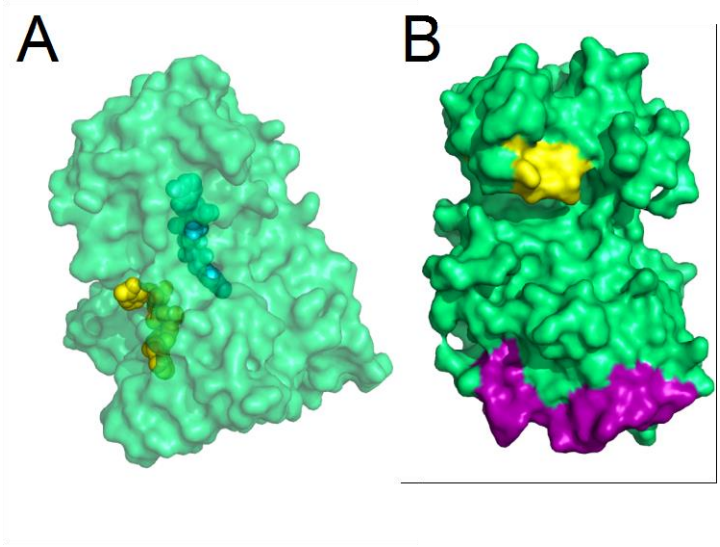

Figure 2.14 Moonlighting protein structures

A, *human dihydrolipoamide dehydrogenase (PDB ID: 1ZMC-A). It binds NAD shown in yellow at residues 208, 243, 279 ("NAD binding" classified as both F1 and F2 function) and FAD shown in cyan at residues 54, 119, 320 ("FAD binding" classified as F2 term).* ***B,*** *mitogen activated protein kinase 1 (PDB ID: 4G6N). It binds ATP (related to F1 function) at residues 31-39 and 54 (shown in yellow), and DNA (related to F2 function) with residues 259-277 (purple).*

The first example is dihydrolipoamide dehydrogenease (DLD) in human (P09622) (Fig. 2.14A). The primary function of this protein is as a mitochondrial enzyme in energy metabolism and its secondary function is protease. To perform the primary function, it utilizes dihydrolipoic acid and NAD+ to generate lipoic acid. Experiments suggest that mutations that destabilize a DLD homodimer can simultaneously induce the loss of a primary metabolic activity and the gain of a moonlighting proteolytic activity [111]. It was also pointed out that the moonlighting proteolytic activity of DLD could arise under pathological conditions, including the presence of dimer-destabilizing mutations or the

acidification of the mitochondrial matrix. The latter condition disrupts the quaternary structure of DLD, leading to a decrease in the dehydrogenase activity and increase in the diaphorase activity, which is a FAD and NAD dependent activity. Based on these information we classified “NAD (nicotinamide adenine dinucleotide) or NADH binding” (GO:0051287) for both functions and term “FAD (flavin adenine dicucleotide) or FADH2 binding” (GO:0050660) to the secondary function. A crystal structure of DLD (PDB ID: 1ZMC-A) shows that the NAD and FAD binding sites are located in physically separate regions in the protein surface.

The second example is MAP kinase (ERK2) in human. The secondary function of this protein was identified as a DNA binding transcriptional repressor that regulates interferon gamma signalling [112]. Naturally, binding ATP is related to the primary function as a kinase (GO:0005524) while “DNA binding” (GO:0003677) belongs to the secondary function. As shown in Figure 2.14B, the binding sites for ATP and DNA are located quite far apart in the protein structure.

To summarize the structural analyses, about 48% of moonlighting proteins have disordered regions longer than five residues and this percentage is larger than that of non-moonlighting ones (29%). Also examples are observed in which moonlighting proteins have relatively longer disordered regions. In terms of the tertiary structures, examples are found where ligand (including DNA) binding sites that are related to either the primary or secondary functions are located in distinct regions on the protein surface. These structure features may be useful for predicting the existence of secondary function of proteins when combined with other evidences.

2.5 Computational prediction of MPs - MPFit

The functional diversity of moonlighting proteins poses a significant challenge to computational protein function annotation as current methods do not explicitly consider the possibility of dual functions for a protein. Conventional sequence-based functional annotation methods, based on the concept of homology [6] or conserved motifs/domains [72-74], will have problems identifying secondary functions because there are cases where a homolog of a moonlighting protein does not possess the secondary function [75] or has a different secondary function [67,76]. Due to these intrinsic computational challenges, systematic studies of moonlighting proteins are still in an early stage for obtaining a comprehensive picture of proteins' moonlighting functions or for developing computational methods for predicting moonlighting proteins [review by [47]]. Existing bioinformatics approaches for detection of moonlighting proteins have two general shortcomings. First, they rely heavily on the existence of functional annotation of a protein (Chapple CE et al., 2015; Pritykin Y et al., 2015), which is a major bottleneck of the problem. Second, all the existing methods address different aspects of moonlighting proteins' functional diversity: sequence similarity [49,78], motifs/domains, structural disorder [81], or protein-protein interaction (PPI) patterns combined with existing gene ontology annotations [79,113,114]. However, the diverse nature of moonlighting proteins' functions, cellular locations, function switching mechanisms, and the organisms in which they are found gives compelling evidence that in order to understand and identify the overall functional aspects of these proteins, one should characterize these proteins in a wider functional/proteomic space.

Previously, we have identified functional characteristics of moonlighting proteins in different proteomic aspects using a computational framework [48]. Here, we have constructed an automated prediction model to identify moonlighting proteins based on features we characterized in our previous study. To address the diverse nature of moonlighting proteins, we have used a wide feature space ranging from gene ontology (GO) and several omics-scale data, namely protein-protein interaction (PPI), gene expression, phylogenetic profiles, genetic interactions, and network-based graph properties (such as node between-ness, degree centrality, closeness-centrality), to protein structural properties such as the number and the length of intrinsically disordered regions in the protein chain. Based on our computed GO and the omics-based protein feature space, we used machine learning classifiers as the framework for moonlighting protein prediction and used an existing moonlighting protein database to cross-validate our prediction model. Since a significant fraction of proteins do not have certain functional/network features in databases, we have additionally developed an imputation technique using random forest to predict missing features for proteins. Cross-validation results on the dataset of known moonlighting and non-moonlighting proteins (control dataset) show that if GO information is available, moonlighting proteins can be predicted with over 98% accuracy. More importantly, leveraging just the non-GO based features, our imputation-classification models can predict moonlighting proteins with over 75% accuracy. The latter result is very important because it indicates that moonlighting proteins without sufficient function annotations can be identified by analyzing available omics data, which is the first such development. Lastly, we have run our imputation-classification models with the best performing omics-based feature combinations on three genomes, *Saccharomyces cerevisiae* (yeast), *Caeno-*

rhabditis elegans, and *Homo sapiens* (human), and found that about 2-10% of the proteomes are potential moonlighting proteins.

2.5.1 Methods

The overall computational prediction model, named MPFit (Moonlighting protein Prediction with missing Feature Imputation) undergoes four phases: data construction, feature computation, missing feature imputation (when needed) and classification into moonlighting protein (MP) or non-moonlighting protein (non-MP). Each of the steps is discussed in detail below.

2.5.1.1 Data construction for MPFit

We used a manually curated moonlighting protein database, MoonProt [84], and extracted 268 proteins that had Uniprot ID mapping. 268 moonlighting proteins (MPs) include those from human (45 proteins, 16.8%), E. coli (30 proteins, 11.19%), yeast (27 proteins, 10.1%), and mouse (11 proteins, 4.1%). In order for our model to train on negative examples of such proteins along with the positive examples, we used the following criteria to select negative examples of MPs (referred as non-moonlighting proteins, non-MPs) from these four genomes as developed in our previous work [48]. A protein was selected as a non-MP if it has a) at least 8 GO term annotations, b) when GO terms in the Biological Process (BP) category were clustered using the semantic similarity score [93] thresholds of 0.1 and 0.5, not more than one cluster was obtained at each threshold. We further added a criterion on Molecular Function (MF) category GO terms: c) not more

than one cluster of MF GO terms at semantic similarity scores of 0.1 and 0.5. In essence, a non-MP is a protein that has a sufficient number of GO annotations but they are not functionally diverse. For this procedure, full GO annotations (including computationally predicted terms such as IEA) were taken from UniProt (ver. Dec 2014) and parental propagation of GO terms was not applied, to be consistent with the criteria established in our previous work [48]. Furthermore, we computed pairwise sequence similarity of the selected non-moonlighting proteins from the above three conditions and further ruled out redundant proteins that had more than 25% sequence identity to other sequences. This process yielded 162 non-MPs, among which 60 are from human (37.0%), 52 from mouse (32.1%), 34 from yeast (20.9%), and 16 from E. coli (9.88%). The MP and non-MP datasets are made available at http://kiharalab.org/MPprediction/.

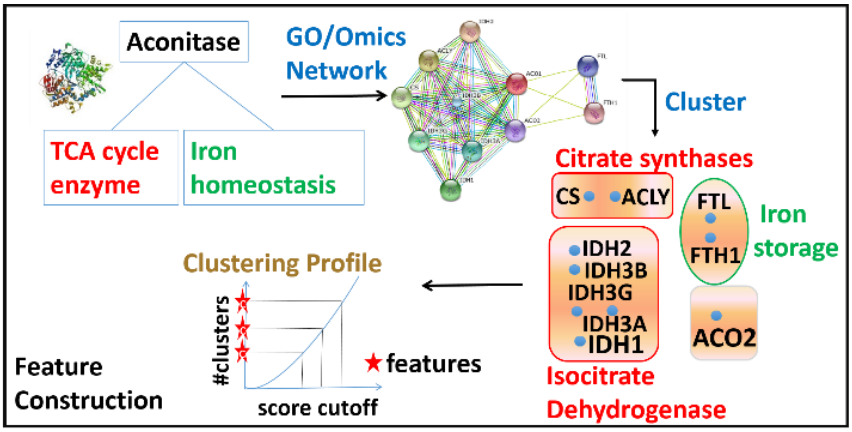

Figure 2.15 Schematic diagram of MPFit

Feature construction of moonlighting protein Aconitase in PPI network.

2.5.1.2 Feature computation and selection

As MPs have dual functions, intuitively they interact with more proteins with different functions compared to non-MPs. This stems from the fact that proteins perform their functions through different forms of associations with other proteins. In our previous work [48], we have characterized MPs and non-MPs in terms of different omics-based features (including PPI, gene expression, phylogenetic profile, genetic interactions) and showed that when the interacting partners are clustered based on their functional similarity, the number of lusters tend to be higher for MPs than non-MPs. Based on this analysis, we develop the MPFit model in this work that uses the number of functional clusters as the features to classify MPs and non_MPs.

We computed features for the dataset of MP and non-MPs to run machine learning classifiers. We selected features from a broad range of information domains, i.e., GO annotations, PPI network, gene expression profiles (GE), phylogenetic profiles (Phylo), genetic interactions (GI), disordered protein regions (DOR), and the protein's graph properties in the PPI network (NET). In order to extract the feature for a protein P_i in any information domain, we first extracted the GO terms or proteins associated with P_i in that domain and built a network N_i for P_i. Each node in N_i can be either a GO term (if the information domain is GO) or a protein (if the information domain is any of the omics-based information); edges in N_i represent association weights among nodes. Then we applied single linkage clustering on N_i and the number of clusters at several score thresholds were selected as features of P_i [48]. Fig. 2.15 illustrates the feature computation procedure for aconitase in human (aco1), an MP, for the PPI network. First, we extracted interacting partners for aco1, then based on the GO annotation similarity score of the in-

teracting partners, the PPI network was clustered and four clusters were obtained with a certain similarity cutoff *i*. Two of these clusters (circled in red) contain proteins related to the TCA cycle and are associated to the first function of aco1 while another cluster (green) was relevant to the second function. Such clustering was performed with five different similarity cutoffs (from 0.1 to 0.9 with an interval of 0.2), which resulted in a clustering profile shown in the bottom of Fig. 2.15. Finally, we extracted the number of clusters at multiple score cutoffs as the PPI network features of aco1. More details about the feature computation in PPI network domain is provided in the Supplementary Fig. A.1.

To construct the gene expression (GE) network, expression profiles were obtained from the COEXPRESdb [115] database. Gene pairs that have an absolute value of their Pearson correlation of expression levels within the top 2% among all the pairs were connected in the network. Phylogenetic profile (Phylo) network was constructed using the STRING [96] database. A protein pair was connected in the network if they have a sufficient score (> 0.7 as recommended by STRING) at "neighborhood", "co-occurrence", or "gene-fusion" in the STRING database. For the genetic interaction (GI) network, we used the BIOGRID database [116] and extracted gene pairs that had the "experiment type" listed as "genetic" to be associated in the GI network. For the NET feature, three graph properties of proteins, namely, degree centrality, closeness centrality, and between-ness centrality, based on the PPI network (STRING database [96]) were computed as features. For the DOR feature, using the D2P2 database [108], we computed three properties of protein's intrinsically disordered regions, namely, the number and the total length of disordered regions as well as the proportion of disordered regions in the sequence.

2.5.1.3 Missing data imputation

In order to deal with missing data, imputation is the approach that fills in the missing features rather than discarding the data points entirely and working with only the complete subset of the data. Among known imputation approaches, there are set of methods that fill in the missing feature from mean or median of the known values of the same features in other instances [117,118]. On the other hand, there are methods that do partial imputation by imputing the missing data based on known features of small neighborhood of the incomplete data [119,120]. In this work, we used a random forest-based imputation technique that predicts missing features [121,122]. Fig. 2.16A-B shows the procedure. In Fig. 2.16A, the training dataset is represented as a matrix where rows are proteins and columns are features. Missing features in the dataset are represented by NAs. The algorithm starts by replacing NAs with the column medians. Then a random forest was constructed using the feature set that are temporally filled by the previous step (pseudo-complete data in the figure). Next, the proximity matrix from the random forest was used to update the imputed values of the NAs. The (*i*, *j*) element of the proximity matrix is the fraction of the trees in which the proteins *i* and *j* fall in the same class. The imputed value for a feature is the weighted average of the non-missing features from other proteins, where weights are the proximities. The imputation was iterated until the proximity matrixes converged or the procedure is iterated ten times, when the missing features were determined. Finally, a random forest RF_{train} was computed with this imputed training data matrix.

In order to impute missing features in the test set (Fig. 2.16B), the training dataset with missing values imputed was used to compute two filler vectors (referred to as MP-filler and non-MP-filler), one for each of the MP and non-MP classes. The i^{th} element of the filler vector MP-filler (non-MP-filler) is the mean of the imputed features at the i^{th} column of the training matrix with the MP (non-MP) class label. The test dataset was represented as a matrix similar to the training data (rows are proteins, columns are features). For the test data row r_i^{test}, since the label (MP/non-MP) is not known, two replicates were made: the missing features in the first replicate were filled using the vector MP-filler and the same for the second replicate was filled using the non-MP-filler vector. Now these two completed test replicates were run down through the previously trained random forest RF_{train}. Each protein receives tree votes of MP and non-MP in RF_{train} from replicates 1 and 2, and the higher vote between the MP vote in replicate 1 and the non-MP vote in replicate 2 finally determines the MP/non-MP-fillers to be used in the missing features of the protein. In Fig. 2.16B, the first protein received higher MP votes from replicate 1 (290 votes) over non-MP votes from replicate 2 (50 votes); thus, the missing features of the protein are filled with the MP-filler vector. Finally, proteins in the test set were predicted to be MP or non-MP using a classifier. When RF was used for the classifier, this

voting was used as the final prediction. We have also used support vector machine (SVM) and naïve-Baiyes (NB) as the final classifier and compared all results.

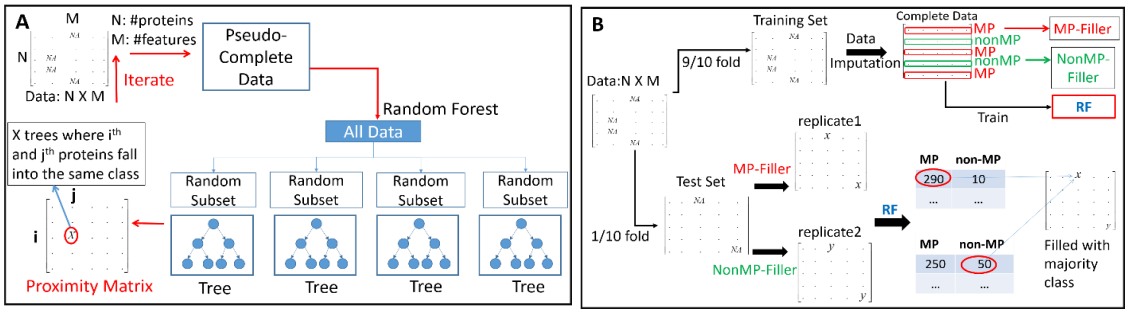

Figure 2.16 Schematic of missing feature imputation by MPFit

A-B: *Missing feature imputation method. RF: Random Forest. See text for details.*

Aside from this explicit random-forest based imputation technique, an alternative imputation method (termed as "probabilistic imputation") was used in this work where the splitting probabilities in the random forest were learned from the subset of complete data and later used to classify the incomplete data. Detail of this method is discussed in Supplementary Fig. A.4 and its associated text.

2.5.2 Results

In this section we present and discuss the performance of MPFit with different combinations of features. MPFit was run and evaluated with the GO term feature and all possible combinations of six omics feature domains (namely, PPI, GE, Phylo, GI, DOR, and NET). There are 1+ (26 – 1) = 64 such combinations.

2.5.2.1 Imputation of missing features facilitates usage of omics data

For a given combination of omics features, there are proteins which lack some of the feature data. One way to handle such missing data by a classifier is to impute the missing data so that a classifier trained on the full features can be applied. Fig. 2.17 contrasts the number of target proteins that were predicted by MPFit before and after the imputation. A point represents one of the 64 feature combinations. For each feature combination considered, proteins that have at least one feature were subject to imputation and those that do not have any features are discarded (data points in Fig. 2.17 with under 100% protein coverage after imputation).

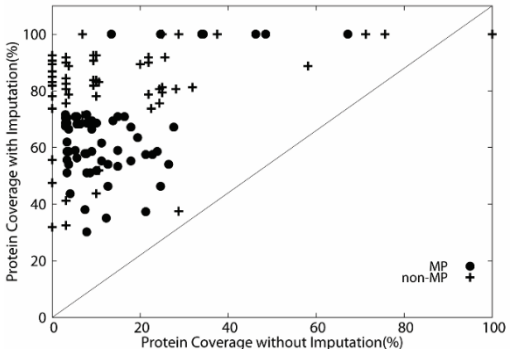

Figure 2.17 Impact of missing feature imputation

It is evident that the imputation technique dramatically increased the dataset coverage, which also consequently improved classifier performance as explained in later sections. For example, the number of MP proteins for a feature combination of (PPI, Phylo, GE, GI, DOR) was originally 8 (2.9%), which increased to 192 (71.7%) after imputation. The features with 100% coverage after imputation are seven single features, GO, GE, Phylo, PPI, GI, NET, and DOR.

2.5.2.2 Prediction accuracy of MPs

Next, we discuss prediction performance of MPFit using random forest (RF) [122] as the final classifier in the pipeline (Fig. 2.16B). The 64 different feature combinations were used including the seven cases that only use one feature. Accuracy of predictions was evaluated by a weighted class average F-score, where the F-score was computed separately for MP and non-MP protein classes and weighted by the number of proteins in the corresponding class. The F-score is defined as (2*precision*recall)/(precision+recall), where precision and recall are defined as (TP/(TP+FP)) and (TP/(FP+FN)), respectively. Here, TP, FP, and FN stand for true positive, false positive, and false negative, respectively. Fig. 2.18 presents results with the seven single features as well as the five combinations of features that showed the highest F-score. Average F-score from a five-fold cross-validation was reported.

When proteins have GO annotations, it is shown that prediction can be very accurate, with an F-score of 0.993. Among the six individual omics features, GE showed the best F-score of 0.710, and the rest of the features performed similarly (F-scores range from 0.597 to 0.651). Results of all the possible combinations of omics features are provided in supplementary Fig. A.2. Their F-scores range from 0.784 to 0.571. Among the feature combinations, Phylo+GI showed highest accuracy (precision, recall and F-score are 0.799, 0.771, and 0.784, respectively), followed by Phylo+GI+NET and Phylo+NET. However, these three combinations have relatively low coverage (Fig. 2.18), while the fourth and fifth best performing feature combinations, Phylo+GE+GI+DOR+NET and PPI+Phylo+GE, have a high coverage with good F-scores that are close to the best value achieved by Phylo+GI (0.7964, 0.7602 for coverage and 0.7109, 0.7538, for F-score, re-

spectively). For this reason we used the fourth and fifth feature combinations in the genome-scale prediction performed in the subsequent section. Among the proteins in MoonProt, there are five protein pairs from the same organism that have over 25% sequence identity. We removed five proteins, one from each of these high-sequence-similarity pairs and recomputed the F-score with cross-validation for the two feature combinations, Phylo+GE+GI+DOR+NET and PPI+Phylo+GE. The changes of F-score were marginal: an increase of 0.87 and 3.09 were observed for the former and the latter combinations, respectively.

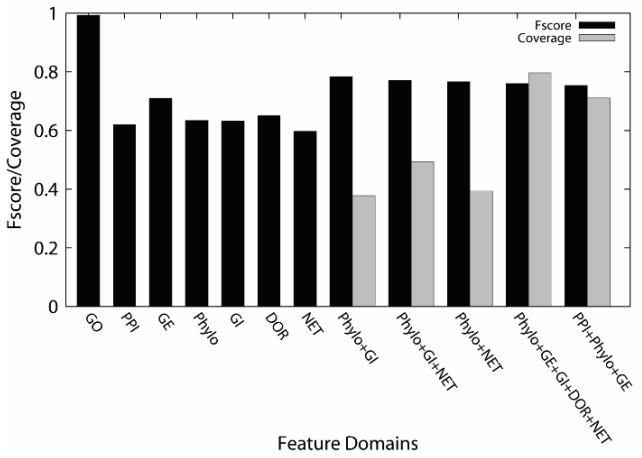

Figure 2.18 Performance of MPFit with random forest

Here we discuss two cases where combinations of different omics-based features improved prediction over single feature. The first example is a MP in human, which is a ribosomal protein (part of the 60S subunit) (UniProt ID: P46777) [123]. This protein also binds to and inhibits HDM2, an ubiquitin ligase, which results in stabilization of the p53 tumor suppressor protein. Using only the PPI features, this protein is incorrectly predicted as non-MP. This is because 63 interacting proteins in PPI network for this target protein

have relatively small number of functional clusters for MP. When clustered using functional similarity (funsim) scores for BP and MF (See Supplementary Fig A.1 for feature computation), the relative number of clusters stay below 0.32 at each clustering cutoff, which is significantly low compared to the MP distribution shown in Suppl. Fig. A.1B. However, the protein was correctly predicted as MP by the PPI+Phylo+GE combination. Phylo features were actual values while GE were imputed for this protein. 25 interacting proteins for this target in the phylogenetic profile network were clustered in to 2, 3, 3, 3, and 24 clusters at similarity cutoffs 0.1, 0.3, 0.5, 0.7, and 0.9 of the funsim score, which are larger than the non-MP distribution shown in Suppl. Fig. A.5A. Thus for this protein, addition of Phylo features to PPI made the prediction correct to MP.

The second example is DNA replication factor Cdt1 (UniProt ID: Q9H211) [124]. Besides its primary function as DNA replication factor, this MP's moonlighting function is a role in mitosis where it localizes to kinetochores through binding to the Hec1 component of the Ndc80 complex. Using PPI features only, this protein is incorrectly predicted as non-MP, because its 29 interacting proteins in the PPI network were clustered into relatively smaller number of functional groups. Clustering using funsim BP+MF score, the relative number of clusters stays below 0.35, which is significantly low compared to the MP distribution. However, the PPI+Phylo+NET feature combination correctly predicted the protein as MP. This is partly because the NET feature of this protein has high values, e.g. a between-ness centrality of 0.2668, which is high (above 75 percentile) compared to this feature's quantile distribution (Suppl. Fig. A.5B).

We also ran MPFit with random forest without imputation, i.e. only on proteins that do not have any missing feature in a feature combination. The results for all the fea-

ture combinations are shown in Supplementary Fig. A.3. Skipping imputation substantially lowers coverage (Fig. 2.18, and Figs. A.2, A.3). Without imputation the coverage decreases as the number of features in a combination increases, which resulted in 0 coverage for 16 out of 64 cases (Fig. A.3). Also, the data sizes of MP and non-MP classes become substantially different and imbalanced for several feature combinations (Fig. A.3). Note that the situation is opposite when the imputation procedure was applied, i.e. the coverage increases as the number of features to combine increases, because proteins that have at least one feature in a combination were subject to prediction by imputing other missing features. Imputation not only increases prediction coverage but also improves accuracy by increasing the size of the training set, as indicated by the cases that improved F-score by imputation.

We examined prediction performance of MPFit when naïve Bayes [125] or SVM [126], was used as the last classifier in the procedure. As explained with Fig. 2.16, the missing data imputation was performed with random forest, and naïve Bayes or SVM was applied as the final classifier to proteins with full imputed features. Results with all 64 feature combinations were shown in comparison with the results by random forest in Fig. 2.19.

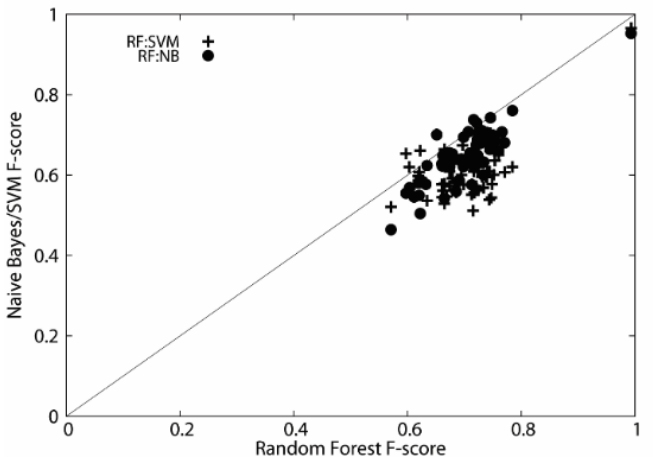

Figure 2.19 Performance comparison of random forest with two other classifiers

F-score using each of the different feature combinations by MPFit with random forest (RF) was compared with SVM (cross) or naïve Bayes (filled circles). The imputed dataset was used. Results are the weighted class average F-score over five-fold cross validation.

Results in the lower triangle in Fig. 2.19 are the cases where random forest performed better than the counterpart. It is apparent that random forest performed better than SVM and naïve Bayes for the majority of the cases. Using the GO term features showed the highest F-score by all the classifiers (the upper right corner of Fig. 2.19). Among the combinations of omics-based features, the Phylo+GI combination performed best also for naïve Bayes (F-scores: 0.784 and 0.760, by random forest and naïve Bayes, respectively). For SVM, the Phylo+GE combination showed the highest F-score (0.705). F-scores of feature combinations by the three classifiers correlated moderately. The correlation coefficient between random forest and naïve Bayes was highest, 0.828, that for random forest with SVM was 0.542, and between SVM and naïve Bayes it was 0.561. Our speculation for random forest outperforming SVM is that the fairly low number of features used in this work is probably more suitable for random forest than SVM, which is shown to perform well for a high dimensional feature space [127].

We also computed cross-validation F-score for the alternative imputation technique (termed as “probabilistic imputation”) and compared the result with the Random Forest Fscore shown in Fig. 2.19 with explicit imputation. The result is discussed in Supplementary Fig. A.4 with the conclusion that explicit imputation outperforms the probabilistic imputation.

To summarize this section, MP and non-MP can be classified very accurately by MPFit when GO terms of the proteins are available. Encouragingly, prediction can be made with a sufficient accuracy even when no function annotation is available using proper combinations of omics-based features. Missing feature imputation increases the coverage of proteins that are subject to prediction and also helps to improve accuracy by increasing the training data of a classifier. Among the three classifiers tested, random forest performed better than SVM and naïve Bayes.

2.5.2.3 Genome wide computational prediction of MPs

In the last section of this work, we report genome-wide prediction of MPs performed with MPFit on three genomes, S. cerevisiae (yeast), C. elegans, and human. We used two feature combinations that gave high performance in both F-score and coverage (Fig. 2.18): Phylo+GE+GI+DOR+NET and PPI+Phylo+GE. MPFit with the two feature combinations were run separately with explicit feature imputation and random forest as the last classifier. Then, proteins that were predicted as MPs by consensus of both runs were taken as plausible MPs. Consensus was taken to only count highly plausible MPs and avoid over-estimation of the MP fraction in the genomes. For MPFit runs with a feature combination, proteins were discarded if they had no features in the combination (i.e.

imputation was only applied if a protein had at least one feature in the combination). In the yeast genome, which has 6718 proteins in UniProt [86], there were 4673 proteins (Coverage: 69.6%) that had at least one feature among PPI, Phylo, or GE, and 5845 proteins (87.0%) that had at least one feature in Phylo, GE, GI, DOR, or NET. The coverages for C. elegans are 79.8% and 89.5%, while that for the human genome are 68.1% and 82.4% respectively for the PPI+Phylo+GE and Phylo+GE+GI+ DOR+NET feature combinations. The results are summarized in Table 2.1. A list of predicted MPs is available at http://kiharalab.org/MPprediction.

Table 2.1 Genome-wide prediction of moonlighting proteins

[a] The fraction of proteins that were subject to the prediction among all the proteins in the genome; [b] the number of known MPs listed in the MoonProt database that were predicted as MPs by MPFit; [c] the fraction of proteins that were predicted as MPs by MPFit among the proteins in the genome.

Genome	# Proteins	Coverage(%) [a]	Known MPs Predicted [b]	MPs (%) [c]
yeast	6,718	69.56	22/27 (81.4%)	10.97
C. elegans	20,133	79.82	1/1 (100%)	2.73
human	20,098	67.91	33/45 (73.3%)	7.82

First, we examined if known MPs listed in the MoonProt database in each genome were correctly predicted as MPs. The results in the second column from the right in Table 2.1 show that MPFit predicts known MPs reasonably well with recall of over 73% to each genome. C. elegans has only one known MP, which was correctly predicted by MPFit.

Next, we moved onto the blind genome-wide prediction to the three genomes. In the yeast genome, MPFit with the two feature combinations Phylo+GE+GI+DOR+NET and

PPI+Phylo+GE predicted 24.6% and 18.5% of the proteins as MPs, respectively, and among them, 10.9% of the proteins have a consensus prediction as MPs with the two feature sets. We note that this number of MPs in yeast is similar to the numbers obtained by a recent work by a different group [114]. In human, 67.6% of the total genome was subject to MPFit by both feature combinations, and 7.8% of the total genome was predicted as MP by consensus of the two feature combinations.

In C. elegans, 79.8% of proteins were subject to prediction by the two feature combinations. For this genome, the two feature combinations showed difference in the number of proteins predicted as MPs. With the Phylo+GE+GI+DOR+NET combination, 15.4% of the proteins were predicted as MPs while the fraction was 4.0% using the PPI+Phylo+GE combination, which resulted in a consensus of 2.73% of the proteins predicted as MPs. The fraction of predicted MPs by the latter feature combination was particularly lower than the other mainly because 48.5% of the predicted MPs by Phylo+GE+GI+DOR+NET were not subject to prediction with the PPI+Phylo+GE combination due to missing features.

To date there are two methods that predict whether a protein is moonlighting. A method by Chapple et al. considers a protein as MP if it is within an overlapping cluster in the PPI network and further passes a GO-based analysis. Out of the 45 known MPs in human in the MoonProt database, only 3 were predicted by this method (recall 0.0667) [113]. The second method by Pritykin et al. uses a GO-based multifunctional filtering criteria to predict MPs. Their method predicted 22 out of 45 known MPs in human (recall 0.4889) and 13 out of 27 known MPs in yeast (recall 0.4815) as MPs [114]. Thus, as

shown in Table 2.1, MPFit showed a larger recall (Table 2.1) in both human and yeast than the two existing methods.

2.5.2.4 Analysis of genome-wide MP prediction

We examined the functions of predicted MPs in the three genomes by considering GO [43] and KEGG pathway association [2]. In order to assign a protein to GO categories, we first mapped its GO annotations onto the terms at the second depth in the GO hierarchy, and performed GO enrichment analysis (NaviGO at http://kiharalab.org/web/compare.php). Table 2.2 lists the enriched GO categories of the predicted MPs. This GO analysis covers 100%, 99.3%, and 99.9% of predicted MPs in yeast, C. elegans, and human, respectively, which have GO annotations. Table 2.3 is a list of associations of the predicted MPs to KEGG pathways. Note that this analysis was based on the the predicted moonlighting proteins that exist in KEGG [2] database (66.36%, 35.21%, and 51.92% in yeast, C. elegans and human genome respectively).

In Table 2.2 and 2.3, the major proportion of MPs are enzymes. This observation is consistent with previous reports that many MPs were known primarily as enzymes when their secondary function was discovered [83,84,128].

Ribosome was listed as a KEGG pathway for the three genomes. An example is 40S ribosomal protein S3 (Uniprot ID: P23396) in human, which functions primarily as a ribosomal protein (part of the 40S subunit), and has a second function of being a subunit of a DNA binding complex involved in NF-kappaB-mediated transcription [129]. This protein has GO term GO:0003735 structural constituent of ribosome, which is a direct descendant of GO:0005198 structural molecule activity, and hence falls under the latter

category in Table 2.2. The second example of MPs is glyceraldehyde-3-phosphate dehydrogenase (GAPDH, Uniprot ID: P04406) in human. Besides its primary function as enzyme in the glycolysis pathway, this protein moonlights as interferon (IFN)-gamma-activated inhibitor of translation that silences ceruloplasmin mRNA translation [130]. In a proteomics study [131], this protein was identified as one of the urinary exosome proteins, and thus contains GO:0070062 extracellular exosome, which is a child term of GO:0005576 extracellular region, and hence falls in the latter GO category in Table 2.2. Both are these examples are correctly predicted MPs in human by the two omics based combinations Phylo+GE+GI+DOR+NET and PPI+Phylo+GE.

Table 2.2 GO categories of the predicted moonlighting proteins

GO category "Enzyme" is upon membership of either GO:0008152 metabolic process or GO:0003824 catalytic activity. The percentage of GO terms will not sum to 100% for a genome because a protein can have multiple assigned GO terms.

Genome	Enriched GO terms	MP (%)
yeast	enzyme (BP/MF)	91.86
	GO:0005488 binding (MF)	59.29
	GO:0032991 macromolecular complex (CC)	51.70
	GO:0071840 cellular component organization or biogenesis (BP)	42.61
	GO:0031974 membrane enclosed lumen (CC)	26.05
	GO:0005198 structural molecule activity (MF)	19.95
	GO:0009295 nucleoid (CC)	0.951
	GO:0016209 antioxidant activity (MF)	0.810
C. elegans	enzyme (BP/MF)	73.67
	GO:0005198 structural molecule activity (MF)	15.72
	GO:0002376 immune system process (BP)	3.47

	GO:0060089 mol. transducer activity (MF)	1.65
	GO:0004872 receptor activity (MF)	1.65
human	enzyme (BP/MF)	76.77
	GO:0005488 binding (MF)	63.84
	GO:0050896 response to stimulus (BP)	45.51
	GO:0032501 multicellular organismal process (BP)	38.19
	GO:0005576 extracellular region (CC)	36.54
	GO:0071840 cellular component organization or biogenesis (BP)	33.23
	GO:0051179 localization (BP)	29.03
	GO:0051704 multi-organism process (BP)	15.15
	GO:0040011 locomotion (BP)	10.18
	GO:0032991 macromolecular complex (CC)	9.41
	GO:0030054 cell junction (CC)	7.51
	GO:0000003 reproduction (BP)	7.26
	GO:0005198 structural molecule activity (MF)	7.07
	GO:0040007 growth (BP)	4.58
	GO:0031012 extracellular matrix (CC)	3.95
	GO:0009055 electron carrier activity (MF)	1.15

Table 2.3 KEGG pathway associations of predicted moonlighting proteins

Genome	Top 5 KEGG pathways	MP (%)
yeast	Metabolic pathways (KEGG ID 1100)	29.17
	Ribosome (3010)	15.33
	Biosynthesis of secondary metabolites (1110)	13.70
	Carbon Metabolism (1200)	6.92
	Biosynthesis of amino acids (1230)	6.38
C. ele-gans	Ribosome (3010)	13.79

	Metabolic pathways (1100)	12.34
	Purine Metabolism (230)	2.72
	Pyrimidine Metabolism (240)	2.54
	Oxidative phosphorylation (190)	2.54
human	Metabolic pathways (1100)	18.38
	Ribosome (3010)	4.45
	Olfactory transduction (4740)	3.94
	Purine metabolism (230)	2.54
	Cytokine-cytokine receptor interaction (4060)	2.42

2.6 Text mining approach for prediction of MPs ─ DextMP

All existing computational studies for moonlighting protein prediction, including our developed method MPFit overlook one major resource of information in automatic identification of MPs: text-based information that underlies in scientific literatures and textual description of protein functions in curated databases such as Uniprot.org [86]. Moreover, there are two existing online repositories that serve as experimentally validated resources for MPs [83,84], and they are built on expert knowledge with manual curation on existing literature, since in most of the cases MPs are not explicitly labelled in the database as “moonlighting”, “dual function”, “multitasking”, or related words. These latter two observations convinced us that a direct application of text mining techniques on MP literature would provide a major boost towards automatic MP prediction. To this aspect, in this work we propose a very first text mining based prediction algorithm for moonlighting protein classification.

For the last decade, text mining techniques has been extensively used to unravel non-trivial knowledge from structured/unstructured text data [132]. Text classification based methods consist of two broad steps: designing the best features, and modelling the classifiers. In terms of feature engineering, most of the existing works are based on bag-of-words that leverage some word related statistics in the text [133]. The next level of text-based feature learning models motivates from representing each text with a distribution of latent topics [134]. These latter topic modelling based representations are able to capture the semantic information underlying the text. In recent years, unsupervised deep learning based feature construction has become popular in text mining [135] as well as speech and image recognition [136,137]. Such deep learning based methods map text into a condensed d-dimensional continuous vector space such that semantically similar texts are embedded nearby each other. In this work we propose DextMP (Deep learning on tEXT for prediction of Moonlighting Proteins) which consists of four broad steps: first, it extracts textual information of proteins by mining scientific literature (publication title or abstracts) and functional descriptions in curated database of Uniprot.org[86]. Second, it undergoes a feature construction phase in order to represent each text with a k-dimensional feature. In this step, we apply a current state-of-the art deep unsupervised learning algorithm called paragraph vector [138] (termed as DEEP and PDEEP in our text, PDEEP as an extended deep learning), along with two other widely popular language models (TFIDF in the bag-of-words model category [132] and LDA in the topic modeling category [134]) in order to provide a comparison among the competitive text-based language models. Third, we use four machine learning classifiers to provide a MP/non-MP prediction on the text data based on the features learnt in the previous step.

Finally, we apply a text-to-protein mapping step to provide moonlighting protein prediction based on MP prediction on protein's associated text. Cross-validation results on the dataset of known moonlighting and non-moonlighting proteins (control dataset) show that DextMP can successfully predict MPs with over 94% accuracy with PDEEP as the language model. Overall PDEEP performs significantly better than the two other baseline models (TFIDF and LDA). However, even with the simple bag-of-words model TFIDF, DextMP achieves over 85% accuracy, which s direct evidence that textual data are rich with information that can be applied for MP prediction. Among the different forms of text information, protein's functional description in Uniprot.org provides better performance than the other two (title and abstract of scientific literature). Lastly, we have run DextMP with the best performing language models and text-based feature combinations on four genomes, Saccharomyces cerevisiae (yeast), Homo sapiens (human), X. laevis (frog), and C. pneumoniae (pneumoniae) and found that about 8~31% of the proteomes are potential moonlighting proteins. Comparison of DextMP with three existing MP prediction models, including our previously developed model MPFit that uses a diverse protein association features shows that DextMP significantly outperforms the others in specificity over known MPs and genome coverage.

2.6.1 Methods

Our method named DextMP (Deep learning on tEXT for prediction of Moonlighting Proteins) is developed to learn features from the text information available for proteins in Uniprot database [86] and literature in order to ultimately provide a prediction of

moonlighting protein class. In this section, we provide details of the framework of DextMP.

2.6.1.1 Data preparation

In order to construct a control dataset for our prediction model, we used the moonlighting and non-moonlighting protein, termed as MP and non-MP for short (negative example of moonlighting proteins) dataset that we built in our previous work [50]. In summary, the MP class of the control dataset consists of 263 MPs selected from the manually curated MP database MoonProt [84]. Only proteins that had Uniprot ID mapping in the MoonProt database were selected, and five MP proteins were further discarded to avoid redundancy as they had over 25% sequence identity with a paralogue protein in the set. To select the non-MP proteins, we applied the following GO-based criteria developed in our previous works [48,50] on top four genomes that are dominant in our dataset of MP, namely, human (45 MP, 17.1%), E. coli (29 MPs, 11%), yeast (23 MPs, 8.7%), and mouse (11 MPs, 4.2%): a protein was selected as a non-MP if it has a) at least 8 GO term annotations, b) when GO terms in the Biological Process (BP) category were clustered using the semantic similarity score [93] thresholds of 0.1 and 0.5, not more than one cluster was obtained at each threshold, and c) not more than one cluster of MF GO terms at semantic similarity scores of 0.1 and 0.5 were formed. In essence, a non-MP is a protein that has a sufficient number of GO annotations but they are not functionally diverse. We further ruled our non-MPs that had above 25% sequence identity with another non-MP sequences, and finally selected 162 non-MPs, among which 60 are from human (37.0%), 52 from mouse (32.1%), 34 from yeast (20.9%), and 16 from E. coli (9.88%). So in

summary, 263 MP and 162 non-MP were selected as control dataset for the DextMP model.

Table 2.4 Data size of DextMP model

[a]*number in the parenthesis indicates the number of proteins for which the text data was found*

	#proteins	#titles[a]	#abstracts[a]	#functions
MP	263	2496 (214)	1450 (158)	194
non-MP	162	1665 (162)	1624 (162)	162

2.6.1.2 Text extraction

Based on our control dataset, we extracted three categories of text information for each protein from Uniprot.org [86]: a) title of each of the reference citations of protein's record, b) abstract of each of the reference citations, and c) summary description of protein's function curated by Uniprot. The text data for category a) and c) were directly collected from Uniprot data dump, and for category b) we crawled the web links for abstracts available in the protein page of Uniprot.org. Table 2.4 shows statistics on the data size. Note that while one protein can have multiple titles and abstracts associated with it (one-to-many relation between protein and title/abstract), it only has one function description (one-to-one relation between protein and function description).

Upon extraction of raw text on our control dataset, we performed several layers of data cleanup. First, we discarded any redundant text information that appears both in MP and non-MP class. Second, we removed all stop words, punctuations, and special symbols (Greek letters) from the text. Finally, we performed stemming and lemmatization in

order to deal with the root and auxiliary forms of words, respectively [132]. We used the nltk package in python [139] for this cleanup operations.

2.6.1.3 Framework of DextMP

The overall framework of DextMP is shown in Fig 2.20. First, based on the control dataset of MP and non-MP, three categories of text information is extracted as raw data. Then a data clean-up step is carried out, and a dataset consisting of N texts of titles/abstracts/function descriptions is constructed (left panel Fig. 2.20). Then, in order to learn features for each of the text in this dataset, we apply a current state-of-the art deep unsupervised feature construction technique [138], along with two other widely popular language models (bag of words [132] and topic modeling [134]) to provide a thorough portrayal of text-based analysis on MPs. Then, based on these learned features, we used four machine learning classification algorithms, namely, logistic regression (LR), random forest (RF), SVM and gradient boosted machine (GBM) [140] to provide a MP/non-MP prediction on the text data. We emphasize that at this point of the DextMP model, a MP/non-MP class label is predicted at text level, i.e., for each of the texts that are associated to our control dataset of MP and non-MP, whether that be a title/abstract/function description. We call this first part of our DextMP model as text model, as shown in the left panel of Fig. 2.20.

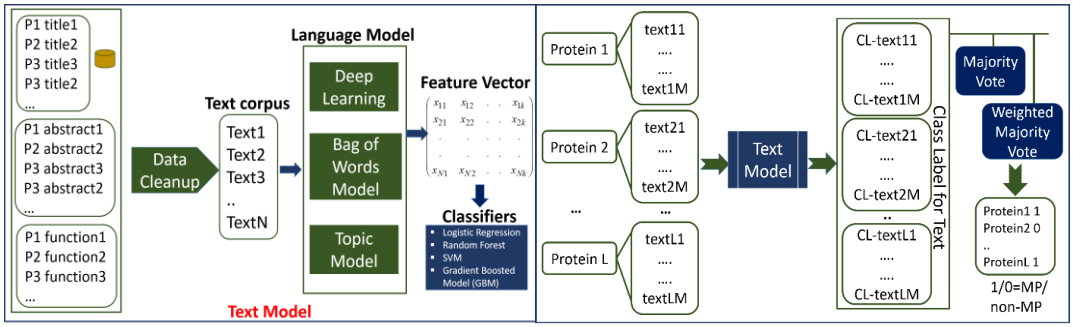

Figure 2.20 Schematic of DextMP: MP prediction by deep learning into text

Once we have a MP/non-MP class prediction for each associated text, we use the model shown in bottom panel of Fig. 2.20 to get a class prediction for the proteins (protein-level prediction). We start with the one-to-many mapping of L proteins to its associated M texts (title/abstract), receive the class labels (indicated as CL in right panel of Fig. 2.20) for the texts using our text model, and apply two heuristics to get the protein-level class label: in majority vote, we simply take the binary class label of protein as the majority class label of its associated text, and we applied different "majority" cutoffs to this end (50%, 70%, 80%, 90%). In weighted majority vote, we use the class prediction probabilities associated to the text instead of the binary class label to find the protein-level class label in the same way as above. Please note this latter part of the DextMP model shown in the right panel of Fig. 2.20 is only applicable to the protein-text data that has a one-to-many mapping, which in this case is title and abstracts, and not the function description.

2.6.1.4 Learning features from text

We apply the following four language models for feature construction from text:

1. Bag-of-words with **TFIDF**: Given a text corpus (collection of sentences/texts), this bag-of-words model first computes the dictionary that contains all the words in the text corpus. Given a dictionary of size N, a text can be represented as a N-dimensional real valued vector with TFIDF (short for Term Frequency-Inverse Document Frequency) [132] values for each word in the dictionary. Intuitively, TFIDF can statistically measure the importance of a keyword to a sentence with respect to its entire dictionary corpus. In this task of MP prediction, the TFIDF measure will help to identify the keywords that have more discriminative power towards MP related texts. For a word w, TFIDF is be computed as follows: TF(w) = (number of times word w appears in a text) / (total number of words in the text); IDF(w) = loge(total number of texts in the corpus/ number of texts with word w); TFIDF(w) = TF * IDF.

2. Topic Modeling with **LDA**: In practice, the bag-of-words model has two critical limitations: for a large dictionary, the size of the feature vector for each text can be huge, which makes it computationally expensive, and it does not take consideration of the word ordering in a text. To alleviate above two challenges, researchers in [141] model each text as a distribution of latent topics (user defined parameter) and each topic as a distribution of words. Latent Dirichlet Allocation (LDA) [134] is one of the most popular topic modeling algorithms in text modeling. LDA is a modification of earlier topic models [142] and uses two Dirichlet-Multinomial distributions to model the mappings between documents and topics, and topics and words. In the DextMP model, we use an open source python implementation of LDA [143] for feature representation of protein's text.

3. Unsupervised Deep Language Model **DEEP**: As our third language model, we use a deep learning based unsupervised feature construction algorithm [138]. This model maps texts into a continuous vector space of dimension d, such that semantically similar texts appears together i.e., forms a cluster. In a nutshell, for a sequence of words $W = \{w_0 , w_1, \dots , w_n\}$, where $w_i \in D$ (D is the dictionary) and a text T containing the sequence of words, the model maximizes $\Pr(w_i | w_0, w_1, \dots w_{i-1}, w_{i+1}, \dots w_n, T)$ over the text corpus. The training of feature vector representation of the text is done using stochastic gradient descent and the gradient is obtained via back-propagation [138]. For a given corpus of texts i.e. titles, abstracts, and functional descriptions, we apply an open source python implementation of the "paragraph vector" deep learning model [143] to find k-dimensional feature representation of each text.

4. **PDEEP**: Generally unsupervised deep language model requires large amount of training data for efficient feature learning. In DEEP the feature construction phase is based on the control dataset of MP and non-MP only. In PDEEP, we expand the training data to the entire protein's text corpus in Uniprot.org. Concretely, we extract a total of 1,060,520 titles available publication titles and 551,056 functional descriptions from the data dump of Uniprot.org to train the feature construction part of the PDEEP regardless of identifying whether the corresponding proteins of the texts are MP or non-MPs. Since publication's abstract is not available in the data dump, we omitted PDEEP training for abstracts.

2.6.1.5 Parameter tuning of DextMP

We use grid search to tune the parameters for the feature construction model LDA and DEEP. In LDA, we execute grid search in [61,61,86] as (min, max, step size) to tune

the “number of topics” parameter for different types of texts and classifiers. In DEEP, we tune three parameters of the paragraph vector model for the 4 classifiers: “minimum count” tuned in [56,69,69] with grid search, “window size” tuned in [44,59,69] and “dimension size” in [40,61,61]. For a word, “minimum count” indicates the minimum number of texts that the word must appear in, “window size” is the size of the convolution context and “dimension size” indicates length of the feature vector representation. For specific values of these parameters on different settings please see Table A.5. We also tune the parameters associated with the four classifiers of DextMP using grid search. For LR and SVM we tune the regularization parameter and use default values for other parameters in the model set by the sklearn’s [140] implementation. For RF and GBM, we tune the “number of trees” parameter and use default values of others.

2.6.2 Results

In this section we demonstrate results of our proposed method DextMP. The layout of this section is as follows: first we show a generic representation of MPs using the three categories of text data we leverage in this study. Second, we show cross-validation result of DextMP on text-level MP prediction. Third, cross-validation result on protein-level MP prediction is discussed. Lastly, we apply DextMP on genome-scale MP prediction and discuss the results along with model comparison with our previous MP prediction method, MPFit and two other external methods. Interesting case studies showing predictive power of DextMP over MPFit is shown.

2.6.2.1 MPs represented as text

In Fig. 2.21, we show word cloud of the three categories of text information that we used in DextMP to represent MP and non-MP proteins: publication title (Fig. 2.21A), function description in Uniprot (Fig. 2.21B) and publication abstract (Fig. 2.21C). Only the MPs from the control dataset are used in the visualization in Fig. 2.21.

From this generic text representation, a few points come to light: some of what we know about experimentally validated MPs are visible from this text representation, as words “enzyme”, “kinase”, “transcription” appear in all three text representations in Fig. 2.21. This is consistent with the previous reports that many MPs were known primarily as enzymes when their secondary function was discovered, in many cases which included acting as transcription factors [50,83,84,128]. The word “ribosome” appear as top word in Fig. 2.21, which is also consistent with our previous finding [50] where predicted MPs were enriched in ribosomal pathways in KEGG database [2], and moonlighting functions of ribosomal proteins were found in literature [144]. Additionally, words that are clear indicator of MPs also appear in text, such as “bifunctional” (word count in title is 21/0 for MP/non-MP), “multifunctional” (word count 12/0 for MP/non-MP). These initial findings lead us to develop more sophisticated text-based feature representation of MP and non-MP proteins, which will ultimately deliver successful MP predictions.

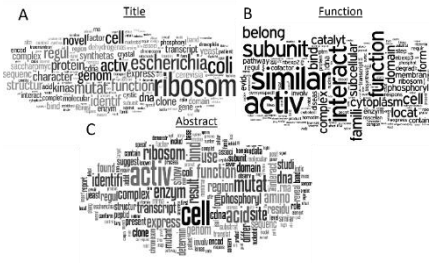

Figure 2.21 Word cloud of extracted text on MP dataset

2.6.2.2 DextMP performance on text-level prediction

Table 2.5 F-Score of DextMP on text-level prediction

	LR	RF	SVM	GBM
TFIDF-title	**0.7774**	**0.7942**	**0.8751**	**0.7218**
LDA-title	0.4071	0.5056	0.4372	0.5162
DEEP-title	0.6236	0.6005	0.6795	0.6157
PDEEP-title	0.6261	0.5436	0.6596	0.5935
TFIDF-abstract	**0.9220**	**0.8682**	**0.9371**	**0.8396**
DA-abstract	0.6102	0.6410	0.6220	0.6604
DEEP-abstract	0.6684	0.7420	0.7327	0.7069
TFIDF-function	0.7412	0.7439	0.7715	0.6947
LDA-function	0.5586	0.6000	0.5581	0.6781
DEEP-function	**0.7700**	**0.8181**	**0.8104**	**0.7369**
PDEEP-fufunction	0.7335	0.7166	0.7564	0.6816

We now demonstrate results of DextMP over 5-fold cross validation on our control dataset for text-level MP prediction in Table 2.5. A schematic of this part of the DextMP model is described in Fig 2.20 (top panel) and Methods section 2.2. As this is the first text information based analysis on MPs, along with running DextMP with two different deep learning based models (DEEP and PDEEP), we used two other popular language model categories (TFIDF in "bag-of-words" model and LDA in the "topic modelling" category) in order to provide a baseline for the text based learning. For each language model, three forms of text information (title, abstract, function description) were used separately (except the PDEEP-abstract combination, which was omitted out due to data unavailability). For MP classification on the learned features, we further use four classifiers LR, RF, SVM and GBM (shown in columns of Table 2.5). See Methods about parameter tuning of these different models.

For each type of text information (title, abstract, function), the best performing model under each of the 4 classifiers is in boldface in Table 2.5. Among the four different language models, DEEP and PDEEP clearly outperforms the baseline model LDA in all three text information categories and all four classifiers. Largest gap in F-Score in this comparison is 0.2523, between LDA-function-SVM and DEEP-function-SVM combinations. In terms of comparison with TFIDF, DEEP shows superior performance than TFIDF in the function category, while TFIDF shows better performance in the other two text categories, i.e., title and abstract. The largest win for DEEP over TFIDF is at the function-RF combination, with F-Score gap of 0.0741. However, TFIDF shows better performance with a much higher margin in the abstract-SVM combination (F-score gap 0.2044).

Overall, in the title category, TFIDF-title-SVM has the best F-score 0.8751 (precision 0.8920, recall 0.8640). In the abstract category, the best combination is again TFIDF-abstract-SVM (F-Score, precision, recall of 0.9371, 0.9376, and 0.9369, respectively). In the function category, DEEP-function-RF stands as the best model (F-Score, precision, recall of 0.8181, 0.8311, and 0.8161, respectively). Overall performance of abstract is superior to the title and function categories. From Table 2.5, the (min, median, max) of the F-scores shown under the abstract category is (0.6102, 0.7198, 0.9371), while the same for function and title are (0.5581, 0.7352, 0.8181) and (0.4071, 0.6197, 0.8751), respectively. Among the 11 different setting (rows in Table 2.5), SVM classifier shows better result than other three (LR, RF, GBM) in 6 cases, RF wins for 3 cases and GBM for 2 cases.

PDEEP was built as an extension from DEEP by enlarging its training set to the whole corpus of proteins in Uniprot.org. While this model shows comparable performance with DEEP in the title category (largest gap of F-Score 0.0569 with title-RF), DEEP shows clearly better performance than PDEEP in the function category (largest F-Score difference 0.1014 at function-RF). Our speculation behind this poor performance of PDEEP is that because of large training data, textual features that are unique for MP becomes somewhat generalized in the PDEEP's feature representation compared to DEEP. An example showing evidence of this speculation is the S13 ribosomal protein in human (Uniprot ID P62277) for which DEEP correctly made a MP prediction, while PDEEP failed. According to MoonProt database [84], apart from being a ribosomal protein, it moonlights by inhibiting the splicing own RNA transcript and inhibiting the removal of intron 1 from rpS13 mRNA [145]. A text data describing the first function is belongs to the ribosomal protein S15P family which appears once in the training dataset for DEEP, while appears 989 times in the extended training dataset for PDEEP. Besides, words describing this proteins moonlighting function, such as "intron" and "RNA splicing" has very different counts in the training dataset of DEEP and PDEEP ("intron" appears 17 times in DEEP and 1391 times in PDEEP, "RNA splicing" appears 7 times in DEEP and 734 times in PDEEP). This gives indication that the larger training dataset reduces the uniqueness of MP features in this case.

In terms of computation time, TFIDF, LDA, DEEP, and PDEEP shows different performance when the total computational time is broken into three phases: training, inference of features for each text, and text classification. In the training phase, for (title, abstract, function), computational time for TFIDF, LDA and DEEP are (0.1457, 0.6172,

0.1350), (2.5952, 5.3780, 3.2526), and (1109.49, 1659.13, 253.61) seconds, respectively. For the inference phase, the same for TFIDF: (2.6048, 7.8240, 0.6350), LDA: (1.0952, 1.6222, 0.2909), and DEEP: (0.6052, 0.5905, 0.0576). For the last phase, classification, TFIDF: (168.95, 299.143, 19.83), LDA: (12.69, 9.80, 1.51), and DEEP: (30.64, 31.43, 1.73). Since the first phase training can be pre-computed based on the control data once and be reused later, it is evident that DEEP can be used much more efficiently than LDA and specially TFIDF (significantly at the classification phase), in terms of computational time.

In summary, with text-based representations, simple bag-of-words model such as TFIDF achieves over 93% accuracy (with TFIDF-abstract-SVM). The DEEP model shows superior performance when the Uniprot function description is used to represent the text information for the proteins with random forest as the final classifier, and achieves a highest F-score of over 81%.

2.6.2.3 DextMP performance on protein-level prediction

In this section we discuss the performance of DextMP over 5-fold cross validation when the text-level MP/non-MP class prediction demonstrated in the previous section is mapped to protein-level class prediction. A schematic of this part of the DextMP model is described in Fig 2.20 (bottom panel) and Methods section. For rest of the two text categories (title/abstract), in order to perform the text-to-protein mapping of the MP/non-MP class labels predicted on the text, we resort to two schemes: majority voting and weighted majority voting. For each combination of text information (title/abstract), language model (TFIDF/LDA/DEEP/PDEEP), the classifiers (LR/RF/SVM, GBM), and for both

weighted and non-weighted majority voting cases, we ran DextMP over 5-fold cross validation with different majority vote cutoffs (50%, 70%, 80%, 90%) (Supplemental Fig. A.5-A.8) and selected the optimal cut-off for each combination. In Fig. 2.22 we show a comparison between the protein-level F-scores at the optimal majority vote cutoffs for the weighted and non-weighted cases. Results in the lower triangle in Fig. 2.22 are the cases where non-weighted majority voting performed better than the counterpart. Although the F-scores differs insignificantly between these two cases in Fig. 2.22, the non-weighted scheme still wins in most of them. So for the rest of the results in this work we use only the non-weighted majority voting with the optimal voting cutoffs.

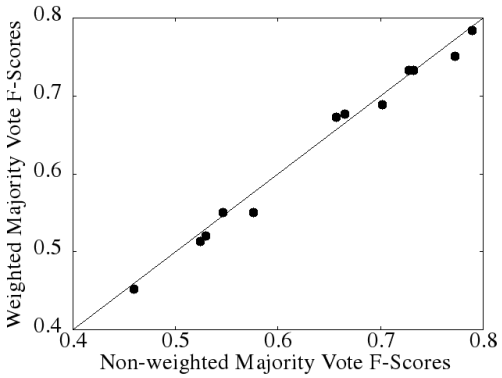

Figure 2.22 Weighted and non-weighted majority voting comparison

F-scores for weighted and non-weighted majority voting at optimal voting cut-offs

Table 2.6 shows the result for 5-fold cross validation on protein-level MP prediction. Similar to Table 2.6, for each type of text information (title, abstract, function), the best performing model under each of the 4 classifiers is in boldface in Table 2.6. Here also, both DEEP and PDEEP models clearly outperforms the LDA model in all three text categories. In the title category, TFIDF shows better result than DEEP for 3 out of 4 classifiers (largest F-Score gap 0.0655 at title-RF), while DEEP wins in 1 out of 4 cases.

TFIDF-title-SVM has the best F-Score in the title category (F-Score 0.8330, precision 0.8479, recall 0.8316).

Table 2.6 F-Score of DextMP on protein-level prediction

Benchmark F-Score of DextMP over 5-fold cross validation on protein-level prediction. LR – Logistic Regression, RF – Random Forest, SVM – Support Vector Machine, GBM – Gradient Boosted Model

	LR	RF	SVM	GBM
TFIDF-title	**0.7703**	**0.7474**	**0.8330**	0.6901
LDA-title	0.5129	0.5708	0.5017	0.5363
DEEP-title	0.7291	0.6819	0.7766	**0.7116**
PDEEP-title	0.6611	0.5079	0.5159	0.6067
TFIDF-abstract	**0.8132**	0.8225	**0.8208**	0.7833
DA-abstract	0.5351	0.5554	0.5458	0.6014
DEEP-abstract	0.7998	**0.8325**	0.7963	**0.7897**
TFIDF-function	0.7412	0.7439	0.7715	0.6947
LDA-function	0.3978	0.5308	0.3878	0.5271
DEEP-function	**0.7700**	**0.8180**	**0.8104**	**0.7369**
PDEEP-function	0.7335	0.7166	0.7564	0.6816

In the abstract category, TFIDF and DEEP shows has equal wins for the 4 classifiers, and DEEP-abstract-RF has the best F-score (0.8325, precision 0.8402, recall 0.8323). For the function category, DEEP wins over TFIDF for all three classifiers, with best F-Score of 0.8180 (precision 0.8311, recall 0.8161). So overall at the protein-level MP prediction, DEEP outperforms TFIDF and LDA by showing better F-Score in 7 out of 12 cases. Intuitively, the DEEP model's superior performance is evident from how these models are built. The bag-of-word models relies on word count (TFIDF) and do not consider more intricate relationships such as ordering of words [132]. LDA is at coarse-grained level over the bag-of-words models as it captures the latent topic distribution of

the text [134]. On the other hand, the deep learning based models are able to capture the semantic relationship within words in a text [138].

Similar to the text-level prediction, the abstract category shows overall better performance than the title and function category in the protein-level prediction as well. From Table 2.6, the (min, median, max) of the F-scores shown under the abstract category is (0.5351, 0.7930, 0.8325), while the same for function and title are (0.3878, 0.7352, 0.8180) and (0.5017, 0.6715, 0.8330), respectively. Among the 11 different setting (rows in Table 2.6), the RF classifier shows better result than other three (LR, RF, GBM) in 5 cases, SVM wins for 4 cases, and both GBM & LR win for 1 case. The highest overall F-Score at protein-level MP prediction in Table 2.6 is 0.8330 (precision 0.8479, recall 0.8316) by the TFIDF-title-SVM combination which is very close to the DEEP-abstract-RF setting (F-score 0.8325, precision 0.8402, recall 0.8323).

Although abstract based models excel in both text-level and protein-level MP prediction, practically it is not usable for large-scale predictions as the data is not directly available in the Uniprot knowledgebase. Hence we chose top four models from Table 2.6 under the title and the function category (i.e., TFIDF-title-SVM, DEEP-function-RF, DEEP-function-SVM, and TFIDF-title-LR) in order to perform blind predictions on genomes using DextMP (described in next subsection).

Table 2.7 Genome-scale prediction by DextMP

Genome	**yeast**	**human**	**X. laevis**	**C. pneumoniae**
#proteins	6,721	20,104	11,078	1,110
Coverage	96.73%	98.06%	30.54%	38.74%
%MP (vote >= 3)	2,438 (36.27%)	4,657 (23.16%)	543 (4.90%)	368 (33.15%)
%MP (vote > 3)	2,008 (9.98%)	1080 (16.07%)	331 (2.99%)	331 (29.82%)
#known MP	23	45	NA	NA
recall (vote>= 3)	0.8889	0.9333	NA	NA
Recall (vote > 3)	0.7404	0.7111	NA	NA

2.6.2.4 Genome-scale prediction of MPs using DextMP

In this section we show results of DextMP model for two genomes on which MP prediction has been performed before: *S. cerevisiae* (yeast), and *H. sapiens* (human), and two genomes novel genomes for which MP prediction was not possible by other models due to lack of data: *X. laevis*, and *C. pneumoniae*. In order to perform genome prediction, we used the title and function description as protein's text information, ran four best performing models (TFIDF-title-SVM, DEEP-function-RF, DEEP-function-SVM, and TFIDF-title-LR) and took the consensus of the predictions. Previously, we have performed genome prediction with our diverse protein association feature based model MPFit [50], and showed that it outperformed two existing models that predicts MP: method by [113] identifies proteins that are members of overlapping clusters in the PPI network and predicts a subset of them as MP by further GO based analysis. The second method by [114] developed a GO based multifunctional filtering criteria to predict MPs. In this section, we discuss comparison of DextMP with MPFit and these two other models for MP prediction in yeast and human genome.

Table 2.7 shows the genome results. In yeast genome, out of 6,721 proteins, 6,500 had both title and function description in Uniprot.org (coverage 96.73%). Among these proteins, 2, 438 are predicted as MP by DextMP consensus. We computed recall of this prediction out of the 27 known yeast MPs in MoonProt [84], and found that 24 of them were predicted correctly (recall 0.8889) by the majority vote consensus (at least 3 MP votes out of 4 DextMP models). This performance is higher than what we achieved with our previous model MPFit (recall 0.8146) which in turn outperformed another existing model by [114], that predicts 876 proteins as MP in the yeast genome, with recall of 0.4815. Note that apart from outperforming the two models MPFit and the model by [114] in terms of recall in yeast genome prediction, DextMP also has much higher coverage than both (coverage for DextMP 96.73%, MPFit 69.56% and [114] 68.69%). With a more stringent consensus protocol (4 MP votes from all 4 DextMP models), the recall over known MP was 0.7404, which is lower than MPFit but higher than the model by [114]. 9.98% of the yeast genome was predicted as MP with this more stringent consensus voting.

In human genome, out of 20, 104 proteins, 19, 713 proteins had both title and function descriptions and could be applied in DextMP (coverage 98.06%), which is higher than both MPFit (coverage 67.91%), work by [114] (coverage 48.08%), and [113] (coverage 64.01%). Out of 45 known MPs in human, 42 are predicted correctly by DextMP (recall 0.9333) when majority voting was applied among the 4 DextMP models (vote >= 3 in Table 2.7), which outperforms all existing models that predicts MP on human by a large margin (MPFit recall 0.7333, [113] 0.0667, [114] 0.4889). With the stringent consensus voting (vote > 3 in Table 2.7), recall was 0.7111 which is lower than MPFit but

higher than both the models by [114] and [113]. 16.07% of the human genome was predicted as MP with this more stringent consensus voting.

So, in summary, DextMP outperforms MPFit and other two MP prediction models in two aspects: in correctly predicting known MPs (recall) with recall as high as 91%, and in coverage, i.e., applicability of the models in the genome corpus. Applicability of the model by [113] relies on availability of proteins in PPI database. For MPFit the coverage depends on availability of proteins in a number of protein association databases including PPI, and the model by [114] solely depends on GO annotation availability. Since DextMP can be applied to any protein that has textual information in Uniprot, it have much larger coverage than the other existing models.

As observed in the higher coverage result by DextMP above, a major advantage of DextMP is that it solely relies on text information of proteins, unlike other available methods including MPFit which cannot be applied for proteins/genomes that lack experimental studies (such as PPI, gene expression etc.). To this aspect, we ran two other genomes with DextMP that are non-applicable for MPFit and the two other existing models compared above due to lack of experimental studies: *X. laevis* and *C. pneumoniae*. The result is in the last two columns of Table 2.7. For *X. laevis*, out of 11,078 proteins, 30.5% has function text information in Uniprot, and DextMP predicted 543 (4.90%) as MP with majority voting. For *C. pneumoniae*, out of 1,110 proteins, 430 proteins has text information in Uniprot, and DextMP predicted 368 of them as MP. The two latter results show the wider applicability of DextMP over other existing models.

We now provide three case studies where DextMP correctly predicts a protein as moonlighting and our previous method MPFit fails. First of these cases is a band 3 anion

transport protein in human (Uniprot ID P02730). As the primary function it transports inorganic anions across the plasma membrane, and as the moonlighting function it acts as scaffold protein providing binding sites for glycolytic enzymes [146]. MPFit model fails to predict this as MP because this lacks features in four out of six different feature domains of MPFit, i.e., lack of data in PPI, phylogenetic profile (PHYL), genetic interaction (GI) and interaction network properties (NET), upon which MPFit model applies machine learning classifiers for MP prediction. However, this protein has functional description in Uniprot.org [86], which provides a clear textual depiction of it's two functions, such as: *functions both as a transporter that mediates electroneutral anion exchange across the cell membrane and as a structural protein*, and *interactions of its cytoplasmic domain with cytoskeletal proteins, glycolytic enzymes, and hemoglobin*. Based on this text, DextMP extracts features and finally makes a correct MP prediction.

Our second case study of successful prediction by DextMP is protein PHGPx (Uniprot ID P36969) in human. Primary function of this MP is cell protection against membrane lipid peroxidation and cell death; moonlighting function is the protein's structural role in mature spermatozoa [147]. In MPFit feature space, this protein lacks PHYL, GI and disordered region features (DOR). From its existing PPI features, it is evident that it's interacting proteins form tight clusters even at high clustering thresholds (number of clusters relative to the number of interacting proteins stays as low as 0.3 for high clustering cutoff), so based on these MPFit incorrectly predicts it as a non-MP. However, the protein's functional description in Uniprot includes texts that indicates both it's functions, such as *protects cells against membrane lipid peroxidation* and *required for normal*

sperm development and male fertility, which finally results in a correct MP prediction by DextMP.

Our final example is protein Gephyrin (Q9NQX3) in human. This protein anchors transmembrane receptors by connecting membrane proteins to cytoskeleton microtubule binding protein. It's moonlighting function is biosynthesis of the molybdenum cofactor [148]. In MPFit feature space, this protein lacks PPI, PHYL, GI and NET features. Although its GE features show high number of clusters of co-expressed partners, MPFit fails to predict it correctly when it combines features from multiple domains. DextMP makes correct prediction for this protein as it's function description include *microtubule-associated protein involved in membrane protein-cytoskeleton interactions,* related to it's first function, and *catalyzes two steps in the biosynthesis of the molybdenum cofactor,* related to the second function. These examples clearly show different scenarios where DextMP provides successful MP prediction through its powerful feature inference from textual data and also higher applicability/coverage compared to existing models.

CHAPTER 3. GROUP FUNCTION PREDICTION

3.1 Background

The second part of this research addresses yet another divergence from the one-protein-one function paradigm by investigating *group function* of proteins. With the overwhelming development of genomic and proteomic technologies, massive amount of proteomic data becomes available. Consequently, the computational challenge of correctly annotating protein's function and explaining the mechanisms through which multiple proteins interact in a cell toward a common phenomenon becomes ever more important. Intuitively, proteins interact in a cell physically, through gene expression or genetic interaction to commemorate a common function that so often ends up causing a disease/disorder. To understand the functional nature of a set of proteins, it is often important to understand the biological process/molecular function/cellular location the proteins are involved in as a group, rather than understanding the detailed functional characteristics of the individual proteins in the group. More often than not, biological experiments reveal sets of proteins involved in a disease/disorder, co-expressed together, or phylogenetically correlated together without sufficient explanation of the functional mechanisms of these group activities. The perspective of "group" function annotation to a set of proteins opens up novel possibilities of understanding the functional nature of complex cellular interactions of such protein groups.

The problem of building computational model to directly predict group functions of a set of proteins is both unique and significant. The present bioinformatics approach that comes closest to the notion of group function is the Gene Ontology (GO) term enrichment analyses based on the functions of known proteins, a direction often used to come to a consensus functionality of a set of protein groups. However, the major drawback of such an approach is that it is based on identified protein functions/GO terms, which is an often sparse knowledge for a group of novel genes found to be involved in disease related phenomenon. As a related effort to this problem, in [149], the authors performed SNP-targeted GWAS studies to identify set of genes involved in the Rheumatoid Arthritis disease and then clustered the PPI network to identify the gene group's common biological pathways in the KEGG [2] database. However, both these latter methods lack an integrative perspective when accounting for the multitudes of levels of associations that the gene groups might be involved in the cell for causing the targeted disease/phenomenon, when comprehending their group functions.

In this study, we propose a novel computational method called Group Function Prediction (GFP) that uses experimental data to predict the function of a protein group, even when individual protein functions cannot be reliably predicted by taking into account protein's interaction networks as well GO annotations in the existing databases. The key concept underlying group function prediction is considering function in the context of functional and physical interaction relationships of genes. To implement this strategy, we use an integration of a number of individual type of protein interaction networks – physical protein-protein interaction (PPI), gene co-expression network (GE), phylogenetic profile similarity network (Phyl), gene ontology (GO) similarity network

and KEGG pathway similarity network. Fig. 3.1 shows a schematic diagram of the GFP model. Briefly, it takes a group of target proteins pre-identified to be involved in disease/disorder as input (1), and builds an integrated interaction network with the target proteins and other proteins in the same organism. We use a network integration tool – similarity network fusion (SNF) [150] to integrate the information of multiple protein interaction network. (2) Then, proteins are clustered using the affinity propagation method [151] based on the similarity of integrated features. The target proteins are grouped in a cluster with some other proteins, whose function will be predicted iteratively in the subsequent steps. Each gene cluster will be assigned GO terms by the majority vote of its component genes. Some clusters remain un-annotated if they do not contain enough annotated genes. (3) Then, GFP predicts function of the un-annotated clusters using a Conditional Random Field (CRF) framework [152]. The essence of the CRF module is to predict cluster functions in the network based on the functional properties of the cluster neighbourhood. (4) Subsequently, GFP propagates the new CRF cluster GO labels to the unknown proteins in the each cluster so that it reflects the group function predicted by the CRF module in the previous step (Fig. 3.2, see Methods for detail). (1') Now that the GO term annotations of genes are updated, protein networks are integrated again with the updated GO similarity network, and computation (1) to (4) is iterated until the function assignments to the groups/clusters between successive iterations come to an agreement, or sufficient number of iterations have been reached.

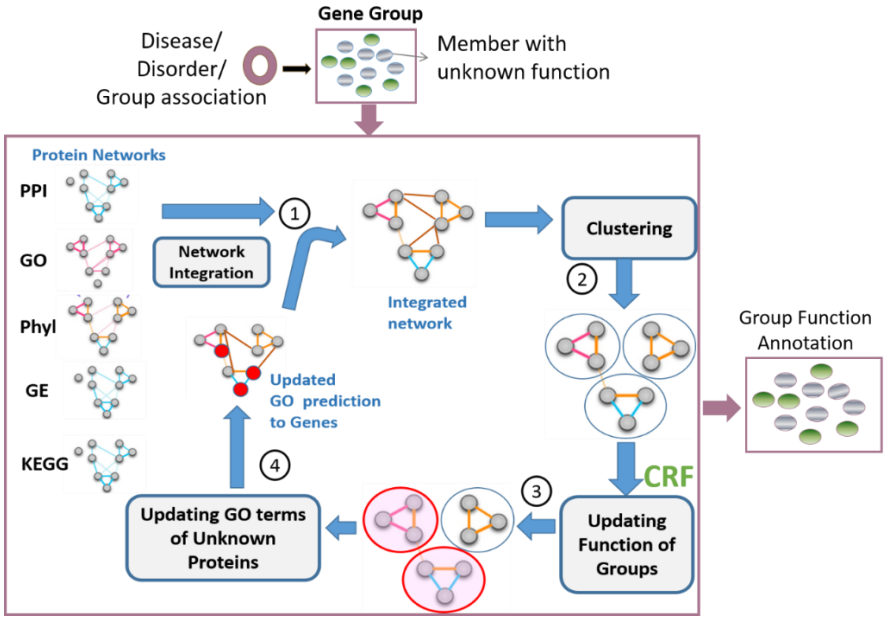

Figure 3.1 Schematic diagram of the group function prediction (GFP) model

Iterative procedure of group function prediction. In (3) and (4), clusters/proteins in red are updated with their predicted GO annotations. PPI, protein-protein interaction; Phyl, phylogenetic profile; GE, gene expression; KEGG, pathway similarity.

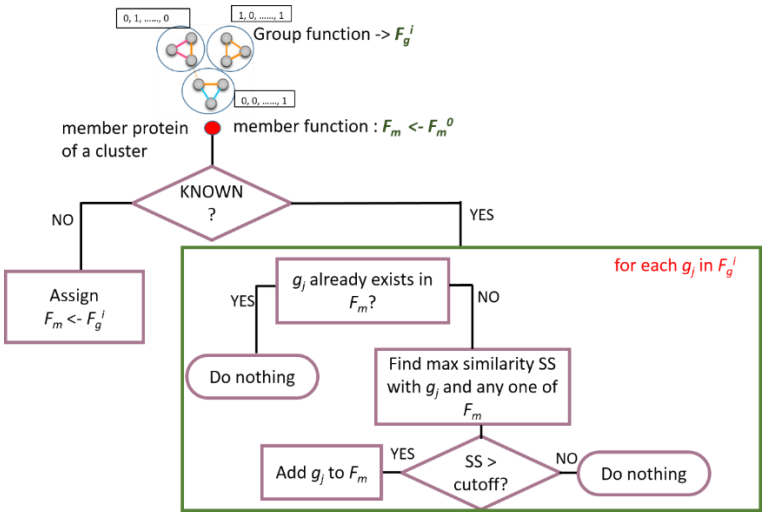

Figure 3.2 Assignment of protein's function derived from the group function

Step 4 of the GFP pipeline shown in Fig. 3.1.

3.2 Methods

3.2.1 Network construction

The backbone of our GFP model is an integrated network of protein-protein association. We choose the human genome to construct the backbone network, as our initial target dataset is from human. We use five resources to construct individual protein interaction network, and then use a network integration tool to combine them.

1. Protein-protein interaction (**PPI**) network – we construct PPI network using the high confidence physical interactions (>0.7 confidence score) of STRING database [96]. From Human proteins (NCBI taxID 9606), a total of 15,036 genes had high confidence interactions in PPI.

2. Phylogenetic Profile network (**Phyl**) – We construct the phylogenetic profile network by taking all interactions from the STRING database that has medium confidence score (>0.4) in any of the following criteria – "neighborhood", "fusion", "co-occurance"[96]. A total of 1197 human genes had medium conifedence phylogenetic profile interactions.

3. Gene Ontology (**GO**) similarity network – For all human proteins, GO annotation is taken from the uniprot database[86]. GO similarity score is computed by the funsim score using BP and MF GO ontology [93]. Two proteins are chosen to have a GO interaction if they have a funsim score above cutoff (0.7).

4. Gene expression (**GE**) network – We extracted gene expression profiles in Human genome from the COEXPRESdb database [115]. We calculated the Pearson correlation coefficient of expression levels of each pair of genes and selected pairs as co-

expressed if the absolute value of the correlation coefficient ranked within the top 2% largest values among all the pairs. A total of 17,341 human proteins were extracted from the database with gene expression profiles.

5. **KEGG** pathway association – We mapped all human genes to KEGG pathways[2]. There were 287 unique pathways found in the 23,658 human genes in KEGG database. We constructed a binary vector of length 287 indicating existance/non-existance of a certain KEGG pathway for each of the human gene, and then computed a cosine similarity between two binary vectors (p) of genes g and g' as a pathway similarity score- $s_{gg'}^{KEGG} = \frac{p_g . p_{g'}}{| p_g \| p_{g'} |}$. We used a score cutoff of 0.2 to selected associated genes in the KEGG network.

3.2.2 Network integration

We use a non-linear message passing based method by Wang et al. [150] to integrate the individual networks described above. We use the R package used for this method (SNF in short for Similarity Network Fusion) that takes multiple networks in terms of similarity matrices. Each matrix is equivalent to a similarity network where nodes are proteins and weighted edges represent pairwise protein similarity. SNF then iteratively fuses the networks by a non-linear method based on message passing theory that iteratively updates every network, making it more similar to the others with every iteration. Within a few iterations, SNF converges to a single network.

3.2.3 Concise Gene Ontology (GO) vocabulary for predicting group function

The GO vocabulary[153] has over 40,000 GO terms and to include everything in our function prediction model would result in significant slow-down of the model run-time. In order to use a concise functional vocabulary for our GFP model, we used the concept of Slim GO terms. Slim GO terms are a cut-down version of the GO ontologies containing a subset of the terms in the whole GO and are selected and maintained by the GO consortium [43]. In order to get sufficiently detailed annotations for our predicted group functions, we used a customized ontology slim that can be applied to specific annotated datasets and exploits latent information in the structure of the ontology graph and in the annotation data [154]. In this method, input annotation terms are mapped to the slim term(s) in closest proximity to the annotation term in the path(s) from the annotation term to the root node. We mapped all the direct GO annotations of 14,885 human proteins into this customized ontology set 303 GO terms. The depth of the customized slim terms can be controlled in the method by a parameter called information content (IC), which refers to information carried by a node based on its annotation and its position within the DAG. We used an IC cut-off of 0.3 as recommended by the authors [154] for generating our GO slim dataset. Thus we limit our function prediction vocabulary to these 303 slim GO terms.

3.2.4 Affinity propagation based clustering method

The affinity propagation based clustering method clusters data by employing an idea of passing messages between them [151]. It was shown to have a low error rate and

fast as compared to other common clustering methods. The number of clusters is influenced by the so-called preference parameter, Setting the preference parameter to the median of the input distances results in a moderate number of clusters and setting them to the minimum of input distances results in a smaller number of clusters. We used the R library “apcluster” for this method.

Clustering of the integrated network (step 2 in Fig. 3.1) of the GFP model is based on a mean of two type of inter-node distances: integrated network’s edge weights outputted from SNF, and a functional similarity score (*funsim*) [93] of protein pairs based on their GO term annotations.

3.2.5 Protein function prediction model using Conditional Random Field (CRF)

Network models can model a biological network data to predict protein function. A graphical model is able to represent complex joint distributions of a large number of variables compactly using a set of local relationships specified by a graph. Each node in the graph represents a random variable and nodes are connected by edges, which describe the dependency between the variables. Probabilistic graphical models can model the entire network simultaneously, and incorporates information of protein function and interactions according to the edges defined in the graph.

Markov Random Fields (MRFs), is a probabilistic graphical model that have been used previously to predict protein functions based on network data. Deng et al. [155] laid the basic framework of an MRF model that predicts protein functional annotation from PPI network. Kourmpetis et. al.[156] extended this model by improving parameter estimation through multiple parameter estimation steps. Other approaches exist that uses

MRF to integrate multiple sources of information to predict protein function from network [157,158]. A MRF-based framework basically models relationships between the input data and assumes independence between them. Conditional random fields are discriminative version of MRFs which model the dependence of the output on the local graph neighborhood input rather than the full joint distribution of the input and the output. Previously, Gehrmann et al [159] used CRF to predict protein function by integrating multiple network resources. In this scope, we extend the work by Gehrmann et. al.[159] by including protein's functional association in the graph neighborhood and build an independent CRF-based function prediction module that we use in our GFP pipeline shown in Fig. 3.1.

We use CRF for predicting GO terms to groups in step (3) of Fig. 3.1. A graphical model such as CRF is able to compactly represent complex joint distributions of a large number of variables using a set of local relationships specified by a graph. CRF can model the entire network simultaneously, and incorporates protein function and interaction information using the edges defined in the network. A CRF computes the probability of having binary labels *Y* (here whether proteins have a particular GO term annotation) given parameters θ and input variables *X* (the protein features provided in the integrated network):

$$p(Y \mid \theta, X) = \frac{1}{Z(X)} \prod_{c \in C} \psi_c(Y_c, X) = \frac{1}{Z(X)} \prod_{c \in C} \{\psi_{c,s}(y_i; \theta, X) + \psi_{c,p}(y_i, y_j; \theta, X)\} \quad \text{(Eqn. 3.1)}$$

where *Z(X)* is a normalization factor, *c* is a clique, and *C* is the set of all cliques in the graph. The rightmost part of Eq. 3.1 shows that the probability is computed from two

terms, a single term $\Psi_{c,s}$, which considers the GO term label y_i of one protein, and a pairwise term $\Psi_{c,p,}$, which takes into account neighboring proteins' GO term labels, y_i and y_j. The two terms are defined concretely by potential functions as:

$$\psi_{c,s}(y_i;\theta,X)+\psi_{c,p}(y_i,y_j;\theta,X)=\exp\{U_s(y_i;\theta,X)\}+\exp\{U_p(y_i,y_j;\theta,X)\} \quad \text{(Eqn. 3.2)}$$

$$U_s(y_i;\theta,X)=\sum_{j\in N_1} w_1 P(y_i \mid y_j)+\sum_{k\in N_0} w_2 P(y_i \mid y_k) \quad \text{(Eqn. 3.3)}$$

and

$$U_p(y_i,y_j;\theta,X)=w_3 y_j e(i,j)+w_4(1-y_j)e(i,j)+w_5 y_j\, funsim(i,j)+w_6(1-y_j)\, funsim(i,j) \quad \text{(Eqn. 3.4)}$$

Eq. 3.3 represents a single term where the probability of labels depends only on features of each node, while Eq. 3.4 are a pairwise term where dependency of neighbouring labels is expressed. In the single term (Eq. 3.3), N_1 and N_0 are the number of GO terms that annotate/do not annotate the protein (i.e. 1s and 0s in the GO annotation vector for the protein), and $P(y_i|y_j)$ is the function association score developed previously in our group[12,13], which basically expresses the conditional probability that y_i is assigned simultaneously with y_j to each sequence in UniProt,. Thus, annotation y_i for a protein depends on existing GO annotation of the protein. In the pairwise term that considers proteins i and j, $e(i, j)$ is the edge weight of the two proteins in the integrated network (Fig. 3.1) and $funsim(i, j)$ is the functional similarity [89] between protein i and j. Weights w_3 to w_6 control the influence of the neighbouring proteins when the protein has the GO term ($y_j = 1$) and when it does not ($y_j = 0$). A previous work using CRF for function prediction from a PPI network [98], used only a pairwise factor. In comparison with their work, the

GFP model proposed above would be advantageous because GFP can consider the coherence of the GO term's annotation or lack thereof relative to the existing GO terms for the protein. Using the equations above, the conditional probability of a protein annotated with a GO term, $p(y_i = 1 \mid Y_{-i}, \theta, X)$, can be expressed in terms of the logistic function [99]. Parameters of the GFP model are trained using a Metropolis-Hastings framework and inference is done using Gibbs sampling.

3.2.5 Assignment of protein's function derived from the group function

At the last step of the GFP model (step 4 of Fig. 3.1), we update the GO annotations of the individual proteins according to their cluster/group function predicted by the CRF module. The procedure of this step follows from Fig. 3.2. Here, $F_g^{\,i}$ denote list of GO terms for the group (cluster) after iteration i and F_m denote the same for an individual member protein. If the protein is an unknown protein with no GO annotations in Uniprot database [86], we directly assign the group function F_g as it's member protein function F_m. Otherwise, for each new GO term g_j in the $F_g^{\,i}$ list, we check the maximum similarity score SS between g_j and any GO term in F_m. We used relevance semantic similarity score [93] as the SS score for within-domain (Biological Process, Molecular Function, and Cellular Component domains) GO pairs, and the function association matrix (FAM) score previously developed in our group [12,13] for cross-domain GO pairs. If the SS score is above a pre-defined cut-off, we add the group function g_j to F_m. After this step, the GO annotations of all the individual protein nodes in the integrated graph is updated according to their respective group functions, i.e., cluster annotations predicted by the CRF

module. Note that at each iteration, F_m is taken from the original known annotation of the member protein, i.e., F_m^0, so that successive updates of the group functions from F_g^i can be performed on protein's originally known annotation, F_m^0.

3.3 Results

3.3.1 Validation of the CRF model

For generation of dataset for the validation of the CRF pipeline, we clustered a protein-protein interaction (PPI) network of 6,124 human proteins that are involved in 1,12,895 interactions and selected 16 clusters that had at least 50 member proteins. The PPI network was extracted from the STRING database[96] with high confidence physical association score (>700). Clustering was done using the affinity propagation based clustering [151] described in Methods. For each of these selected clusters we tested whether the CRF with different combination of features (used in Eqns. 3.3-3.4) can correctly predict the GO terms of proteins in the network using the GO term annotation of neighbouring proteins. This is to test if the CRF itself is correctly implemented and if the features are useful for prediction. For all validation results shown in this section, a slimmed GO vocabulary of 303 GO terms is used, as described in Methods. In the PPI network clusters, 10% of the proteins were chosen as prediction targets and their annotations were removed.

4-fold cross validation result for 6 selected clusters out of 14 is shown in Fig. 3.3A-C (See supplemental Fig B.1 for the selection of these 6 clusters). Here we tested 3 different levels of feature combinations along with 2 different prior assignment of GO terms in the CRF model. "2-features" in Fig. 3.3 refer to the first and second terms in Eq.

3.4 in Methods, and "4-features" use all four terms in Eq. 3.4. The "6-features" CRF modules use all 6 term in Eq. 3.3 and Eq. 3.4. The two prior GO term assignments used are: *RandPrior* which is assigned based on frequency of GO terms in the training set, and *PFPPrior* which is taken from the GO prediction by the sequence based function prediction algorithm PFP previously developed in our group [12,13]. Supplemental Fig. B.2 shows the same results for all 14 clusters we selected above from the human PPI. Overall, the 6-feature combination shown in Eq. 3.3-3.4 with RandPrior and two specified cut-offs for the $P(y_i|y_j)$ score and *funsim(i,j)*(0.25 and 0.4, respectively) outperforms the other feature combinations we applied. The highest F-score achieved through CRF module was 0.7975 (precision 0.7957 and recall 0.7993) by the C8 cluster with 6-features-RandPrior and the cut-offs mentioned above. The subsequent results in this paper uses this best feature combination in the CRF module.

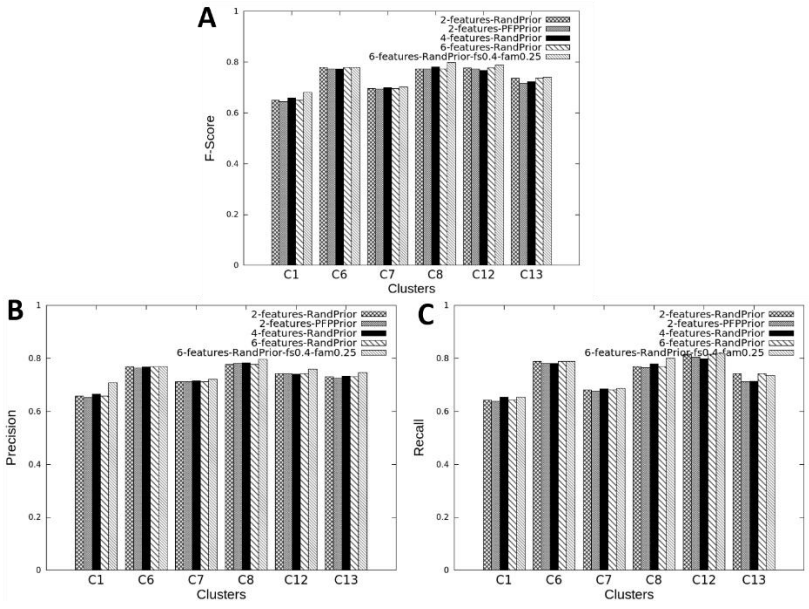

Figure 3.3 F- score on the GO prediction by CRF model

Next, we computed the average F-score accuracy over 4-fold cross validation computed at individual GO term level. Fig. 3.4(A-F) reports the result with CRF (Δ) for the six selected clusters as in Fig. 3.3 in comparison with a naïve prediction based simply on the frequency of GO terms in the group (● in the plot). The x-axis in the plot is the fraction of GO term occurrence in the training set, and y-axis is the average cross-validation F-score for that GO term. For all six selected clusters in Fig. 3.4A-F, CRF (Δ) showed a strong ability to make a correct GO assignment when GO terms are not common in the group (left half of the plots), where the frequency-based prediction breaks down.

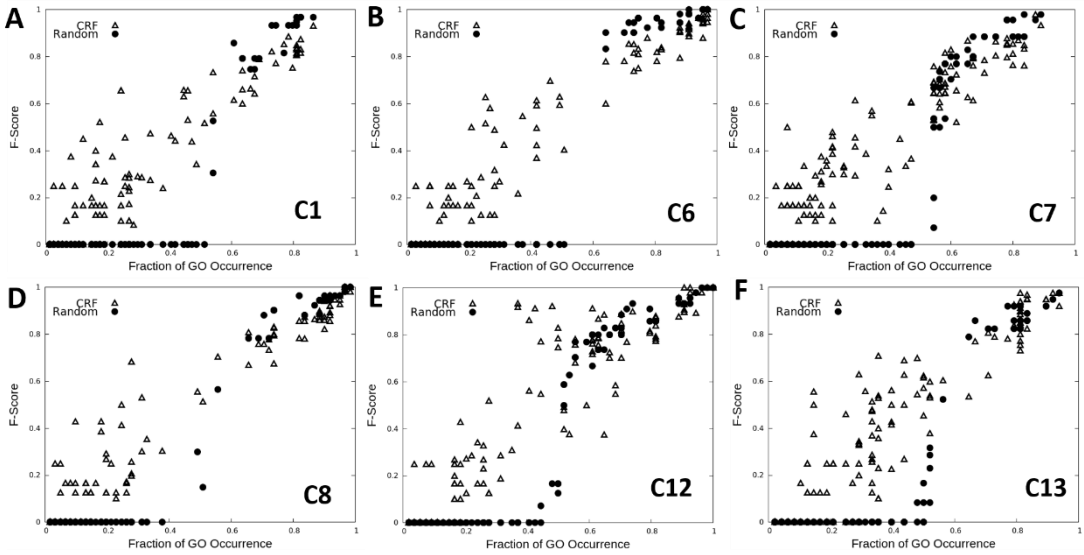

Figure 3.4 Per-GO term f-score of CRF

Reported is the average F-score of a four-fold cross validation (Δ) for 6 clusters in Human PPI network in comparison with a naïve prediction based simply on the frequency of GO terms in the group (● in the plot). Δ, CRF-based annotation; ● annotation based on frequency of GO terms.

3.3.2 Validation of the GFP pipeline

As dataset for the validation of the entire GFP pipeline (Fig. 3.1), we select 10 group of genes found in a SNP-targeted Genome-wide association studies (GWAS) studies as set of proteins involved in the Rheumatoid Arthritis disease [149]. Starting with a list of SNPs found to be associated with disease in GWAS, this study devises functionally important KEGG pathways through the identifications of SNP-targeted gene groups within these pathways.

Table 3.1 GFP validation dataset and network size

[a]number of nodes in the direct PPI neighbourhood of the gene groups; [b]LTM: Leukocyte transendothelial migration

DataSet	#genes	#netNodes[a]	#PPI	#Phylo	#KEGG	#GO	#GE	#SNF
ALLOGTAFT	8	37	189	0	10	17	37	220
APOPTOSIS	11	155	1877	0	33	145	159	2074
CANCER	32	1159	13859	0	3295	3141	5224	23907
CHEMOKINE	26	1013	23430	15	7613	3584	3703	33914
JAKSTAT	15	403	4577	0	782	833	847	5817
LTMb	17	757	9254	0	1811	1589	2184	13715
MAPK	20	715	8717	0	1634	1533	2243	12019
NEUROTROPHIN	20	779	10126	0	1754	1736	2391	14950
TCELL	16	595	7666	0	1579	1210	1660	11240
TOLL	13	611	7310	0	914	1286	1580	10405

For each of these group of genes (named according to the KEGG pathway they have shown to be involved in [149]), we first map them to different protein association networks and extract the portion of each component network consisting of the genes in the

group and their direct neighbors and then use SNF [150] platform to integrate the component network. Table 3.1 shows the data size for each of the groups and their associated networks.

After the integrated network (size shown in the last column of Table 3.1) is built for each of the 10 gene groups, we run the iterative GFP pipeline shown in Fig. 3.1 on these genes until convergence. Result of this GFP pipeline validation is shown for MAPK dataset in Fig. 3.5 (A-C). GFP was run on 713 proteins including the 20 target proteins in the integrated network with 12,019 interactions, and group function of the target proteins were predicted using the GO enrichment analysis performed on the predicted GO terms of the proteins after completion of each GFP iteration. The GFP pipeline was run until either the predicted enriched GO terms of the protein group from iteration *i* had sufficient change from iteration *i*-1 and *i*-2 or the number of iterations reached 10. To examine the robustness of the GFP pipeline's prediction, an increasing fraction of the GO terms annotating the 20 proteins in the gene group were removed (shown in x-axis), and the accuracy in terms of F-score, precision and recall of the prediction was computed (Fig. 3.5, A-C). The last iteration is shown separately (Δ in the plots), which in this case co-indices with the 5^{th} iteration (● in the plots). In comparison with the reference, a set of enriched GO terms after GO term removal from the existing partial annotation of the dataset (dotted line, denoted as RA-MAPK-ENRICH), GFP showed robust accuracy even after more than 50% of GO terms were removed. In contrast, the reference GO enrichment analysis quickly loses correct annotations and cannot infer the group function as GO terms are removed from proteins. Notably, recall performance shown in Fig. 3.5A grows significantly better with successive iterations. Precision shown in Fig. 3.5B is the opposite,

which is intuitive as GFP mostly adds GO terms denoting group function of the protein with successive iterations (see Fig. 3.2 and Methods for the procedure for updating individual protein's function derived from their group function). Note that since along the x-axis we are essentially removing GO terms from the existing partial annotation of the 20 proteins in the group (i.e., from the annotation at 0.0 x-axis for the baseline RA-MAPK-ENRICH), precision computed for the baseline does not drop until we remove 100% of the annotations. Overall, F-score for GFP showed significant improvement over the baseline for all x-axis points after 50% of the annotations were removed with a high recall of 0.8387 at x-axis = 0.5 compared to the baseline recall of 0.3226.

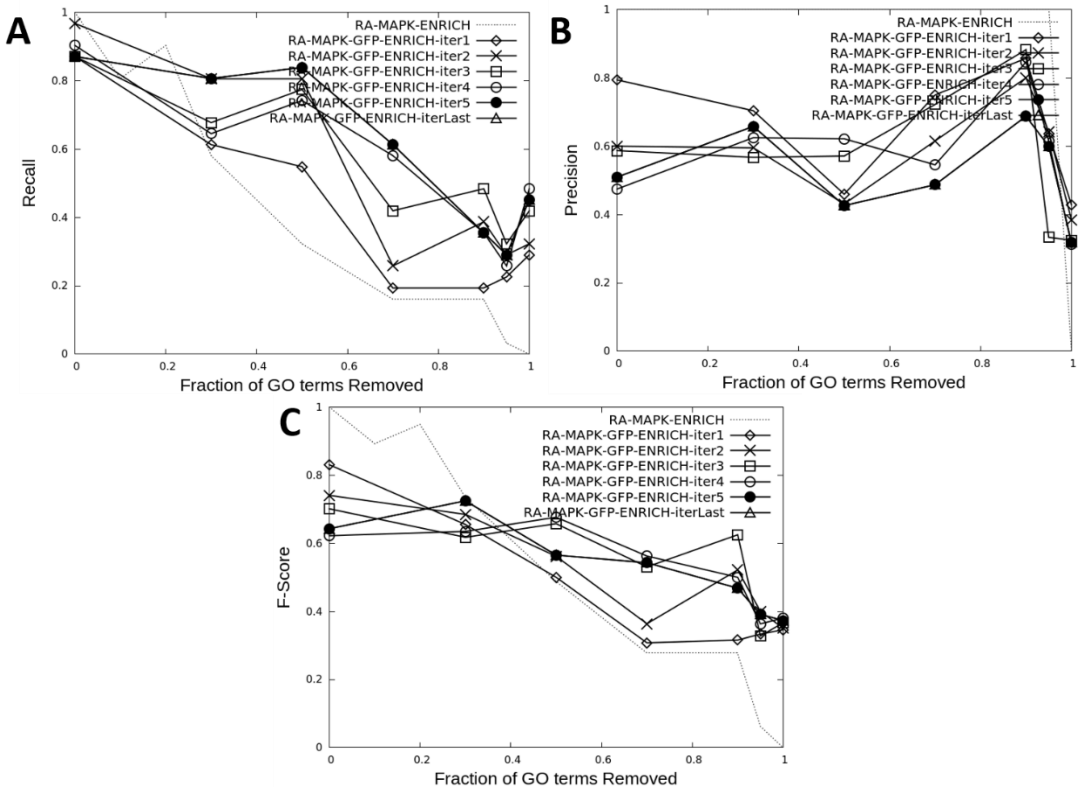

Figure 3.5 Group function prediction with GO-removal simulation

F-score of prediction was reported after removing a fraction of GO terms from a group of 20 proteins in the MAPK signaling pathway

.

Next, we ran another set of validation pipeline with a different method of removing annotations from the existing partial annotations of the protein groups. Instead of removing an increasing fraction of GO terms from the existing annotation, we remove entire GO annotations for an increasing fraction of proteins. The intuition behind this second way of validation is to remove any bias due to removal of individual GO terms from the protein group's annotation as done in Fig. 3.5, and to create a more realistic simulation of under-annotated datasets. The result is shown in Fig. 3.6A-C. Overall, the conclusion remains the same as the latter result. However, the baseline model has slightly higher accuracy than Fig. 3.5, since after removal of a protein P^a's annotation a certain true GO^i may still exist in another protein P^b, hence still retaining the precision and recall for the baseline. Nevertheless, GFP achieves recall as high as 0.8064 at 70% protein's annotation removal, compared to baseline recall of 0.4194.

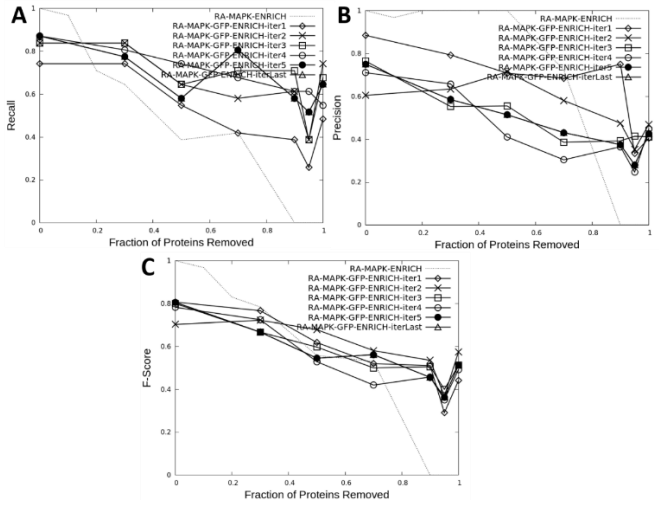

Figure 3.6 Group function prediction with protein-removal simulation

We ran similar tests for the rest of the 10 datasets in Table 3.1, and confirmed that the CRF-based GFP model is capable of robustly predicting correct GO terms for proteins and protein groups even when a substantial amount of GO annotations are missing. Results (F-score and Recall) for rest of the dataset is shown in Supplemental Figure B.3-B.4 and B.5-B.6 for the GO removal and protein removal simulations, respectively.

3.3.3 GFP parameter tuning

At the last step of the GFP model (step 4 of Fig. 3.1), we update the GO annotations of the individual proteins according to their cluster/group function predicted by the CRF module. For this procedure (Fig. 3.2), the similarity score cut-off SS represents how similar a new group function needs to be to a member protein's function in order to be added to the protein's annotation list. Here we show how we tuned this parameter separately for GO removal and protein removal simulations described in Fig. 3.5-3.6 for the MAPK dataset. We ran similar simulations with three different SS cut-offs, i.e., 0.3, 0.5, and 0.7 and computed recall, precision and F-score for the MAPK dataset. Figure 3.7-3.8 shows the result separately for two different schemes we took for gradual removal of annotations. Based on these results, we chose to use a SS cut-off of 0.3 and 0.7 for the GO removal results shown in Fig. 3.5 and the protein removal result shown in Fig. 3.6, respectively.

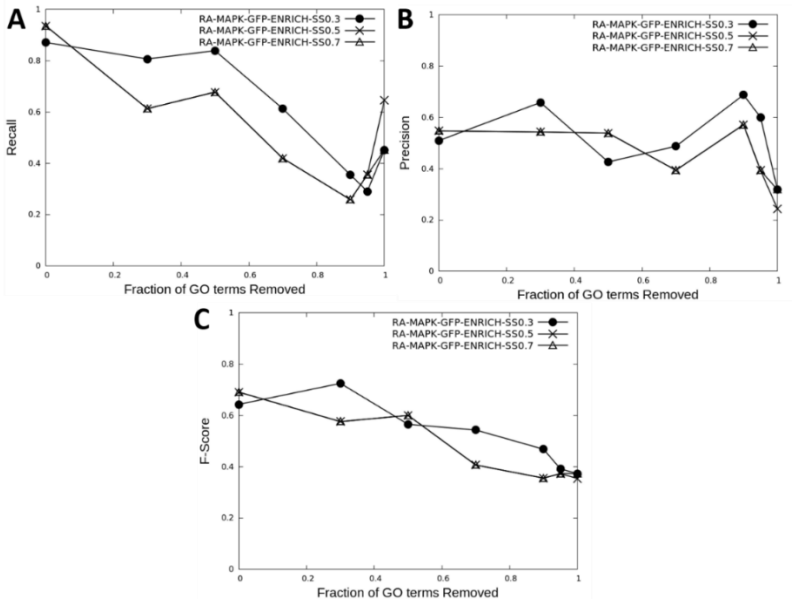

Figure 3.7 SS parameter tuning for GO removal

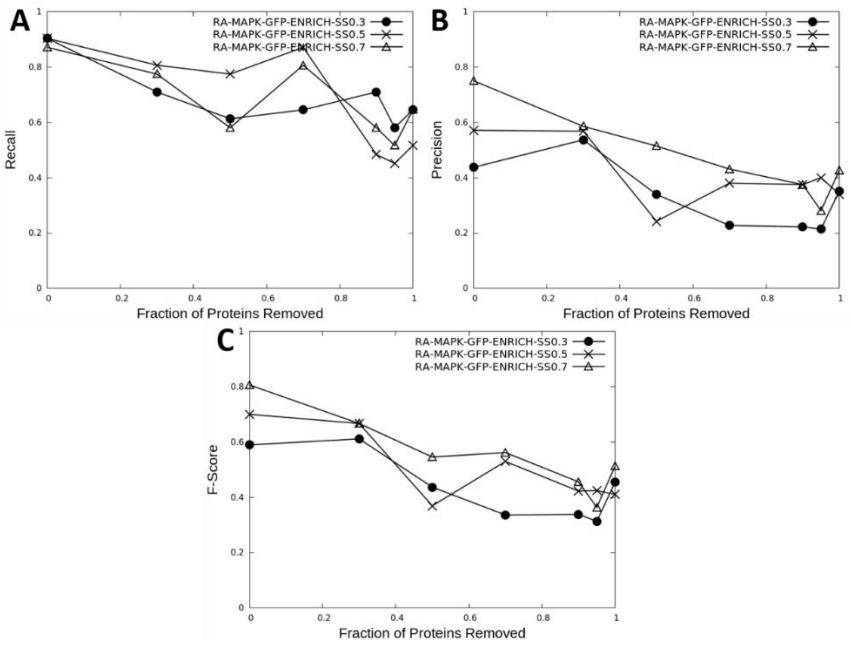

Figure 3.8 SS parameter tuning for protein removal.

Group function prediction to a group of 20 proteins in the MAPK signalling pathway for different SS cutoffs. F-score of prediction was reported after removing entire GO annotations of a fraction of proteins in the group.

CHAPTER 4. UPDATE OF AFP METHODS & CAFA CHALLENGE

4.1 Background

An essential task in bioinformatics is to propose and develop new tools and new ideas. However, to support the biology community, it is equally important to maintain and update previously-developed software tools so that users can continue using them. For a prediction method, it is important that the prediction accuracy be improved over time so that it can keep pace with other existing methods of the same type. For the advancement of such computational techniques it is very important that there are community wide efforts for objective evaluation of prediction accuracy. Community-wide prediction assessments have become popular in several computational prediction areas. Such experiments include CASP (Critical Assessment of techniques for Structure Prediction) [160] CAPRI (Critical Assessment of PRediction of Interactions) [161], and CAGI (Critical Assessment of Genome Interpretation) (http://cagi2010.org/). For the field of AFP, some experiments have been held in the past, which include MouseFunc 2006 (http://hugheslab.med.utoronto.ca/supplementary-data/mouseFunc_I/), ISMB (Intelligent Systems in Molecular Biology) AFP SIG (Special Interest Group) 2005 [162], the 2006 AFP meeting [163], and also the function prediction category in CASP6 [164] and CASP7 [165]. As a part of recently concluded ISMB conference 2011, CAFA (Critical Assessment of Function Prediction) experiment was conducted to gauge the

Gene Ontology (GO) [166] prediction accuracy of various AFP methods (http://biofunctionprediction.org/).

The last part of my research [51-53] copes with the AFP problem in three aspects: A. database update and improvement of methods previously developed in our group-PFP[12,13] and ESG [14], B. development of a web-server for the methods, and C. participation in CAFA[54] and benchmarking the performances. Along the same line of work, we develop two ensemble methods that combine GO predictions from multiple AFP models.

4.2 PFP/ESG servers and GO visualization tools

Here we developed web servers for our two function prediction algorithms, Protein Function Prediction (PFP) and Extended Similarity Group (ESG). As described in Methods 2.3.1.1, PFP predicts Gene Ontology (GO) terms for a query protein based on sequence information [12,13]. PFP extends traditional PSI-BLAST [6] search by extracting and scoring GO annotations from distantly similar sequences and by applying contextual associations of GO terms observed in the annotation database to the scoring scheme. PFP was ranked the best in the function prediction category in the Critical Assessment of Techniques for Protein Structure Prediction (CASP) [167].

As described in detail in Methods 2.3.1.2, ESG performs iterative sequence database searches and assigns a probability score to each GO term based on its relative similarity scores to the multiple-level neighbors in a protein similarity graph [14]. ESG was shown to outperform conventional methods in a thorough benchmark study. In the large-scale community based critical assessment of protein function annotation (CAFA) exper-

iment, ESG was ranked 4th in predicting Molecular Function (MF) GO terms among 54 participating groups [54]. Thus, both PFP and ESG have been rigorously benchmarked both in the original papers and in objective assessments by the community.

PFP and ESG were designed to achieve complementary goals: PFP is for large prediction coverage by retrieving annotations widely including weakly similar sequences. On the other hand, ESG is for improving specificity by accumulating contribution of consistently predicted GO terms in an iterative search. Here, we introduce a publicly available webserver for these two function prediction methods. The interactive webserver of PFP and ESG reported in this scope [52] is developed to assist in the sequence-based function prediction and to enhance the understanding of predicted functions by an effective visualization of the predictions in a hierarchical GO topology.

4.2.1 Results

4.2.1.1 Input & output visualization of the webserver

PFP and ESG accept query inputs of FASTA formatted protein sequences. Users may submit sequences separated by line breaks in the text box titled “Enter Query Sequence(s)” or upload a FASTA file containing multiple sequences. To view a sample of the format, users may click on “Load Sample” to fill the field with an example sequence. Selecting “Clear” will remove all inputs sequences including uploaded files. Currently, up to 100 sequences may be annotated in order to avoid blocking the job queue, particularly for ESG.

Particularly for ESG, there are two more parameters that need to be entered for ESG algorithm: “Number of hits” and “Number of stages”. “Number of hits” indicates number of PSI-BLAST hits to be considered at each level of ESG. Default value of this parameter is set to 10 in our web server. “Number of stages” indicates the levels of neighborhood around the query protein that will be considered by ESG. Default value for this parameter is chosen as 2. User can change the value to any other numbers but currently we are limiting the “Number of hits” to be smaller than 100 due to computational constraints. We recommend not changing the “Number of stages” parameter to a larger value as the computational time will suffer exponentially and we did not observe an improvement during benchmark in the original paper [14]. As for the “Number of hits” argument, we would encourage the user to test different settings and increase the value. For example, if user increases the default value from 10 to 50, roughly it takes 5 times more computational time (2 stage setting) but an improvement in the accuracy.

Both PFP and ESG algorithms predict GO terms for a given protein sequence. ESG outputs a score that ranges from (0,1). Predicted GO terms are listed on the result page (Fig. 4.1, left panel). Predictions are classified into four confidence levels: very high, high, moderate, and the rest. In addition, a XML file is provided that summarizes the prediction. Moreover, predicted GO terms are visualized as discussed below. Submitted jobs are tracked and kept in a MySQL database so that the user can retrieve the results later.

4.2.1.2 Tracing origin of the predicted GO terms

The servers provide sequence IDs indicating the source of each predicted GO term. We implemented this functionality because it is a common question by users how a GO term is predicted by the servers (Fig. 4.1, right panel).

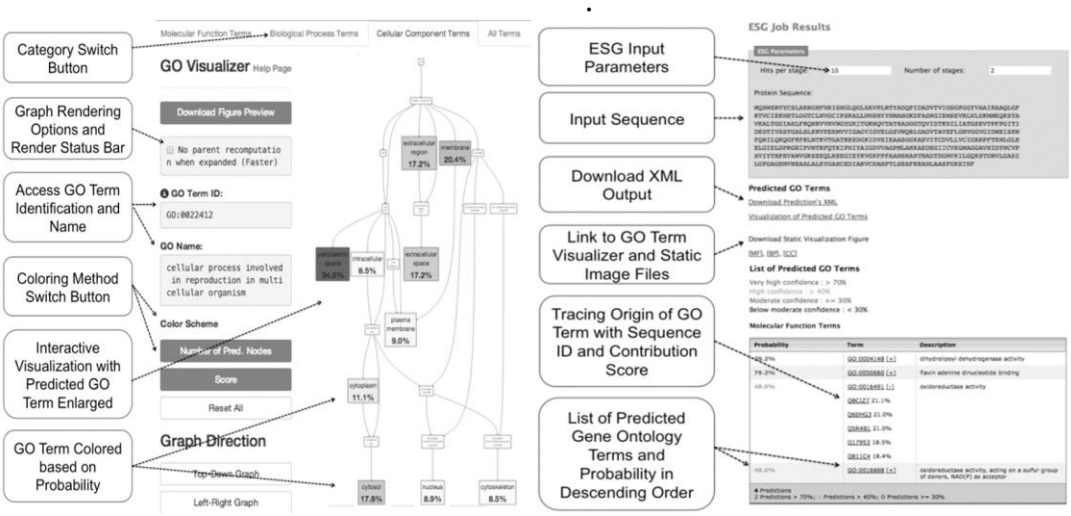

Figure 4.1 Output page of ESG & GO visualization

A result page of PFP is essentially the same. Below the input sequence, links are provided for downloading the prediction result in an XML file and for visualizing predicted GO terms in GO hierarchy.

4.2.1.3 GO term visualization

The GO term visualizer intuitively shows predicted GO terms in the GO hierarchy (Fig. 4.1, right panel). A visualized GO graph can be zoomed in/out or further expanded to see sub-nodes of a branch. GO terms are colored based on their assigned probability. GO terms can be also colored based on the number of child nodes of them that are predicted. In addition, visualization in cytoscape allows 3 modes of GO hierarchy visualiza-

tion (Tree, Radial, Circle) and enables users to select and drag around groups of GO terms.

4.3 Performance evaluation of PFP/ESG on CAFA'2011 experiment

In the CAFA experiment in 2011, in total of 48,298 target protein sequences were released for prediction, which consist of seven eukaryotic genomes, eleven prokaryotic genomes, and a supplementary set of additional sequences. The participating predictor groups were expected to submit GO annotations for these targets in Biological Process (BP) and Molecular Function (MF) domains. Out of these set, the organizers selected 436 targets in BP domain and 366 targets in MF domain that newly obtained experimental annotation in the SWISS-PROT database from January to May 2011, which is after the closing of the submission,. Submitted predictions were evaluated using different prediction accuracy measures described in Methods.

We have submitted predictions using two methods developed in our group, the Protein Function Prediction (PFP) method [12,13] or the Extended Similarity Group (ESG) method [14]. PFP and ESG use PSI-BLAST sequence database search results, from which function information is extracted extensively, even from weakly similar sequences. In this article, we analyze the prediction performance of these two methods in comparison with BLAST, the Prior method, and GOtcha [15], whose predictions are provided by the CAFA organizers. Prediction performance evaluation employed four metrics used by the organizers; the threshold method, the top N method, the weighted threshold method, and the semantic similarity method (see Methods). Besides evaluating original predictions by PFP and ESG submitted to CAFA, we further investigated the followings

to have a better understanding of their performance: 1) For PFP predictions, we reranked predicted GO terms using a different score from the originally used score and compared the performances; 2) We combined PFP and ESG predictions with those from the Prior method that simply ranks GO term by the background frequency in a database; 3) We evaluated prediction accuracies of each method separately for different functional categories; and 4) We examined successful and unsuccessful predictions by PFP and ESG in comparison with BLAST. The in-depth analysis discussed here will complement the overall assessment of by the CAFA organizers that will be published elsewhere. Since PFP and ESG are based on sequence database search results, our analyses are not only useful for PFP and ESG users but will also shed light on the relationship of the sequence similarity space and functions that can be inferred from the sequences.

4.3.1 Methods

In this section we briefly describe the AFP methods that are compared in this study. Predictions in the MF and the BP domain were evaluated by comparing them with annotations with experimental evidences (i.e. non Inferred Electronic Annotations; non-IEA) in the Uni-Prot database. For each target, predictions were restricted to 1000 highest score predictions with the score ranging between 0 and 1.

4.3.1.1 The Prior method

In the prior method, each GO term is assigned the frequency of its occurrence in SWISS-PROT (January 2011 version) including a pseudo count of 1. For a given target

sequence, top 1000 GO terms with highest frequencies were selected as predictions. Thus, all target sequences have the same set of predictions by this method. The prior predictions for each target were provided by the organizers.

We have also combined the prior predictions with predictions by PFP and ESG. These predictions are called the enriched PFP/ESG or PFP/ESG + Prior. In PFP + Prior, we added GO terms to PFP predictions that are not predicted by PFP (the expected accuracy was used for the PFP score). The score (i.e. frequency) for GO terms imported from the prior method was rescaled by considering maximum and minimum scores of PFP predictions for that target. GO terms originally predicted by PFP and ones imported from the prior method are sorted by the score. Similar to the PFP + Prior, ESG + Prior also combined the original ESG predictions and GO terms from the prior method that are not predicted by ESG. Since both the ESG score and the frequency in the prior method range from 0 to 1, GO terms from the two methods were sorted by the score without rescaling.

4.3.1.2 BLAST

BLAST search [5] with default parameters was performed for each target sequence. Score for a particular annotation term was the maximum sequence identity with the hit annotated with that term. Predictions by BLAST were provided by the organizers.

4.3.1.3 Gotcha

GOtcha [15] incorporates the hierarchical structure of GO vocabulary with the idea of homology based annotation transfer to achieve improved coverage. It uses BLAST [5] to search similar sequence hits and assigns a score, -log(E-value), to each GO annotation

of the sequence hits and its less specific ancestors in the GO hierarchy. The scores assigned to each GO node from all the sequence hits are summed and then normalized using the score of the root of either MF or BP ontology. The normalized score thus obtained is referred as I-score, which was used for selecting target annotations. Predictions by Gotcha were provided by the organizers.

4.3.1.4 Assessment methods for prediction accuracy

In CAFA, predictions were evaluated using four different methods. The threshold and the top N methods count exact match of predicted and the actual annotations, punishing any predictions that are more or less specific than the actual annotations. On the other hand, the weighted threshold and the semantic similarity take into account the information content of terms being matched on the GO hierarchy. Please refer to the organizers' paper in the same journal issue for more details. We have used Gene Ontology version October 2011 for obtaining ancestors for each GO term.

4.3.1.4.1 The Threshold method

For each prediction method we use thresholds ranging from 0.01 to 1.0 to calculate the average precision, recall, and specificity for all targets. For each target if a particular prediction has a score above the threshold, the predicted GO term is propagated to the root of the ontology. The performances are analyzed in terms of precision-recall curve and the receiver operator characteristic (ROC). For the threshold method, when using PFP raw scores that are not scaled between 0 and 1, we selected 1 to 1000 GO term pre-

dictions by the increments of 5 and compute average precision, recall and specificity for all targets.

4.3.1.4.2 TopN

The top N highest scoring predictions for a prediction method are taken into consideration with N varying from 1 to 20. For all the predictions within top N, parental annotations until the root of the ontology are included. All predicted annotations with a tie score at a particular ranking are considered for the cutoff.

4.3.1.4.3 Weighted threshold

As shown in Equation 4.1, frequency of a GO term *c* in the database is computed as the number of gene products annotated by term *c* and its children *h* in the GO hierarchy.

$$freq(c) = annot(c) + \sum_{h \in child(c)} freq(h) \qquad \text{(Eqn. 4.1)}$$

where *annot(c)* is the number of gene products annotated by non IEA evidence codes in September 2011 version of SWISS-PROT database. Probability of a particular term *c*, *p(c)=freq(c)/freq(root)*, is computed as the ratio of the frequency of *c* against the frequency of the root term of the MF or BP ontology. Information content of term *c* is given by $IC(c) = -\log_{10}(p(c))$. Using this information content, weighted precision is calculated as the sum of information content of the terms in the true positive set divided by the sum of information content of the terms in the true and false positive sets. Similarly, weighted recall is computed as the sum of information content of the terms in the true positive set

divided by the sum of information content of the terms in the true positive and false negative sets. As with the previous methods, if a particular prediction is above the given threshold, then its ancestors till the root of the ontology are included in the prediction set.

4.3.1.4.4 Semantic similarity

Semantic similarity for a pair of GO terms is given by the maximum information content of a shared ancestor of both terms and it is averaged between all pairs of true and predicted terms to obtain the semantic similarity for a target. We calculate the semantic precision for a target protein as the average of the difference between the IC of a predicted term and the maximum of the IC of common parental terms between the predicted term and any correct term. Similarly, semantic recall is calculated for a target as the average of the difference between the IC of a true term and the maximum of the IC of common parental terms between the true term and any predicted term. Here the information content values are based on the Prior probabilities for each term provided by the CAFA organizers. The average semantic similarity, semantic precision and semantic recall are computed across all targets at each threshold varying from 0.01 to 1.0.

4.3.2 Results

4.3.2.1 PFP with raw scores

In the CAFA experiment we submitted PFP predictions sorted by the confidence score. In this section, we rank predicted GO terms by PFP according to the raw score and see how its performance compares with the confidence score and the other methods.

From ranked list of PFP predictions by their raw score, precision, recall, and specificity are calculated at each of the top N predictions taken with an interval of 5.

Figure 4.2 shows the precision-recall curve and the ROC of PFP with raw score compared with the other methods. For the BP domain, we observe that PFP with raw score (PFP_RAW in the plots) has slightly higher precision for a given recall value than PFP predictions ranked by the confidence score (PFP). PFP with raw score has clearly better performance than PFP with confidence score in the ROC curve (Fig. 4.2B), particularly at lower false positive range (x-axis). The similar behavior of PFP raw score is observed for predictions in the MF domain (Figs. 4.2C & 4.2D). These results indicate that the confidence score of PFP, which is computed in two steps from the raw score via the p-score distribution (see Methods), was not very successful in ranking predicted GO terms especially at top ranks (lower false positive regions). Thus, derivation of the confidence score needs to be reexamined and probably revised.

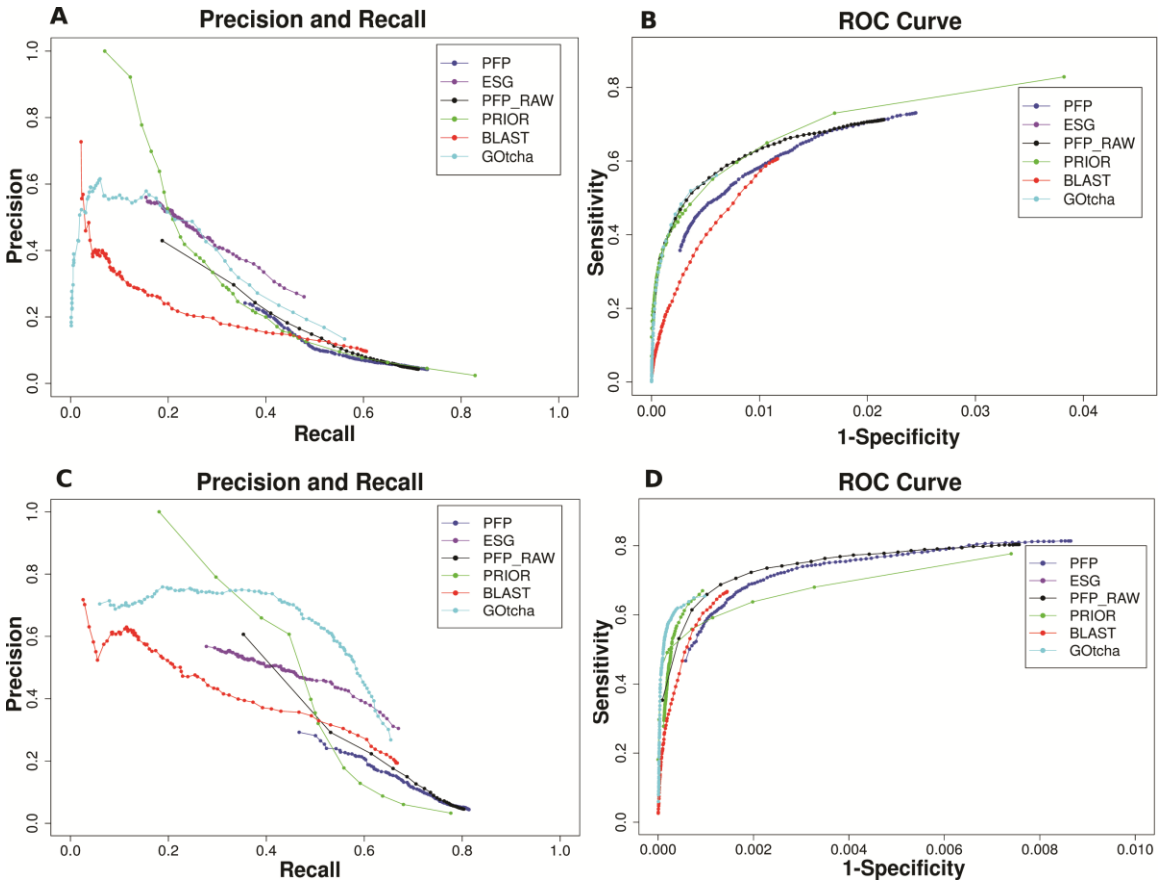

Figure 4.2 Performance comparison of AFP methods

Performance of PFP(confidence score), PFP prediction sorted by the raw score (PFP_RAW), ESG, PRIOR, BLAST, and Gotcha. ***A****, Precision – Recall plot for the BP domain.* ***B****, ROC for the BP domain.* ***C****, Precision – Recall plot for the MF domain.* ***D****, ROC for the MF domain.*

4.3.2.2 PFP and ESG with enriched priors

Next, we combined the PFP and ESG predictions with the prior predictions (PFP + Prior, ESG + Prior) to see if PFP/ESG predictions were missing obvious GO terms (Fig. 4.3). We show the performance of the methods is evaluated with the top N method, where N ranges from 1 to 20.

ESG with enriched priors (ESG + Prior) shows the best performance among all the methods in BP domain when evaluate by the precision-recall plot (Fig. 4.3A). The improvement by ESG + Prior over ESG is also observed in terms of ROC (Fig. 4.3B). ESG + Prior also performed better than ESG in the MF domain (Figs. 4.3C & 4.3D). ESG tends to predict fewer GO terms than even BLAST since its algorithm essentially selects terms that are consistently identified by iterative searches. The results in Figure 4.3 indicate that obvious GO terms in Prior were not included in ESG predictions. Since some GO terms may be lost in the iterative process of the ESG algorithm, the scoring scheme needs to have a close inspection. On the other hands, adding Prior prediction to PFP did not show any improvement over PFP, which indicates that PFP's predictions already include correct terms from Prior.

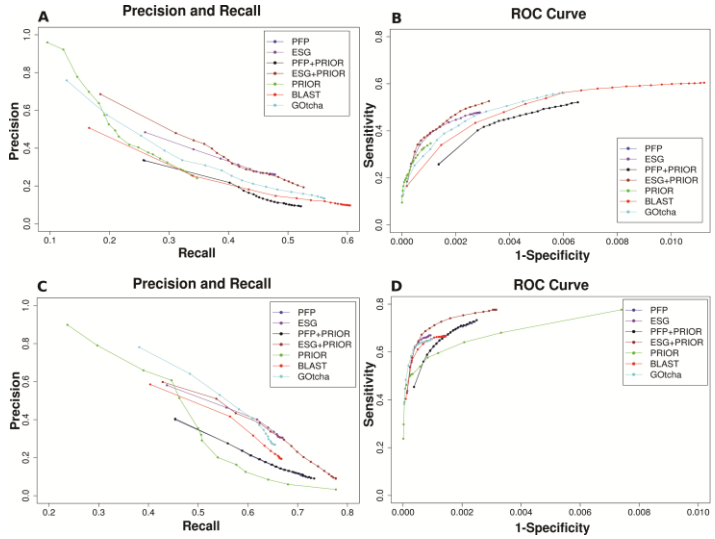

Figure 4.3 Performance comparison of AFP methods with enriched priors

A, *Precision – Recall plot for the BP domain;*

B, *ROC for the BP domain;*

C, *Precision – Recall plot for the MF domain;*

D, *ROC for the MF domain.*

4.3.2.3 PFP and ESG with semantic similarity

In Figure 4.4 the performance of the methods are evaluated in terms of the semantic similarity. The average of the semantic similarity between all pairs of true and predicted GO terms is for each method is plotted relative to thresholds in Figure 4.4A and 4.4C for the BP and MF domain, respectively. It is shown that ESG's performance is significantly better than the other methods for both BP and MF targets. PFP performance is average among all the teams in this measure. On the other hand, PFP stands out in the semantic precision and recall plots (Figs. 4.4B & 4.4D). ESG comes second in the BP domain (Fig. 4.4B) but shows worst performance among all in the prediction of MF terms (Fig. 4.4D).

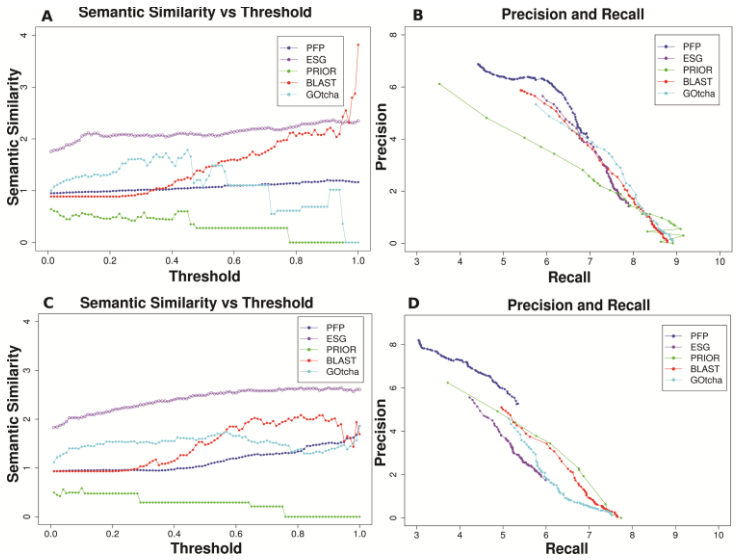

Figure 4.4 Performance comparison of AFP methods with semantic similarity

***A**, Semantic similarity relative to the score threshold. Predictions in the BP domain are evaluated;*

***B**, Semantic precision vs semantic recall for the BP domain;*

***C**, Semantic similarity relative to the score threshold in the MF domain;*

***D**, semantic precision vs semantic recall for the MF domain.*

4.3.2.4 Examples of successful and failure PFP/ESG predictions

Finally, we discuss the prediction examples where PFP, ESG, and BLAST succeeded at different levels that provide insights into the advantages and shortcomings of our methods. The first example is T06450, *Escherichia coli* protein trbA, which is annotated with *GO:0042026 protein refolding* as per the CAFA target annotations. BLAST search finds only one sequence hit O26024 that does not have any non-IEA annotation in the database resulting in no predictions. As for ESG, some of the correct low resolution annotations are extracted from a sequence hit Q9UZ03 retrieved in the first iteration of PSI-BLAST search with very large E-value (above 1) and its second level hits, including Q8A608, Q64PS6, Q5L9I8. These predicted annotations are parental terms of actual annotations: *GO:0008152 metabolic process* is a parental term of *GO:0042026 protein refolding*, and *GO:0008652 amino acid biosynthetic process* shares a common ancestor *GO:0044237 cellular metabolic process* with the target annotation *GO:0042026 protein refolding*. PFP was able to predict some low resolution parental terms of the correct annotation such as *GO:0046483 cellular macromolecule metabolic process* and *GO:0044260 cellular protein metabolic process*, with significantly high confidence scores of 0.81 and 0.99. Both these terms are not part of annotations of any of the PSI-BLAST hit but received partial scores by considering co-occurrence of GO terms.

The second example, T06299, rutE from *E. coli*, is annotated by two leaf terms *GO:0019740 nitrogen utilization* and *GO:0019860 uracil metabolic process*. For this target BLAST again does not predict anything as there are no search hits with non IEA annotations. Using IEA annotation of highly similar PSI-BLAST hits, ESG predicted

GO:0055114 oxidation-reduction process, which shares a shallow common ancestor *GO:0008152 metabolic process* with a target term *GO:0006212 uracil catabolic process.* Similar to the previous example, PFP again predicted low resolution annotations *GO:0006139 nucleobase, nucleoside, nucleotide and nucleic acid metabolism* and *GO:0046131 pyrimidine ribonucleoside metabolism* thereby showing higher sensitivity when no close homologs are available for annotation transfer.

The third target T05345 is sensor protein CpxA from *E. coli* with leaf annotation *GO:0046777 protein amino acid autophosphorylation*. ESG predicted *GO:0018106 peptidyl-histidine phosphorylation,* which shares an immediate parent *GO:0006468 protein amino acid phosphorylation* with the target term *GO:0046777 protein amino acid autophosphorylation.* Also another term *GO:0016310 phosphorylation,* which is an ancestor of the target annotation is predicted by ESG with a high score of 0.93. PFP correctly predicts the ancestors of the target term, *GO:0016310 phosphorylation, GO:0006464 protein modification and GO:0006468 protein amino acid phosphorylation* with very high scores. BLAST predicts the target term and its ancestors with lower scores along with a number of unrelated predictions with high scores. Overall all the methods are able to predict the target term or its close ancestors, but the total number of terms predicted by BLAST (193 terms) and PFP (134 terms) are significantly higher than ESG (7 terms), resulting into more precise predictions by ESG.

The last example, T18799, *Homo sapiens* Ribonuclease H2 subunit B, is annotated by a leaf term *GO:0006401 RNA catabolic process* which has been accurately predicted by BLAST. BLAST obtains this correct annotation from sequence hits such as Q5TBB1, Q5XI96, Q3ZBI3, Q80ZV0, Q28GD9, and Q5HZP1. These sequences were

also found by ESG, however, due to use of an older database that do not have updated annotations for these sequences, no correct annotation was retrieved. There are some shared ancestors, e.g. *GO:0016070 RNA metabolic process, GO:0090304 nucleic acid metabolic process, GO:0044260 cellular macromolecule metabolic process* between the low scoring ESG prediction *GO:0006429 leucyl-tRNA aminoacylation* and the target annotation *GO:0006401 RNA catabolic process*. PFP was able to correctly predict low resolution terms, *GO:0044260 cellular macromolecule metabolism* and *GO:0016070 RNA metabolism*.

To summarize, the first and the second examples illustrate a situation where PFP predicts low resolution parental terms of actual annotations while BLAST can only find 1 or 0 terms. There are PFP's successful prediction which were found indirectly by using the GO term co-occurrence. In the second example, IEA annotations lead to correct predictions for ESG and PFP. The third example is the case that ESG made predictions with higher precision with smaller number of false positives than BLAST and ESG. The last example is that ESG missed to make correct prediction because the sequence database which was searched was not up-to-date.

4.4 PFP/ESG update for CAFA2 & novel ensemble approaches

In the second round of CAFA, CAFA2, for which an evaluation meeting was held as a SIG meeting at the ISMB conference in Boston in 2014, a total of 100,816 target protein sequences from 27 species were provided. These are more than double the amount of targets than in CAFA1 (48,298 targets in 18 species), which was held in 2011.

The participants could submit up to three models (variations in the prediction method) for a registered prediction method (group).

In this work, we report benchmark results and enhancements made as preparation for CAFA2 prior to participation. We first discuss the effect of an update of annotation databases that are used in our sequence-based function prediction methods, PFP and ESG. In the CAFA1 experiment, ESG was ranked 4th in MF among 54 participating groups [54]. We then examined whether considering prior distribution of GO terms in the UniProt sequence database [86] improved the accuracy. PFP and ESG using the updated databases performed significantly better than the same with older databases. We did not observe meaningful improvement by adding GO terms' prior probability.

Finally, we constructed two ensemble function prediction methods, CONS and FPM, that combine GO predictions from PFP [12,13], ESG [14], PSI-BLAST [6], PFAM [72], FFPRED [170], and HHblits [171]. Among the six individual methods, ESG with the updated database performed the best. One of the ensemble methods, CONS, performed the best while the other one, FPM, ranked in the middle when compared with the six individual methods. Successful and unsuccessful cases of ensemble methods are discussed.

4.4.1 Benchmark dataset

The benchmark consists of 2055 non-redundant protein sequences selected from the UniProt Reference Clusters (UniRef) database [86]. UniRef provides clustered sets of sequences from the Uniprot knowledgebase. We selected a cluster resolution of 50% sequence identity. Among these UniRef50 clusters, we selected the representative proteins

from clusters that satisfy the following two criteria: 1), each cluster representative should have at least 1500 proteins in its cluster and 2), the cluster representative protein should have a non-empty GO term annotation in the UniProt database.

4.4.2 Methods

4.4.2.1 FFPRED method

FFPred [172] predicts more than 440 possible GO terms for a query protein using support vector machines (SVMs) that use more than 200 features of the query. These features are spread among fourteen feature types. These types include twenty features describing amino acid composition; seven features describing the sequence itself; fifty features describing the phosphorylation and others [173]. The SVM-Light [174] package was used to create the SVM classifiers. For each GO term, an SVM classifier was trained by empirically determining the set of kernel parameters and features that performed best in a *k*-fold cross validation of the set of training proteins. The best features were determined on the level of the feature types, so that if the inclusion of the features in a feature type did not improve the SVM, all the features for that feature type were discarded.

4.4.2.2 HHBlits method

HHblits [171] takes a sequence or multiple sequence alignment as a query and produces a profile HMM from this input. Using the computed HMM, the program iteratively searches a database of profile HMMs, with similar HMMs used to update the query HMM. A pre-filter of discretized HMM profiles is used in order to dramatically speed up

the process. There are two pre-filtering steps when comparing the extended sequence profiles to those of the database. The first makes sure that the score of the largest un-gapped alignment between two profiles passes a threshold. Out of the remaining sequences, those with a Smith-Waterman alignment better than a threshold are used. The GO terms from the protein sequences in the final HMM are collected as the predictions of GO terms of the query.

4.4.2.3 Consensus method (CONS)

CONS is one of the ensemble methods we constructed that combines predicted GO terms for a target protein from the following six AFP methods, namely, PFP [12,13], ESG [14], PSI-BLAST [6], PFAM [72], FFPred [172], and HHblits [171]. PSI-BLAST was run up to three iterations and GO terms were taken from the top five hits. PFAM [57] is a database of HMMs of protein families and domains. A protein can be associated with more than one protein domain HMM. A query sequence was compared with HMMs in PFAM using the HMMER software suite [175] and GO terms were retrieved from hits equal to or below an E-value of 0.01 using the model2GO file associated with PFAM.

CONS combines GO term predictions from each of the individual methods and provides a consensus confidence score. The consensus confidence score for a GO term is essentially the weighted sum of scores of the GO term from individual methods. The score for GO term GO^i is defined as

$$CONS\_score(GO^i) = \frac{\sum_{m=1}^{6} w_m conf(GO_m^i)}{\max_{k=1}^{N}(CONS - score(GO^k))} \qquad \text{(Eqn. 4.2)}$$

where m is an index through each of the six individual methods, and N is the total number of unique GO terms for the target predicted by the six methods. The weights w_m reflect prior knowledge of the performances of individual methods m, which are the accuracies of the methods (F_{max} score). w_m for a target sequence was computed on the benchmark dataset after removing the target from the dataset.

4.4.2.4 Frequent Pattern Mining (FPM): an ensemble method

The Frequent Pattern Mining (FPM) is a widely-used data mining technique for finding frequently occurring patterns of items. Agrawal *et al.* [176] first introduced an *a priori* technique of mining all frequent item sets from a transactional database. Later, Tao *et al.* refined the technique for datasets where each item can have weights [177]. Here we used the flavour of the latter technique to construct an ensemble protein function prediction method from the underlying six individual AFP methods.

We describe the FPM method in the function prediction setting with a toy example. Let us consider GO term predictions from three AFP methods, Method A, B, C, for a certain target protein. Let us also assume that each method has a pre-computed Fmax accuracy score, accuracy(Method A) = 0.6, accuracy(Method B) = 0.7, and accuracy(Method C) = 0.5. We assume the three methods predict GO terms as follows:

- Method A: GO1: 0.5, GO2: 0.6, GO3: 0.4
- Method B: GO2: 0.7, GO3: 0.8, GO4: 0.4, GO5: 0.6
- Method C: GO2: 0.8, GO3: 0.9, GO5: 0.6

Here, GO1:0.5 under “Method A” denotes that Method A predicts GO1 with a confidence score 0.5.

First, we define two weights that we use throughout the FPM process. *weight(m_k)* is a weight given to each method m_k as follows:

$$weight(m_k) = \frac{\sum_{i=1}^{|m_k|} weight(GO_i)}{|m_k|} * accuracy(m_k) \quad \text{(Eqn. 4.3)}$$

$|m_k|$ is the number of GO terms predicted by the method m_k. accuracy(m_k) for a target sequence is computed on the benchmark dataset after removing the target from the dataset.

When the benchmark dataset has multiple target proteins, method weights can be different for each target. For the target in the above toy data,

$$weight(MethodA) = \frac{0.5 + 0.6 + 0.4}{3} \times 0.6 = 0.3$$
$$weight(MethodB) = \frac{0.7 + 0.8 + 0.4 + 0.6}{4} \times 0.7 = 0.44$$
$$weight(MethodC) = \frac{0.8 + 0.9 + 0.6}{3} \times 0.5 = 0.38$$

weight(GO_{set}) is a weight given to a set of GO terms with set size |set| as follows:

$$weight(GO_{set}) = \frac{\sum_{k=1}^{|S|} weight(m_k)}{\sum_{k=1}^{|M|} weight(m_k)} \quad \text{(Eqn. 4.4)}$$

Here M is the set of all methods and S is the set of methods that predicted GO_{set}. For the above toy example, M is 3 and S is 2 for GO2 (since 2 methods, *i.e.*, Method A and Method B, have GO2. GO2 is a GO_{set} of size, |set| = 1). Initially, FPM generates all possible GO_{set}s of |set| = 1 and computes the weights of each GO_{set} using Eqn 4.4. In the above toy example, the generated GO_{set}s are {GO1, GO2, GO3, GO4, GO5} and the weights are:

$$weight(GO1)$$
$$= \frac{weight(MethodA)}{weight(MethodA) + weight(MethodB) + weight(MethodC)}$$
$$= \frac{0.3}{0.3 + 0.44 + 0.38} = 0.27$$
$$weight(GO2) = \frac{weight(MethodA) + weight(MethodB) + weight(MethodC)}{weight(MethodA) + weight(MethodB) + weight(MethodC)} = \frac{1.12}{1.12} = 1.0$$
$$weight(GO3) = 1.0, weight(GO4) = 0.39, weight(GO5) = 0.73$$

Then FPM uses a pre-defined weight cut-off to select the GO_{set}s with weights higher than the cut-off and maintains a lexicographic ordering of this selected GO_{set}s, L, throughout the rest of the process. In the above toy example, for a weight cut-off 0.5, FPM selects L = {GO2, GO3, GO5}.

Now, the FPM algorithm runs iteratively starting from |set| = 2 and increases |set| by 1 at each iteration. At each iteration *i*, FPM creates a list, $GList_i$ of frequently-occurring GO_{set}s at the current iteration *i*. At iteration 1, $GList_1$ = L. In each iteration *i*, FPM generates GO_{set} where |set|=i by lexicographically extending each element in $GList_{i-1}$ with each element in set L. FPM then keeps the GO_{set}s that have $weight(GO_{set})$ above the weight cut-off and stores them in $GList_i$. Iterations continue until no new GO_{set} can be generated. We demonstrate the generation of $GList_i$ at each iteration for the above toy example.

- Iteration 1: Candidate GO_{set}: {GO1, GO2, GO3, GO4, GO5}, $GList_i$: {GO2, GO3, GO5}
- Iteration 2: Candidate GO_{set}: {GO2-GO3, GO2-GO5, GO3-GO5}, $GList_i$: {GO2-GO3, GO2-GO5, GO3-GO5}
- Iteration 3: Candidate GO_{set}: {GO2-GO3-GO5}, $GList_i$: {GO2-GO3-GO5}

At iteration *i*, weight(GO_{set}) with |set| = i is calculated using Equation 4.4. In the above list, the weight of GO_{set}, GO2-GO5 at iteration 2 is calculated as-

$$weight(GO2-GO5) = \frac{weight(MethodB)+weight(MethodC)}{weight(MethodA)+weight(MethodB)+weight(MethodC)} = \frac{0.44+0.38}{0.3+0.44+0.38} = 0.73$$

The final result (most frequently occurring GO_{set}) is chosen in two ways: FPM_maxLen chooses the maximum-length GO_{set} among all in $GList_i$ (for all i), and FPM_maxScoreLen chooses the maximum-length GO_{set} among the highest scoring GO_{set}s in all $GList_i$ (among all i). For each target in the benchmark data, the FPM algorithm runs once and generates the most frequently predicted GO terms for that target. We used 0.7 as the predefined weight cut-off.

4.4.2.5 Evaluation metric: The F_{max} score

The Fmax score is computed according to the evaluation strategy taken in CAFA1 [54]. For each target, given a true annotation set T and a predicted annotation set P_t from an AFP method above a certain GO confidence score threshold *t*, precision and recall is calculated as follows:

$$precision_t = \frac{TP}{TP+FP}$$

$$recall_t = \frac{TP}{TP+FN}$$

where $TP = T \cap P_t; FP = P_t \setminus T; FN = T \setminus P_t$. Then, at each confidence threshold *t*, average precision and recall is calculated across all targets. From these average values, F-

measure is calculated as the harmonic mean between precision and recall at each confidence threshold value. Then the maximum F-measure across all thresholds is taken as the Fmax score.

$$F\max = \max_{t}\left\{\frac{2 * precision_t * recall_t}{precision_t + recall_t}\right\} \quad \text{(Eqn. 4.6)}$$

4.4.3 Result

4.4.3.1 Database update for PFP/ESG

First we discuss the effect of updating the underlying databases of PFP and ESG. The framework of both methods consists of three steps: 1) retrieving similar sequences to a query sequence from a sequence database; 2) extracting GO terms that are associated with the retrieved sequences; 3) and finally predicting GO terms for the query (see Methods). Two different databases are used in the procedure, *i.e.* a sequence database used in Step 1, against which the query is searched and another database in Step 2 that stores GO terms for the retrieved sequences. The latter database is referred to as the annotation database.

The sequence database to be searched against (Step 1) for both PFP and ESG is UniProt (the Swiss-Prot portion). This database is referred to as Swiss-Prot-SeqDB. We have been using a 2008 version of Swiss-Prot, but this time it was updated to the version 01/20/2013.

PFP and ESG use different annotation databases (Step 2). PFP uses the so-called PFPDB, which is an integrated database of GO terms taken from multiple databases.

PFPDB is discussed in details later in this section. ESG uses the GO database downloaded from http://geneontology.org/page/download-ontology as its annotation database. The old version used earlier is from 2008 and the new version that is used in this work (and in CAFA2) was downloaded in 2013.

Table 4.1 describes the differences in the number of sequences and GO terms between the old and new databases. The number of sequences in Swiss-Prot-SeqDB is expanded in the new database to more than double the size (2.45 times) of the old database.

The second row of Table 4.1 is data for PFPDB, the annotation database used for PFP. PFPDB is a collection of GO terms from multiple annotation resources, including UniProt-SwissProt. The updated PFPDB database did not include annotations from SwissProt-Keywords and added two new annotation resources to the previous ones (PIRSF [178] and Reactome [179]). With the updated PFPDB, the functional association matrix (FAM), which is the conditional probability $P(f_a|f_i)$ in Equation 2.1 used in PFP was also updated. In PFPDB, the total number of GO terms in the updated database is increased to almost double (1.91 times) from the old database. The number of unique GO terms in the annotation database for ESG, which is the GO database, increased by 1.78 times from 2008 to 2013.

In Table 4.2, we show the effects of combining multiple annotation resources (from which annotations are transferred) for the updated PFPDB in terms of the sequence coverage and the GO coverage. The sequence coverage is the percentage of the sequences in Swiss-Prot that have at least one GO term annotation. The GO coverage is the percentage of GO terms that are included in PFPDB relative to the entire GO vocabulary. Having a large coverage is essential for the PFP and ESG function prediction methods, because it

directly affects the algorithms' ability to retrieve function information from a PSI-BLAST search result.

Table 4.1 PFP/ESG database update

	2008 version	**2013 version**
Sequence Database (Swiss-Prot-SeqDB)		
Number of sequences	211,104	514,673
PFPDB (Annotation database for PFP)		
Number of unique GO terms	18,327	35,029
External resources for PFPDB	HAMAP, InterPro, SwissProt-keywords, Pfam, PRINTS, ProDom, PROSITE, SMART, TIGRFam	HAMAP, InterPro, Pfam, PRINTS, ProDom, PROSITE, SMART, TIGRFam, PIRSF, Reactome
Annotation Database for ESG		
Number of GO terms	13,420	23,896

Each of SwissProt-GO, InterPro, and Pfam has a very high (>90%) sequence coverage as an annotation resource. In terms of the GO coverage, SwissProt-GO has the highest percentage. The rest of the databases have relatively small coverage, with InterPro being the highest among them; however, its GO coverage is as small as 10.59%. Overall, 98.42% of Swiss-Prot sequences have at least one GO annotation and 60.83% of GO terms in the current GO vocabulary are represented in PFPDB. Compared with the sequence and GO coverage of SwissProt-GO, which was the starting point of the annotation, adding more GO terms from additional sources did not gain much, only about 4% for the sequence coverage and 0.5% for the GO coverage. These results are substantially different from when we constructed PFPDB originally in 2008 [12,13]. At that time, the sequence coverage jumped from 13.4% to 92.9% by importing GO terms from the addi-

tional sources (Hawkins et al., 2008 [12,13], Table II). The reason for the small gain in coverage can probably be attributed to the fact that GO annotations in Swiss-Prot have been far better developed since then and annotations in different databases are better shared between databases now.

Table 4.2 Coverage from additional resources in updated PFPDB

[a]Sequence coverage is the percentage of sequences in Swiss-Prot annotated with at least one GO term after addition of translated terms from the format in column 1. [b]GO coverage is the percentage of terms in the GO vocabulary represented in Swiss-Prot after addition of translated terms from the resource in column 1.

	Sequence Coverage (%)[a]	**GO Coverage (%)[b]**
SwissProt-GO	94.50	60.27
HAMAP	58.35	3.55
InterPro	95.75	10.59
Pfam	92.34	6.47
PRINTS	22.26	3.09
ProDom	5.39	1.18
ProSite	56.45	2.53
SMART	23.25	1.26
TIGRFam	49.92	4.78
PIRSF	18.38	4.29
Reactome	1.46	0.01
ALL	98.42	60.83

4.4.3.2 Benchmarking prediction accuracy of updated for PFP/ESG

Figure 4.5 shows the results of PFP using the old and the updated PFPDB. To simulate a realistic scenario in which close homologs of a query do not exist in the sequence database, when predicting function for a target in the benchmark dataset, similar sequences in the sequence database to the target that have a certain E-value or smaller (*i.e.* more significant) were removed. The E-value cut-off is shown along the x-axis of the

figure. Thus, for example, at E-value of 0.01 (shown by x = 0.01 in the figure), all the sequences in the database that have an E-value of 0.01 or smaller to the query were removed. At x = 0, sequence hits with an E-value of 0 were removed in order to avoid annotation transfer from exactly matched sequences. The y-axis reports the average Fmax score over all benchmark targets.

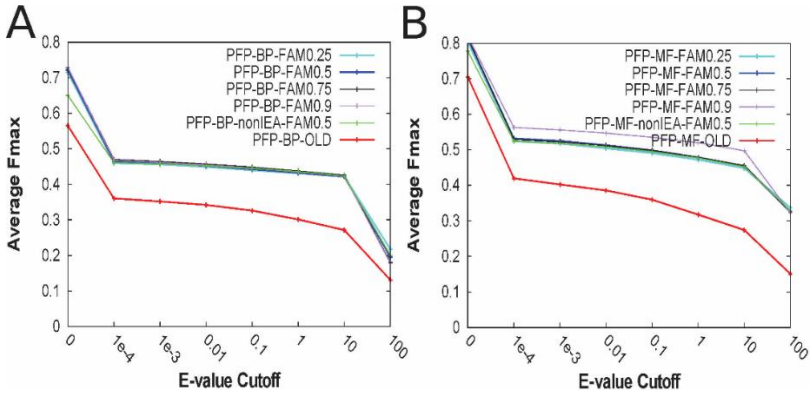

Figure 4.5 Performance of PFP evaluated on GO terms including parental terms

Performance of PFP using the new and the old PFPDB. Before evaluating predictions, both predicted and true GO terms were propagated to the root of the ontology.

***A**, Evaluation on BP GO terms.*

***B**, Evaluation on MF GO terms.*

For this evaluation, we extend both predicted and true GO terms of each target with parental GO terms in the GO hierarchy. For a predicted or true GO term GO^i, all parental GO terms of GO^i in the GO hierarchy (more precisely, a Directed Acyclic Graph or DAG) were added and the performance evaluation was done by comparing the extended GO term sets. This parental propagation on the true and predicted annotation sets was also adopted in the official CAFA assessments. For PFP with the updated PFPDB, different functional association matrix (FAM) score cut-offs were tested. The FAM score is the probability that a GO term f_a co-exists in the annotation of a protein when another GO

term f_i already exists in the annotation of the protein. Concretely, it is the conditional probability $P(f_a|f_i)$ in Equation 1 in the Methods section. For example, in Figure 4.5, PFP-BP(or MF)-FAM0.9 represents the prediction results of PFP using the updated PFPDB and only very strongly associated GO terms in FAM, with a FAM score of 0.9 or higher. On the other hand, PFP-BP(or MF)-FAM0.25 used many GO term associations including ones that are weakly associated, with a conditional probability of 0.25 or higher. For more details of the FAM score, refer to the original paper of the PFP algorithm [12,13].

Figure 4.5 shows predictions for the Biological Process (BP) GO category (Figure 4.5A) and for the Molecular Function (MF) GO category (Figure 4.5B) separately. In Figure 4.5A, all of the PFP predictions with the new PFPDB performed better than PFP with the old database (PFP-BP-OLD). For PFP-BP/MF-OLD, a FAM score threshold of 0.9 was used. Among five different FAM score threshold values (0.25 to 0.9), PFP-BP-FAM0.9 showed the largest average Fmax accuracy across all the E-value cut-off scores. At the first E-value cut-off, 0.0, PFP-BP-FAM0.9 achieved the largest average Fmax score of 0.6873 and PFP-BP-FAM0.75 showed the second highest score of 0.6856.

Comparing the results using the full PFPDB (PFP-BP-FAM0.5) and those using a subset of GO terms in PFPDB that have experimental evidence (i.e. GO terms that are not Inferred from Electronic Annotation, non-IEA) (PFP-BP-nonIEA-FAM0.5), the former had a larger average Fmax score as shown in Fig. 4.5A-B. In Figure 4.5 we excluded IEA GO terms only from PFPDB and kept IEA GO terms for the target proteins as correct terms. Figure 4.5B is the performance on MF GO terms. Overall, prediction accuracy for MF (Figure 4.5B) were higher than for BP (Figure 4.5A). The best-performing prediction setting for MF was again PFP-MF-FAM0.9, with an average Fmax score of 0.7817 at an

E-value cut-off of 0.0 and PFP-MF-FAM0.75 was the second best (0.7644). Consistent with Figure 4.5A, PFP with the old database was the worst (an Fmax score of 0.6479 at an E-value cut-off of 0.0). In the original paper of PFP [12,13], a similar performance comparison was conducted with different FAM score thresholds (Figure 4 in the original paper of PFP [12,13]), where PFP with a FAM score cut-off of 0.9 was shown to perform best among others. Thus, the findings for the current benchmark with the updated database is consistent with the earlier study [12,13].

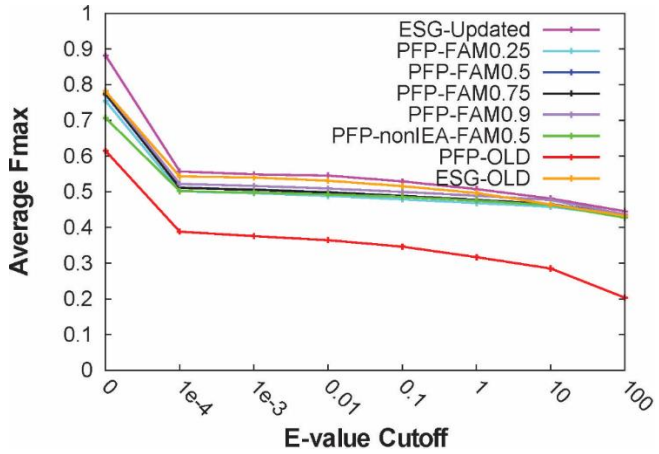

Figure 4.6 Performance of PFP and ESG on GO terms including parental terms

Each predicted and true GO term was propagated to the root of the ontology before evaluation. GO terms in all three ontologies (BP, MF, CC) were used in computing prediction accuracy.

In Figure 4.6, we added the ESG's results to the plots. The Fmax score was computed using GO terms for all three ontologies (BP, MF, and Cellular Component (CC)). ESG with the updated database (ESG-Updated) performed the best (average Fmax of 0.8401 at an E-value cut-off of 0.0) among the eight settings compared. ESG-OLD was the second best (an average Fmax of 0.7655 at E-value 0.0), and PFP-OLD had the lowest accuracy (an average Fmax of 0.5852 at E-value 0.0).

In summary, updating the databases contributed in improving the prediction accuracy (average Fmax scores) substantially for both PFP and ESG. ESG showed a higher average Fmax score than PFP. The best-performing FAM score threshold value for PFP was 0.9, which was consistent with our earlier study.

Table 4.3 Average Fmax for individual and ensemble methods

All true and predicted annotations have been propagated to the root of the ontology. All three GO categories were used in the evaluation.

Method	**Average Fmax**
PFP-Updated	0.7447
PFP-OLD	0.5852
ESG-Updated	0.8401
ESG-OLD	0.7655
FFPred	0.3248
PFAM	0.5583
HHblits	0.4662
PSI-BLAST	0.5991
CONS	0.8085
FPM_MaxLen	0.7937
FPM_MaxScoreLen	0.4628

4.4.3.3 Prediction performance of ensemble methods

Next we discuss the prediction accuracy of two ensemble methods in comparison with individual component methods (Table 4.3). Two ensemble methods, CONS and FPM, were constructed that combine GO predictions from six individual methods: PFP, ESG, PFAM, PSI-BLAST, HHblits, and FFPred. The CONS method computes a score for a GO term as a weighted sum of scores of the GO terms from the component methods. The weight of a method is prior knowledge of the accuracy of the method. FPM selects combinations of GO terms that are computed by multiple methods with a sufficiently

high score (see Methods). In Table 4.3, we show results of two variations of FPM. FPM_maxLen is an FPM method that selects a GO term set with the largest size (number of GO terms) from a candidate pool of predicted GO term sets with a sufficiently large score. FPM_maxScoreLen, on the other hand, selects the GO term set with the highest overall score (often resulting in outputs of a small number of GO terms). Overall, out of all the individual and ensemble methods, the most successful method was ESG-Updated, which showed the largest average Fmax score of 0.8401. CONS came at a second (Fmax score of 0.8085), followed by FPM_maxLen (Fmax score 0.7937), ESG-Old, and PFP-Updated in this order. On this benchmark, FFPred, PFAM, and HHblits performed very poorly relative to PFP-Updated and ESG-Updated.

To further understand performance of the ensemble methods, we next examined the number of wins for each method, *i.e.* the number of times that each method showed the largest Fmax score (Figure 4.7). In this analysis, for each target the confidence cut-off values used for each component method were optimized to give the largest Fmax score to the target, in order to understand how well ensemble methods can assemble individual predictions in the best case scenario in which each component method offers its best possible prediction. In terms of the number of wins, ESG is the best and CONS and FPM follow in that order, which is consistent with the results on the average Fmax scores (Table 4.3) (note that there are queries where multiple methods tied for same Fmax score). Overall, the two ensemble methods did not show better performance than the best component method, ESG, but as illustrated later there are many cases in which the ensemble methods successfully selected correct GO terms from different component methods.

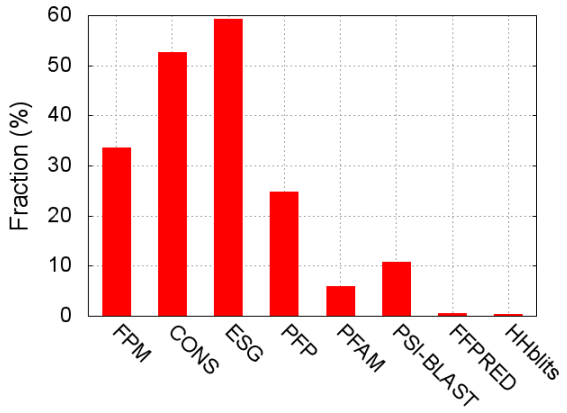

Figure 4.7 Fraction of queries where method showed largest Fmax score

The fraction on the y-axis was computed as the number of queries in which a method had the largest Fmax score over the total number of queries (2055 protein sequences). FPM in this graph denotes FPM_MaxLen because it performed better than its counterpart, FPM_maxscoreLen. The fraction does not sum up to 100% due to cases where multiple methods tied for the largest Fmax score.

From Figure 4.7, we can see that CONS and FPM provided the most accurate prediction for 52.2% and 40.0% of the queries.

4.4.3.4 Case studies of the CONS method

Table 4.4 illustrates how CONS combines predictions of the individual methods. The first two examples (Table 4.4A and Table 4.4B) are cases where CONS improved the prediction over the individual methods. Similar to Figure 4.7, the Fmax computation for this analysis is done at the individual protein level. The first example, Table 4.4A, is predictions for a capsid protein from the Hepatitis E virus (UniProt ID: Q9IVZ8). For this protein, CONS had the highest Fmax score, 0.667, and PFP had the second-highest, with an Fmax score of 0.575 (Fmax was computed after parental propagation). In its top hits, CONS correctly predicted all five GO annotations of this protein (shown in bold in the

table) together with two parental terms of correct GO terms (shown in italics in the table). Interestingly, PFP, the second-best predictor, predicted only four of the five correct GO terms, whereas the last one GO:0039615, came from ESG.

For the second example (Table 4.4B), CONS had the largest Fmax score of 0.915, followed by PSI-BLAST, which had an Fmax score of 0.824. The query, succinate dehydrogenase iron-sulfur subunit, has eight GO term annotations. Among them, CONS predicted seven with high confidence scores, and one, GO:0000104, at a low score. Out of these eight GO term annotations, GO:00051539, GO:0046872, and GO:0006099 were predicted with high scores by three individual methods, PFP, ESG, and PSI-BLAST. GO:0000104 was strongly predicted by PSI-BLAST. GO:0009055 and GO:0022900 were predicted with relatively high scores by ESG and PFP. Thus, this is an example which shows that CONS can successfully select different correct terms from different methods.

There are also cases that show the opposite trend, where CONS could not improve prediction (Table 4.4C). In the third example, showing the GO annotations of ATP-dependent RNA helicase, the best Fmax score among the component methods was from ESG (0.761), followed by PSI-BLAST (0.673), PFP (0.667), and PFAM (0.653), while CONS had an Fmax score of 0.66 and was ranked fourth among all methods. In this example, all five correct GO terms were predicted by ESG, but four of them were with weak scores. PFP predicted only two correct terms, GO:0005524 (ATP binding) with a high score and GO:0000027 (ribosomal large subunit assembly) with a low score, while PSI-BLAST, FFPred, and PFAM only predicted GO:0005524 among the five correct

terms. Thus, combining them could not increase the scores of the correct terms, and rather, introduced over 100 incorrect terms.

Table 4.4 Examples of predictions by CONS and individual component methods

A Capsid protein (UniProt ID: Q9IVZ8)
GO terms in bold are correct annotations of the protein. Terms in italic indicate parental terms of correct GO terms. Terms in parentheses are wrong predictions.
For CONS prediction, GO terms that have a confidence score larger than 0.4 are listed.
For PFP prediction, GO terms that have a confidence score larger than 0.5 are listed.
For ESG, all predicted GO terms are shown.

CONS	GO:0019028 1.00 viral capsid GO:0005198 0.97 structural molecule activity GO:0019012 0.70 virion GO:0039615 0.68 T=1 icosahedral viral capsid (GO:0032774) 0.43 GO:0003723 0.43 RNA binding GO:0044228 0.43 host cell surface GO:0030430 0.43 host cell cytoplasm
PFP	GO:0044228 1.00 host cell surface (GO:0032774) 1.00 GO:0030430 1.00 host cell cytoplasm GO:0005198 1.00 structural molecule activity GO:0003723 1.00 RNA binding (GO:0006351) 0.71 GO:0043656 0.65 intracellular region of host GO:0033646 0.65 host intracellular part (GO:0008150) 0.59 GO:0003676 0.59 nucleic acid binding
ESG	GO:0019012 1.00 virion GO:0019028 1.00 viral capsid GO:0039615 0.99 T=1 icosahedral viral capsid (GO:0019048) 0.15 (GO:0030683) 0.15 (GO:0039573) 0.15

B Succinate dehydrogenase iron-sulfur subunit (UniProt ID: P51053)
For CONS, PFP, and ESG prediction, GO terms that have a confidence score equal to or larger than 0.10, 0.20, and 0.56 are shown (i.e. up to the last correct GO term). For PSI-BLAST all predicted GO terms are shown.

CONS	GO:0051536 1.00 iron-sulfur cluster binding GO:0009055 0.25 electron carrier activity GO:0051539 0.24 4 iron, 4 sulfur cluster binding GO:0046872 0.24 metal ion binding GO:0006099 0.22 tricarboxylic acid cycle (GO:0016020) 0.21 GO:0051537 0.21 2 iron, 2 sulfur cluster binding GO:0051538 0.21 3 iron, 4 sulfur cluster binding GO:0016491 0.16 oxidoreductase activity GO:0055114 0.16 oxidation-reduction process GO:0009060 0.16 aerobic respiration GO:0022900 0.14 electron transport chain (GO:0008177) 0.13 …and 9 more terms GO:0000104 0.10 succinate dehydrogenase activity
PFP	GO:0055114 1.00 oxidation-reduction process GO:0051540 1.00 metal cluster binding …and 10 more terms GO:0051539 0.52 4 iron, 4 sulfur cluster binding GO:0009055 0.46 electron carrier activity (GO:0005886) 0.46 (GO:0071944) 0.44 (GO:0044435) 0.43 GO:0022900 0.42 electron transport chain …and 9 more terms GO:0046872 0.35 metal ion binding …and 6 more terms GO:0006099 0.33 tricarboxylic acid cycle …and 8 more terms GO:0000104 0.25 succinate dehydrogenase activity (GO:0050136) 0.23 (GO:0003954) 0.23 GO:0051537 0.22 2 iron, 2 sulfur cluster binding GO:0051538 0.20 3 iron, 4 sulfur cluster binding

ESG	(GO:0005743) 0.66 GO:0006099 0.66 tricarboxylic acid cycle (GO:0008177) 0.66 GO:0009055 0.66 electron carrier activity GO:0046872 0.66 metal ion binding GO:0051537 0.66 2 iron, 2 sulfur cluster binding GO:0051538 0.66 3 iron, 4 sulfur cluster binding GO:0051539 0.66 4 iron, 4 sulfur cluster binding (GO:0005749) 0.60 (GO:0048039) 0.60 GO:0022900 0.56 electron transport chain
PSI-BLAST	(GO:0016020) 0.80 GO:0051538 0.80 3 iron, 4 sulfur cluster binding GO:0051539 0.80 4 iron, 4 sulfur cluster binding GO:0051536 0.80 iron-sulfur cluster binding (GO:0006810) 0.80 (GO:0009061) 0.80 GO:0046872 0.80 metal ion binding GO:0006099 0.80 tricarboxylic acid cycle GO:0009060 0.80 aerobic respiration (GO:0005489) 0.80 GO:0051537 0.80 2 iron, 2 sulfur cluster binding (GO:0005506) 0.80 GO:0000104 0.80 succinate dehydrogenase activity (GO:0006118) 0.80 GO:0016491 0.80 oxidoreductase activity

C ATP-dependent RNA helicase SrmB (UniProt ID: P21507)

CONS	GO:0005524 1.0000 ATP binding GO:0003676 0.2937 nucleic acid binding GO:0004386 0.2445 helicase activity GO:0000166 0.2370 nucleotide binding GO:0008026 0.2350 ATP-dependent helicase activity GO:0016787 0.1987 hydrolase activity GO:0003723 0.1860 RNA binding (GO:0003677) 0.1683 …and 37 more terms GO:0004004 0.0364 ATP-dependent RNA helicase activity GO:0044424 0.0364 intracellular part (GO:0051716) 0.0353 (GO:0071843) 0.0351 …and 142 more terms

	GO:0000027 0.0079 ribosomal large subunit assembly (GO:0050789) 0.0078 (GO:0051252) 0.0078 …and 3 more terms GO:0033592 0.0073 RNA strand annealing activity GO:0030687 0.0073 preribosome, large subunit precursor
PFP	GO:0044464 1.00 cell part GO:0008150 1.00 biological process GO:0005623 1.00 cell GO:0003676 1.00 nucleic acid binding GO:0004386 0.99 helicase activity GO:0005575 0.94 cellular component GO:0022613 0.84 ribonucleoprotein complex biogenesis GO:0003674 0.84 molecular function (GO:0090304) 0.77 GO:0032559 0.76 adenyl ribonucleotide binding GO:0005524 0.76 ATP binding …and 116 more terms GO:0004004 0.11 ATP-dependent RNA helicase activity (GO:0080090) 0.10 GO:0070013) 0.10 …and 407 more terms GO:0000027 0.01 ribosomal large subunit assembly
ESG	GO:0000166 0.80 nucleotide binding GO:0003676 0.80 nucleic acid binding GO:0003723 0.80 RNA binding GO:0005524 0.80 ATP binding GO:0004386 0.73 helicase activity GO:0008026 0.73 ATP-dependent helicase activity GO:0016787 0.73 hydrolase activity (GO:0000184) 0.46 (GO:0005634) 0.46 (GO:0006364) 0.46 GO:0042254 0.46 ribosome biogenesis (GO:0005737) 0.38 GO:0004004 0.28 ATP-dependent RNA helicase activity GO:0000027 0.07 ribosomal large subunit assembly (GO:0005515) 0.07

CHAPTER 5. DISCUSSSION & SUMMARY

5.1 Moonlighting proteins

Moonlighting proteins have more than one independent function. It is speculated that moonlighting proteins are not few in number and expected to be found more in the future. Identification of moonlighting proteins indicates that potential secondary functions need to be considered when it comes to protein function, which has significant impact on functional genomics, proteomics, and computational gene function annotation [61].

In the first part of MP characterization, we examined current GO annotations of known moonlighting proteins. We found that the GO term annotations for moonlighting proteins can be clustered into more than one cluster based on the semantic similarity between pairs of GO terms. Thus, even in the case that moonlighting proteins are not labelled as such in the annotation database, we will be able to identify them by observing the functional divergence of annotated GO terms. Based on this intuitive observation, we analyzed E. coli proteins in the database and identified novel moonlighting proteins. The majority of interacting proteins of a moonlighting protein shared the primary function of the moonlighting protein and we found that a substantial fraction of the interacting proteins were themselves moonlighting proteins.

The characteristics of moonlighting proteins were investigated by comparing their features with those of non-moonlighting proteins. In general, finding examples that do not possess a certain property is not straightforward as future research may find that the examples actually do have the property. So are non-moonlighting proteins – there is an undeniable possibility that non-moonlighting proteins used in this study will be found as moonlighting in the future. Nevertheless we believe the current research is valuable and has contributed in progressing our understanding of moonlighting proteins because the non-moonlighting proteins were selected in a reasonable way and also because the differences and similarities of characteristics of moonlighting and non-moonlighting proteins were clarified that can serve as hypotheses in the future works. We would also like to point out that similar approaches of selecting negative data sets were taken in analyzing protein-protein interactions (by constructing a non-interacting protein dataset, Negatome [180]) and in analyzing proteins with particular functions (by constructing the NoGo database [181]), which contributed in development of computational prediction methods and thereby advance our understanding and the research field.

We observed significant functional divergence in physically interacting proteins with moonlighting proteins, which could be a good feature to use for predicting of moonlighting proteins. However, the other features of moonlighting proteins in omics data were weak. Thus, predicting moonlighting proteins from an individual feature may not be an easy task. This also reminds us that moonlighting functions are observed in various physiological conditions of a cell, which differ for each moonlighting protein. Therefore, ultimately, prediction of moonlighting proteins or secondary functions of a protein needs a holistic understanding of behavior of molecules in a cell. In practice, this means that

integrating various different cell-level data will be effective in prediction, which includes proteomics, ionomics, phenotypic data of mutants, bioinformatics predictions, computational simulations of pathways, and molecular dynamics of biomolecules. Such an automated computational method would be useful in resolving many ambiguities in proteomics analysis as well as in unfolding many complexities of protein functions. Improved understanding of moonlighting functions of proteins can be a touchstone for our knowledge of molecular biology, because it requires comprehensive, multilevel data and deep knowledge of the cell.

Based on the above analysis, we proposed a novel computational approach, MPFit, for detecting MPs from GO annotations or omics-based features. Compared to existing MP prediction methods that use only the GO term feature [114] or one feature type [49,113], MPFit can be applied to a larger fraction of proteins in a genome due to the use of several omics-based features and the implemented imputation protocol for filling missing features. As the mechanisms by which MPs exhibit multiple functions differ from case by case, using various feature types is reasonable to capture MPs of different nature. MPFit was developed as a model that leverages a diverse protein interaction features [50] to predict MPs. Complementary to MPFit, we used a completely different knowledgebase for extracting unique features of MPs in order to make MP prediction and complements our previous MP study. Our proposed method DextMP is the first text-based MP prediction method to our knowledge. Compared to existing methods that use only the GO term feature [114] or one feature type [49,113] or our previous method MPFit [50], DextMP shows significant improvement of performance for both specificity over known MPs and wide applicability due to its sole reliability over textual description associated to proteins.

Based on these current works on MPs, a useful future direction would be to extend MPFit and DextMP to work as not only a binary prediction models for MP/non-MP, but predict the GO terms of the multiple functions of the predicted MPs. Such an extension would give a more comprehensive understanding of the functional landscape of MPs, even for the predictions made in genome-scale. Current MPFit model makes the MP prediction essentially from the functional clusters in the protein association networks (i.e., PPI, GE etc.). Performing GO enrichment on the functional clusters of the interaction networks could be start to find out the different biological functions predicted for the MPs. Another future direction on MPFit model would be make it's feature space broader with usage of more omics data, for example, incorporation of KEGG [2] pathway information along with other omics association network of proteins. Lastly, development of publicly available servers for both the methods, i.e., MPFit and DextMP, would provide a huge platform for making blind MP prediction on novel proteins or genomes.

5.2 Group function prediction

Existing computational AFP methods aims at identifying individual functions of proteins, and there is no existing model that can identify protein's group function. The perspective of "group" function annotation to a set of proteins opens up novel possibilities of understanding the functional nature of complex cellular interactions of such protein groups. In this research, we propose a model that takes groups of proteins found to work together in certain biological experiment, disease, or pathway, maps them to several functional linkage networks and integrates them, and then uses an iterative clustering and graphical modeling based schema to find group functions of the input proteins. As a

backbone to the function prediction model of protein group, we use an integration of a number of major protein interaction networks. We propose a conditional random field (CRF)-based framework that predicts function of the "protein groups" in the network based on group neighborhood, and iteratively updates the function annotation of the unknown group members such that it reflects the protein's group activity.

A future direction on this group function prediction problem would be to answer other associated questions regarding "group function" of the set of proteins, such as: A. for an input group of proteins that may have multiple group function, can the group functions be directly inferred from the function annotations of the clusters in the GFP model, rather the enriched GO terms of the predicted functions of the input gene groups? B. What are the proteins other than the ones in the input gene group that may be involved in the group functions? C. From the predicted group functions, can the unannotated input protein's functions inferred in a more detailed level than the group function notion? Extension of the current GFP model that can answer these associated questions would be useful in understanding the group activities of proteins in the cell.

5.3 Update on AFP methods and CAFA challenge

An essential task in bioinformatics is to propose and develop new tools and new ideas. However, to support the biology community, it is equally important to maintain and update previously-developed software tools so that users can continue using them. For a prediction method, it is important that the prediction accuracy be improved over time so that it can keep pace with other existing methods of the same type. Since the original development of PFP and ESG, the two methods have been benchmarked in CAFA1

by the organizers [54] as well as by our group [51] and their webservers have been recently renovated so that users can obtain prediction information in more organized fashion [52] (http://kiharalab.org/pfp and http://kiharalab.org/esg). The participation in CAFA2 provided us with a suitable opportunity to update databases for PFP and ESG and to develop ensemble approaches.

We have shown that the prediction performance of PFP and ESG improved by updating databases. Although it may sound obvious to expect better performance with updated databases, it is not necessarily a given, especially considering the recent very-fast expansion of databases. This fast expansion has caused several problems, such as increasing sparseness of useful data (*i.e.* functional annotation) relative to the size of sequence databases and error propagation of incorrect annotations [182].

The ensemble methods, CONS and FPM, showed the largest average Fmax score over all individual component methods except for ESG. The six individual methods used in the ensemble methods may not be the best choice, since their performances were imbalanced, *i.e.* a large discrepancy in accuracy between PFP/ESG and the rest of the methods. Also, it is noteworthy that all the individual methods use the same source of information as input, *i.e.* sequence data. Since both CONS and FPM seem to have an ability to assemble the more accurate GO term set as predictions compared to individual methods (Figure 4.7), it will be interesting to apply the two ensemble methods to integrate a better combination of individual methods that use a wide variety of information sources such as protein structures and protein-protein interaction data and whose performance is more balanced.

REFERENCES

REFERENCES

1. Benson DA, Cavanaugh M, Clark K, Karsch-Mizrachi I, Lipman DJ, Ostell J *et al*.: **GenBank.** *Nucleic Acids Research* 2013, 41: D36-D42.

2. Kanehisa M, Goto S: **KEGG: Kyoto encyclopedia of genes and genomes.** *Nucleic Acids Research* 2000, 28: 27-30.

3. Pearson WR, Lipman DJ: **Improved tools for biological sequence comparison.** *Proceedings of the National Academy of Sciences* 1988, 85: 2444-2448.

4. Pearson WR: **Rapid and sensitive sequence comparison with FASTP and FASTA.** *Methods in Enzymology* 1990, 183: 63-98.

5. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ: **Basic local alignment search tool.** *Journal of Moleculer Biology* 1990, 403-410.

6. Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, Miller W *et al*.: **Gapped BLAST and PSI-BLAST: A new generation of protein database search programs.** *Nucleic Acids Research* 1997, 25: 3389-3402.

7. Pietrokovski S, Henikoff JG, Henikoff S: **The Blocks database: A system for protein classification.** *Nucleic Acids Research* 1996, 24: 197-200.

8. Bru C, Courcelle E, Carrere S, Beausse Y, Dalmar S, Kahn D: **The ProDom database of protein domain families: More emphasis on 3D.** *Nucleic Acids Research* 2005, 33: D212-D215.

9. Attwood TK, Bradley P, Flower DR, Gaulton A, Maudling N, Mitchell AL *et al*.: **PRINTS and its automatic supplement, prePRINTS.** *Nucleic Acids Research* 2003, 31: 400-402.

10. Finn RD, Mistry J, Schuster-Bockler B, Griffiths-Jones S, Hollich V, Lassmann T *et al*.: **Pfam: Clans, web tools and services.** *Nucleic Acids Research* 2006, 34: D247-D251.

11. Hunter S, Apweiler R, Attwood TK, Bairoch A, Bateman A, Binns D *et al*.: **InterPro: The integrative protein signature database.** *Nucleic Acids Research* 2009, 37: D211-D215.

12. Hawkins T, Luban S, Kihara D: **Enhanced automated function prediction using distantly related sequences and contextual association by PFP.** *Protein Science* 2006, 15: 1550-1556.

13. Hawkins T, Chitale M, Luban S, Kihara D: **PFP: Automated prediction of gene ontology functional annotations with confidence scores using protein sequence data.** *Proteins: Structure, Function, and Bioinformatics* 2009, 74: 566-582.

14. Chitale M, Hawkins T, Park C, Kihara D: **ESG: Extended similarity group method for automated protein function prediction.** *Bioinformatics* 2009, 25: 1739-1745.

15. Martin D, Berriman M, Barton G: **GOtcha: A new method for prediction of protein function assessed by the annotation of seven genomes.** *BMC Bioinformatics* 2004, 5: 178-194.

16. Vinayagam A, del Val C, Schubert F, Eils R, Glatting KH, Suhai S *et al*.: **GOPET: A tool for automated predictions of Gene Ontology terms.** *BMC Bioinformatics* 2006, 7: 161-167.

17. Zehetner G: **OntoBlast function: From sequence similarities directly to potential functional annotations by ontology terms.** *Nucleic Acids Research* 2003, 31: 3799-3803.

18. Khan S, Situ G, Decker K, Schmidt CJ: **GoFigure: Automated Gene Ontology annotation.** *Bioinformatics* 2003, 19: 2484-2485.

19. Wass MN, Sternberg MJ: **ConFunc: Functional annotation in the twilight zone.** *Bioinformatics* 2008, 24: 798-806.

20. Engelhardt BE, Jordan MI, Muratore KE, Brenner SE: **Protein molecular function prediction by Bayesian phylogenomics.** *PLoS Computational Biology* 2005, 1: e45.

21. Krishnamurthy N, Brown D, Sjolander K: **FlowerPower: Clustering proteins into domain architecture classes for phylogenomic inference of protein function.** *BMC Evolutionary Biology* 2007, 7: S12.

22. Storm CEV, Sonnhammer ELL: **Automated ortholog inference from phylogenetic trees and calculation of orthology reliability.** *Bioinformatics* 2002, 18: 92-99.

23. Brown MPS, Grundy WN, Lin D, Cristianini N, Sugnet CW, Furey TS *et al*.: **Knowledge-based analysis of microarray gene expression data by using support vector machines.** *Proceedings of the National Academy of Sciences* 2000, 97: 262-267.

24. Eisen MB, Spellman PT, Brown PO, Botstein D: **Cluster analysis and display of genome-wide expression patterns.** *Proceedings of the National Academy of Sciences* 1998, 95: 14863-14868.

25. Gao L, Li X, Guo Z, Zhu M, Li Y, Rao S: **Widely predicting specific protein functions based on protein-protein interaction data and gene expression profile.** *Sci China C Life Sci* 2007, 50: 125-134.

26. Khatri P, Dr-âghici S: **Ontological analysis of gene expression data: Current tools, limitations, and open problems.** *Bioinformatics* 2005, 21: 3587-3595.

27. van Noort V, Snel B, Huynen MA: **Predicting gene function by conserved co-expression.** *Trends in Genetics* 2003, 19: 238-242.

28. Gherardini PF, Helmer-Citterich M: **Structure-based function prediction: Approaches and applications.** *Briefings in functional genomics & proteomics* 2008, 7: 291-302.

29. Marti-Renom M, Rossi A, Al-Shahrour F, Davis F, Pieper U, Dopazo J *et al*.: **The AnnoLite and AnnoLyze programs for comparative annotation of protein structures.** *BMC Bioinformatics* 2007, 8: S4.

30. Martin ACR, Orengo CA, Hutchinson EG, Jones S, Karmirantzou M, Laskowski RA *et al*.: **Protein folds and functions.** *Structure* 1998, 6: 875-884.

31. Pal D, Eisenberg D: **Inference of protein function from protein structure.** *Structure* 2005, 13: 121-130.

32. Ponomarenko JV, Bourne PE, Shindyalov IN: **Assigning new GO annotations to protein data bank sequences by combining structure and sequence homology.** *Proteins: Structure, Function, and Bioinformatics* 2005, 58: 855-865.

33. Thornton JM, Todd AE, Milburn D, Borkakoti N, Orengo CA: **From structure to function: Approaches and limitations.** *Nature Structural Biology* 2000, 7: 991-994.

34. Chikhi R, Sael L, Kihara D: **Realtime ligand binding pocket database search using local surface descriptors.** *Proteins: Structure, Function, and Bioinformatics* 2010, 78: 2007-2028.

35. Sael L, Kihara D: **Binding ligand prediction for proteins using partial matching of local surface patches.** *International Journal of Molecular Sciences* 2010, 11: 5009-5026.

36. Sael L, Chitale M, Kihara D.: **Structure and sequence-based function prediction for non-homologous proteins. Journal of Structural and Functional Genomics.** *Journal of Structural and Functional Genomics* 2012, 13: 111-123.

37. Brun C, Chevenet F, Martin D, Wojcik J, Guenoche A, Jacq B: **Functional classification of proteins for the prediction of cellular function from a protein-protein interaction network.** *Genome Biology* 2003, 5: R6.1-R6.13.

38. Chua HN, Sung WK, Wong L: **Exploiting indirect neighbours and topological weight to predict protein function from protein-protein interactions.** *Bioinformatics* 2006, 22: 1623-1630.

39. Letovsky S, Kasif S: **Predicting protein function from protein/protein interaction data: A probabilistic approach.** *Bioinformatics* 2003, 19 Suppl 1: i197-i204.

40. Nariai N, Kolaczyk ED, Kasif S: **Probabilistic protein function prediction from heterogeneous genome-wide data.** *PLoS One* 2007, **2:** e337.1-e337.7.

41. Sharan R, Ulitsky I, Shamir R: **Network-based prediction of protein function.** *Molecular Systems Biology* 2007, 3: 88-100.

42. Deng M, Tu Z, Sun F, Chen T: **Mapping gene ontology to proteins based on protein-protein interaction data.** *Bioinformatics* 2004, 20: 895-902.

43. Gene Ontology Consortium: **Gene Ontology annotations and resources.** *Nucleic Acids Research* 2013, 41: D530-D535.

44. Jeffery C: **Moonlighting proteins.** *Trends in Biochemical Sciences* 1999, 24: 8-11.

45. Campbell RM, Scanes CG: **Endocrine peptides 'moonlighting' as immune modulators: Roles for somatostatin and GH-releasing factor.** *Journal of Endocrinology* 1995, 147: 383-396.

46. Weaver DT: **Telomeres: Moonlighting by DNA repair proteins.** *Current Biology* 1998, 8: R492-R494.

47. Khan I, Kihara D: **Computational characterization of moonlighting proteins.** *Biochemical Society Transactions* 2014, 42: 1780-1785.

48. Khan I, Chen Y, Dong T, Hong X, Takeuchi R, Mori H *et al*. (Eds):**Genome-scale identification and characterization of moonlighting proteins.** *Biology Direct* 2014, 9: 1-29.

49. Khan I, Chitale M, Rayon C, Kihara D: **Evaluation of function predictions by PFP, ESG, and PSI-BLAST for moonlighting proteins.** *BMC Proceedings* 2012, 6(Suppl 7):S5.

50. Khan I, Kihara D: **Genome-scale prediction of moonlighting proteins using diverse protein association information.** *Bioinformatics* 2016, In Press.

51. Chitale M, Khan IK, Kihara D (Eds):**In-depth performance evaluation of PFP and ESG sequence-based function prediction methods in CAFA 2011 experiment.** In *BMC Bioinformatics* 2013, 14: S2.

52. Khan IK, Wei Q, Chitale M, Kihara D (Eds):**PFP/ESG: Automated protein function prediction servers enhanced with Gene Ontology visualization tool.** *Bioinformatics* 2014, 31: 271-272.

53. Khan IK, Wei Q, Chapman S, Kc DB, Kihara D: **The PFP and ESG protein function prediction methods in 2014: Effect of database updates and ensemble approaches.** *Gigascience* 2015, 4: 1-14.

54. Radivojac P, Clark WT, Oron TR, Schnoes AM, Wittkop T, Sokolov A *et al*.: **A large-scale evaluation of computational protein function prediction.** *Nature Methods* 2013, 10: 221-227.

55. Piatigorsky J, Wistow GJ: **Enzyme/crystallins: Gene sharing as an evolutionary.** *Cell* 1989.

56. Wistow GJ, Kim H: **Lens protein expression in mammals: Taxon-specificity and the recruitment of crystallins.** *Journal of Moleculer Evolution* 1991, 262-269.

57. Piatigorsky J: **Multifunctional lens crystallins and corneal enzymes. More than meets the eye.** *Annals of New York Academoy of Sciences* 1998, 842: 7-15.

58. Piatigorsky J: **Crystallin genes: Specialization by changes in gene regulation may precede gene duplication.** Journal of *Structal & Functional Genomics* 2003, 131-137.

59. Graham C, Hodin J, Wistow GJ: **A retinaldehyde dehydrogenase as a structural protein in a mammalian eye lens. Gene recruitment of eta-crystallin.** *The Journal of Biological Chemistry* 1996, 15623-15628.

60. Jeffery C: **Moonlighting proteins ─ An update.** *Molecular BioSystems* 2009, **5:** 345-350.

61. Jeffery C: **Moonlighting proteins: Complications and implications for proteomics research.** *Drug Discovery Today: TARGETS* 2004, 3: 71-78.

62. Moghaddam A, Bicknell R: **Expression of platelet-derived endothelial cell growth factor in Escherichia coli and confirmation of its thymidine phosphorylase activity.** *Biochemistry* 1992, 31: 12141-12146.

63. Ostrovsky de Spicer P, Maloy S.: **PutA protein, a membrane-associated flavin dehydrogenase, acts as a redox-dependent transcriptional regulator.** *Proceedings of the National Academy of Sciences* 1993, 90: 4295-4298.

64. Mowbray SL, Koshland DE Jr.: **Mutations in the aspartate receptor of Escherichia coli which affect aspartate binding.** *Journal of Biological Chemistry* 1990, 265: 15638-15643.

65. Meyer-Siegler K, Mauro DJ, Seal G, Wurzer J, deRiel JK, Sirover MA.: **A human nuclear uracil DNA glycosylase is the 37-kDa subunit of glyceraldehyde-3-phosphate dehydrogenase.** *Proceedings of the National Academy of Sciences*1991, 88: 8460-8464.

66. Soker S, Takashima S, Miao HQ, Neufeld G, Klagsbrun M: **Neuropilin-1 is expressed by endothelial and tumor cells as an isoform-specific receptor for vascular endothelial growth factor.** *Cell* 1998, 92: 735-745.

67. Banerjee S, Nandyala AK, Raviprasad P, Ahmed N, Hasnain SE: **Iron-dependent RNA-binding activity of Mycobacterium tuberculosis aconitase.** *Journal of Bacteriology* 2007, 189: 4046-4052.

68. Lu M, Sautin Y, Holliday L, Gluck S: **The glycolytic enzyme aldolase mediates assembly, expression, and activity of vacuolar H+-ATPase.** *The Journal of Biological Chemistry* 2004, 8732-8739.

69. Huberts DH, Vander Klei IJ.: **Moonlighting proteins: An intriguing mode of multitasking.** *Biochimica et Biophysica Acta* 2010, 1803: 520-525.

70. Jeffery C: **Proteins with neomorphic moonlighting functions in disease.** *IUBMB Life* 2011, 63: 489-494.

71. Sriram G, Martinez JA, McCabe ER, Liao JC, Dipple KM: **Single-gene disorders: What role could moonlighting enzymes play?** *American Journal of Human Genetics* 2005, 76: 911-924.

72. Finn RD, Bateman A, Clements J, Coggill P, Eberhardt RY, Eddy SR *et al*.: **The Pfam protein families database.** *Nucleic Acids Research* 2014, 42: D222-D230.

73. Bru Catherine, Courcelle Emmanuel, Carrère Sébastien, Beausse Yoann, Dalmar Sandrine, Kahn Daniel: **The ProDom database of protein domain families: More emphasis on 3D.** *Nucleic Acids Research* 2005, 212-215.

74. Hunter S, Jones P, Mitchell A, Apweiler R, Attwood TK, Bateman A *et al*.: **InterPro in 2011: New developments in the family and domain prediction database.** *Nucleic Acids Research* 2012, 40: D306-D312.

75. Ozimek P, Kotter P, Veenhuis M, Klei IJ: **Hansenula polymorpha and Saccharomyces cerevisiae Pex5p's recognize different, independent peroxisomal targeting signals in alcohol oxidase.** *FEBS Letters* 2006, 580: 46-50.

76. Chen XJ, Wang X, Kaufman BA, Butow RA.: **Aconitase couples metabolic regulation to mitochondrial DNA maintenance.** *Science* 2005, 307: 714-717.

77. Tang Y, Guest J (Eds):**Direct evidence for mRNA binding and post-transcriptional regulation by Escherichia coli aconitases.** *Microbiology* 1999, 145: 3069-3079.

78. Gomez A, Domedel N, Cedano J, Pinol J, Querol E: **Do current sequence analysis algorithms disclose multifunctional (moonlighting) proteins?** *Bioinformatics* 2003, 19: 895-896.

79. Gómez A, Hernández S, Amela I, Piñol J, Cedano J, Querol E.: **Do protein-protein interaction databases identify moonlighting proteins?** *Molecular BioSystems* 2011, 7: 2379-2382.

80. Becker E, Robisson B, Chapple CE, Guénoche A, Brun C. (Eds):**Multifunctional proteins revealed by overlapping clustering in protein interaction network.** *Bioinformatics* 2012, 28: 84-90.

81. Hernández S, Amela I, Cedano J, Piñol J, Perez-Pons J, Mozo-Villarias A *et al*. (Eds):**Do Moonlighting Proteins Belong to the Intrinsically Disordered Protein Class?** *Journal of Proteomics and Bioinformatics* 2011, 5: 262-264.

82. Hernández S, Calvo A, Ferragut G, Franco L, Hermoso A, Amela I *et al*. (Eds):**Can bioinformatics help in the identification of moonlighting proteins?** *Biochemical Society Transactions* 2014, 42: 1692-1697.

83. Hernández S, Ferragut G, Amela I, Perez-Pons J, Piñol J, Mozo-Villarias A *et al*.: **MultitaskProtDB: A database of multitasking proteins.** *Nucleic Acids Research* 2014, 42: D517-D520.

84. Mani M, Chen C, Amblee V, Liu H, Mathur T, Zwicke G *et al*.: **MoonProt: A database for proteins that are known to moonlight.** *Nucleic Acids Research* 2015, 43: D277:D282.

85. Ashburner M, Ball C, Blake J, Botstein D, Butler H, Cherry J *et al*.: **Gene ontology: Tool for the unification of biology. The Gene Ontology Consortium.** *Nature Genetics* 2000, 25: 25-34.

86. UniProt Consortium.: **Activities at the Universal Protein Resource (UniProt).** *Nucleic Acids Research* 2014, 42: D191-D198.

87. Jeffery CJ: **Molecular mechanisms for multitasking: Recent crystal structures of moonlighting proteins.** *Current Opinion in Structural Biology* 2004, 14: 663-668.

88. Spiess C, Beil A, Ehrmann M.: **A temperature-dependent switch from chaperone to protease in a widely conserved heat shock protein.** *Cell* 1999, 97: 339-347.

89. Lipinska B, Zylicz M, Georgopoulos C.: **The HtrA (DegP) protein, essential for Escherichia coli survival at high temperatures, is an endopeptidase.** *Journal of Bacteriology* 1990, 172: 1791-1797.

90. Pellegrini M, Marcotte EM, Thompson MJ, Eisenberg D, Yeates TO: **Assigning protein functions by comparative genome analysis: Protein phylogenetic profiles.** *Proceedings of the Nationnal Academy of Sciences of USA* 1999, 96: 4285-4288.

91. Babu M, Di´az-Meji JJ, Vlasblom J, Gagarinova A, Phanse S, Graham C *et al*.: **Genetic Interaction Maps in Escherichia coli Reveal Functional Crosstalk among Cell Envelope Biogenesis Pathways.** *PLoS Genetics* 2011, **7**.

92. Hawkins T, Luban S, Kihara D: **Enhanced automated function prediction using distantly related sequences and contextual association by PFP.** *Protein Science* 2006, 15: 1550-1556.

93. Schlicker A, Domingues F, Rahnenführer J, Lengauer T: **A new measure for functional similarity of gene products based on Gene Ontology.** *BMC Bioinformatics* 2006, 7.

94. Qi Y, Noble W. **Protein interaction networks: Protein domain interaction and protein function prediction.** In *Handbook of Computational Statistics: Statistical Bioinformatics*. 2011. Springer-Verlag.

95. Vinayagam A, Zirin J, Roesel C, Hu Y, Yilmazel B, Samsonova AA *et al*.: **Integrating protein-protein interaction networks with phenotypes reveals signs of interactions.** *Nature Methods* 2014, 11: 94-99.

96. Szklarczyk D, Franceschini A, Wyder S, Forslund K, Heller D, Huerta-Cepas J *et al*.: **STRING v10: Protein-protein interaction networks, integrated over the tree of life.** *Nucleic Acids Research* 2015, 43:D447-D452.

97. Cascalho M, Wong J, Steinberg C, Wabl M.: **Mismatch repair co-opted by hypermutation.** *Science* 1998, 279: 1207-1210.

98. Phung QH, Winter DB, Alrefai R, Gearhart PJ: **Hypermutation in Ig V genes from mice deficient in the MLH1 mismatch repair protein.** *Journal of Immunology* 1999, 162: 3121-3124.

99. Hu S, Xie Z, Onishi A, Yu X, Jiang L, Lin J *et al*.: **Profiling the human protein-DNA interactome reveals ERK2 as a transcriptional repressor of interferon signaling.** *Cell* 2009, 139: 610-622.

100. Meysman P, Sonego P, Bianco L, Fu Q, Ledezma-Tejeida D, Gama-Castro S *et al*.: **COLOMBOS v2.0: An ever expanding collection of bacterial expression compendia.** *Nucleic Acids Research* 2014, 42: D649-D653.

101. Mani R, St Onge RP, Hartman JL 4th, Giaever G, Roth FP.: **Defining genetic interaction.** *Proceedings of the National Academy of Sciences*2008, 105: 3461-3466.

102. Takeuchi R, Tamura T, Nakayashiki T, Tanaka Y, Muto A: **Colony-live - A high-throughput method for measuring microbial colony growth kinetics - Reveals diverse growth effects of gene knockouts in Escherichia coli.** *BMC Microbiology* 2014, 14: 171.

103. Typas A, Nichols RJ, Siegele DA, Shales M, Collins SR, Lim B *et al*.: **High-throughput, quantitative analyses of genetic interactions in E. coli.** *Nature Methods* 2008, 5: 781-787.

104. Butland G, Babu M, Díaz-Mejía JJ, Bohdana F, Phanse S, Gold B *et al*.: **eSGA: E. coli synthetic genetic array analysis.** *Nature Methods* 2008, 5: 789-795.

105. Baryshnikova A, Costanzo M, Kim Y, Ding H, Koh J, Toufighi K *et al*.: **Quantitative analysis of fitness and genetic interactions in yeast on a genome scale.** *Nature Methods* 2010, 7: 1017-1024.

106. Dunker AK, Brown CJ, Lawson JD, Iakoucheva LM, Obradovic' Z.: **Intrinsic disorder and protein function.** *Biochemistry* 2002, 41: 6573-6582.

107. Tompa P, Szász C, Buday L: **Structural disorder throws new light on moonlighting.** *Trends in Biochemical Sciences* 2005, 484-489.

108. Oates ME, Romero P, Ishida T, Ghalwash M, Mizianty M, Xue B *et al*.: **D2P2: Database of disordered protein .** *Nucleic Acids Research* 2013, 41:D508-D516.

109. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H *et al*.: **The Protein Data Bank.** *Nucleic Acids Research* 2000, 28: 235-242.

110. Rose PW, Bi C, Bluhm WF, Christie CH, Dimitropoulos D, Dutta S *et al*.: **The RCSB Protein Data Bank: New resources for research and education.** *Nucleic Acids Research* 2013, 41: D475-D482.

111. Frey A, Kallio P: **Bacterial hemoglobins and flavohemoglobins: Versatile proteins and their impact on microbiology and biotechnology.** *FEMS Microbiology Reviews* 2003, 27: 525-545.

112. de Leeuw E, Graham B, Phillips G, ten Hagen-Jongman C, Oudega B, Luirink J: **Molecular characterization of Escherichia coli FtsE and FtsX.** *Molecular Microbiology* 1999, 31: 983-993.

113. Chapple CE, Robisson B, Spinelli L, Guien C, Becker E, Brun C: **Extreme multifunctional proteins identified from a human protein interaction network.** *Nature Commnunications* 2015, 6.

114. Pritykin Y, Ghersi D, Singh M: **Genome-Wide Detection and Analysis of Multifunctional Genes.** *PLoS Comput Biol* 2015, 11: e1004467.

115. Okamura Y, Aoki Y, Obayashi T, Tadaka S, Ito S, Narise T *et al*. (Eds):**COXPRESdb in 2015: Coexpression database for animal species by DNA-microarray and RNAseq-based expression data with multiple quality assessment systems.** In *Nucleic Acids Research* 2014, 43: D82-D86.

116. Stark C, Breitkreutz BJ, Reguly T, Boucher L, Breitkreutz A, Tyers M. (Eds):**BioGRID: A general repository for interaction datasets.** In *Nucleic Acids Research* 2006, 34: D535-D539.

117. Little R.J.A., Rubin DB: **Statistical Analysis with Missing Data**. Wiley; 1987.

118. Zloba E: **Statistical methods of reproducing of missing data.** *Journal of Computer Modelling & New Technologies* 2002, 6: 51-61.

119. Morin RL, Raeside DE: **A reappraisal of distance-weighted k-nearest neighbor classification for pattern recognition with missing data.** *IEEE Transactions on Systems, Man and Cybernetics* 1981, 241-243.

120. Zhang S: **Parimputation: From imputation and null-imputation to partially imputation.** *IEEE Intelligent Informatics Bulletin* 2008, 9: 32-38.

121. A.Liaw. **Missing Value Imputations by randomForest**, R Documentation. 2003.

122. Breiman L: **Random Forests.** *Machine Learning* 2001, 45: 5-32.

123. Horn HF, Vousden KH: **Cooperation between the ribosomal proteins L5 and L11 in the p53 pathway.** *Oncogene* 2008, 27: 5774-5784.

124. Varma D, Chandrasekaran S, Sundin LJ, Reidy KT, Wan X, Chasse DA *et al*.: **Recruitment of the human Cdt1 replication licensing protein by the loop domain of Hec1 is required for stable kinetochore-microtubule attachment.** *Nature Cell Biololgy* 2012, 14: 593-603.

125. M Andrew, N Kamal: **A comparison of event models for Naive Bayes text classification.** *AAAI-98 workshop on learning for text categorization* 1998, 752: 41-48.

126. Cortes C, Vapnik V: **Support-vector network.** *Machine Learning* 1995, 20: 273-297.

127. Caruana R, Karampatziakis N, Yessenalina A: **An Empirical Evaluation of Supervised Learning in High Dimensions.** *Proceedings of the 25th international conference on Machine learning* 2008, 96-103.

128. Jeffery CJ: **Moonlighting proteins: Old proteins learning new tricks.** *Trends in Genetics* 2003, 19: 415-417.

129. Wan F, Anderson DE, Barnitz RA, Snow A, Bidere N, Zheng L *et al*.: **Ribosomal protein S3: A KH domain subunit in NF-kappaB complexes that mediates selective gene regulation.** *Cell* 2007, 131: 927-939.

130. Sampath P, Mazumder B, Seshadri V, Gerber CA, Chavatte L, Kinter M *et al*.: **Noncanonical function of glutamyl-prolyl-tRNA synthetase: Gene-specific silencing of translation.** *Cell* 2004, 119: 195-208.

131. Prunotto M, Farina A, Lane L, Pernin A, Schifferli J, Hochstrasser DF *et al*.: **Proteomic analysis of podocyte exosome-enriched fraction from normal human urine.** *Journal of Proteomics* 2013, 82: 193-229.

132. Manning CD, Raghavan P, Schütze H: **Introduction to information retrieval,** 1 edn. Cambridge: Cambridge University press; 2008.

133. Joachims T: **Text categorization with support vector machines: Learning with many relevant features.** *ECML '98 Proceedings of the 10th European Conference on Machine Learning* 1998:137-142.

134. Hoffman M, Bach F.R., Blei D.M.: **Online learning for latent Dirichlet allocation.** 2010:856-864.

135. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J: **Distributed representations of words and phrases and their compositionality.** *Advances in Neural Information Processing Systems* 2013:3111-3119.

136. Socher R, Huang EH, Pennington J, Manning CD, Ng A.: **Dynamic pooling and unfolding recursive autoencoders for paraphrase detection.** *Advances in Neural Information Processing Systems* 2011:801-809.

137. Ciresan D., Meier U., Schmidhuber J: **Multi-column deep neural networks for image classification.** *IEEE conference on Computer Vision and Pattern Recognition* 2012:3642-3649.

138. Le QV, Mikolov T: **Distributed representations of sentences and documents.** *arXiv preprint* 2014, 1405.4053.

139. Bird S: **NLTK: The natural language toolkit.** *Proceedings of the COLING/ACL on Interactive presentation sessions,* 2006:69-72.

140. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O *et al.*: **Scikit-learn: Machine learning in Python.** *The Journal of Machine Learning Research* 2011, 12: 2825-2830.

141. Blei DM: **Probabilistic topic models.** *Communications of the ACM* 2012, 55: 77-84.

142. Papadimitriou CH, Tamaki H, Raghavan P, Vempala S: **Latent semantic indexing: A probabilistic analysis.** 1998:159-168.

143. Rurek R, Sojka P: **Software Framework for Topic Modelling with Large Corpora.** 2010:45-50.

144. Wool IG: **Extraribosomal functions of ribosomal proteins.** *Trends in Biochemical Sciences* 1996, 21: 164-165.

145. Malygin AA, Parakhnevitch NM, Ivanov AV, Eperon IC, Karpova GG: **Human ribosomal protein S13 regulates expression of its own gene at the splicing step by a feedback mechanism.** *Nucleic Acids Research* 2007, 35: 6414-6423.

146. Low PS, Rathinavelu P, Harrison ML: **Regulation of glycolysis via reversible enzyme binding to the membrane protein, band 3.** *J Biol Chem* 1993, 268: 14627-14631.

147. Scheerer P, Borchert A, Krauss N, Wessner H, Gerth C, Hhne W *et al.*: **Structural basis for catalytic activity and enzyme polymerization of phospholipid hydroperoxide glutathione peroxidase-4 (GPx4).** *Biochemistry* 2007, 46: 9041-9049.

148. Stallmeyer B, Schwarz G, Schulze J, Nerlich A, Reiss J, Kirsch J *et al.*: **The neurotransmitter receptor-anchoring protein gephyrin reconstitutes molybdenum cofactor biosynthesis in bacteria, plants, and mammalian cells.** *Proc Natl Acad Sci U S A* 1999, 96: 1333-1338.

149. Bakir-Gungor B, Sezerman OU: **A new methodology to associate SNPs with human diseases according to their pathway related context.** *PLoS ONE* 2011, 6: e26277.

150. Wang B, Mezlini AM, Demir F, Fiume M, Tu Z, Brudno M *et al.*: **Similarity network fusion for aggregating data types on a genomic scale.** *Nature Methods* 2014, 11: 333-337.

151. Frey BJ, Dueck D: **Clustering by passing messages between data points.** *Science* 2007, 315: 972-976.

152. Tang M, Tan KM, Tan XL, Sael L, Chitale M, Esquivel-Rodriguez J et al.: **Graphical models for protein function and structure predictions.** *Handbook of Biological Knowledge Discovery*. Edited by Elloumi M , omaya A. Wiley; 2013:191-222.

153. **The Gene Ontology in 2010: Extensions and refinements.** *Nucleic Acids Research* 2010, 38: D331-D335.

154. Davis MJ, Sehgal MS, Ragan MA: **Automatic, context-specific generation of Gene Ontology slims.** *BMC Bioinformatics* 2010, 11.

155. Deng M, Zhang K, Mehta S, Chen T, Sun F: **Prediction of protein function using protein-protein interaction data**. *Journal of Computational Biology* 2003, 10: 947-960.

156. Kourmpetis Y.A.I, van Dijk A.D.J, Bink M.C.A.M., van Ham R.C.H.J., ter Braak C.J.F.: **Bayesian Markov random field analysis for protein function prediction based on network data.** *PLoS ONE* 2010, 5: 9293.

157. Deng M, Chen T, Sun F.: **An integrated probabilistic model for functional prediction of proteins. Journal of Computational Biology.** *Journal of Computational Biology* 2004, 11: 463-475.

158. Deng M, Tu Z, Sun F, Chen T: **Mapping gene ontology to proteins based on protein-protein interaction data.** *Bioinformatics* 2004, 20: 895-902.

159. Gehrmann T, Loog M, Reinders M.J.T., Ridder D.: **Conditional Random Fields for Protein Function Prediction.** *Pattern Recognition in Bioinformatics* 2013, 7986: 184-195.

160. Moult J, Hubbard T, Fidelis K, Pedersen JT: **Critical assessment of methods of protein structure prediction (CASP): Round III.** *Proteins: Structure, Function, and Bioinformatics* 1999, 37: 2-6.

161. Janin J: **Protein-protein docking tested in blind predictions: The CAPRI experiment.** *Mol BioSyst* 2010, 6: 2351-2362.

162. Friedberg I, Jambon M, Godzik A: **New avenues in protein function prediction.** *Protein Science* 2006, 15: 1527-1529.

163. Rodrigues A, Grant B, Godzik A, Friedberg I: **The 2006 automated function prediction meeting.** *BMC Bioinformatics* 2007, 8: S1.

164. Soro S, Tramontano A: **The prediction of protein function at CASP6.** *Proteins: Structure, Function, and Bioinformatics* 2005, 61: 201-213.

165. Lopez G, Rojas A, Tress M, Valencia A: **Assessment of predictions submitted for the CASP7 function prediction category.** *Proteins: Structure, Function, and Bioinformatics* 2007, 69: 165-174.

166. **The Gene Ontology in 2010: Extensions and refinements.** *Nucleic Acids Reseach* 2010, 38: D331-D335.

167. López G, Rojas A, Tress M, Valencia A.: **Assessment of predictions submitted for the CASP7 function prediction category.** *Proteins* 2007, 69: 165-174.

168. Chitale M, Hawkins T, Park C, Kihara D: **ESG: Extended similarity group method for automated protein function prediction.** *Bioinformatics* 2009, 25: 1739-1745.

169. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ: **Basic local alignment search tool.** *Journal of molecular biology* 1990, 215: 403-410.

170. Lobley AE, Nugent T, Orengo CA, Jones DT: **FFPred: An integrated feature-based function prediction server for vertebrate proteomes.** *Nucleic Acids Res* 2008, 36: W297-W302.

171. Remmert M, Biegert A, Hauser A, Söding J.: **HHblits: Lightning-fast iterative protein sequence searching by HMM-HMM alignment.** *Nature Methods* 2011, 9: 173-175.

172. Minneci F, Piovesan D, Cozzetto D, Jones DT.: **FFPred 2.0: Improved homology-independent prediction of gene ontology terms for eukaryotic protein sequences.** *PLoS ONE* 2013, 8: e63754.

173. Lobley A, Swindells MB, Orengo CA, Jones DT: **Inferring function using patterns of native disorder in proteins.** *PLoS computational biology* 2007, 3: e162.

174. Joachims T: **Making large-scale support vector machine learning practical.** In *Advances in kernel methods*. MIT Press; 1999:169-184.

175. Breazeale S, Ribeiro A, McClerren A, Raetz C (Eds): **A formyltransferase required for polymyxin resistance in Escherichia coli and the modification of lipid A with 4-Amino-4-deoxy-L-arabinose. Identification and function of UDP-4-deoxy-4-formamido-L-arabinose.** In *The Journal of Biological Chemistry* 2005, 280: 14154-14167.

176. Agrawal R, Srikant R: **Fast Algorithms for Mining Association Rules in Large Databases.** *Proceedings of the 20th International Conference on Very Large Data* 1994, 487-499.

177. Tao F, Murtagh F, Farid M: **Weighted association rule mining using weighted support and significance framework.** *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 2003, 661-666.

178. Wu CH, Nikolskaya A, Huang H, Yeh LS, Natale DA, Vinayaka CR *et al*.: **PIRSF: Family classification system at the Protein Information Resource.** *Nucleic Acids Research* 2004, 32: D112-D114.

179. Joshi-Tope G, Gillespie M, Vastrik I, D'Eustachio P, Schmidt E, de Bono B *et al*.: **Reactome: A knowledgebase of biological pathways.** *Nucleic Acids Research* 2005, 33: D428-D432.

180. Blohm P, Frishman G, Smialowski P, Goebels F, Wachinger B, Ruepp A *et al*.: **Negatome 2.0: A database of non-interacting proteins derived by literature mining, manual annotation and protein structure analysis.** *Nucleic Acids Research* 2014, 42: D396-D400.

181. Youngs N, Penfold-Brown D, Bonneau R, Shasha D: **Negative example selection for protein function prediction: The NoGO database.** *PLoS Comput Biol* 2014, 10.

182. Galperin MY, Koonin EV: **Sources of systematic error in functional annotation of genomes: Domain rearrangement, non-orthologous gene displacement and operon disruption.** *Silico Biol* 1998, 1: 55-67.

183. Bamford VA, Andrews SC, Watson KA. **EfeB, the peroxidase component of the EfeUOB bacterial Fe(II) transport system, also shows novel removal of iron from heme.** Submitted to the PDB data bank . 2011.

184. Subedi K, Choi D, Kim I, Min B, Park C: **Hsp31 of Escherichia coli K-12 is glyoxalase III.** *Molecular Microbiology* 2011, 81: 926-936.

185. Foti J, Persky N, Ferullo D, Lovett S: **Chromosome segregation control by Escherichia coli ObgE GTPase.** *Molecular Microbiology* 2007, 65: 569-581.

186. Jiang M, Datta K, Walker A, Strahler J, Bagamasbad P, Andrews P *et al*.: **The Escherichia coli GTPase CgtAE is involved in late steps of large ribosome assembly.** *The Journal of Bacteriology* 2006, 188: 6757-6770.

187. Cohen-Ben-Lulu G, Francis NR SE, Noy D DY, Prasad K, Sagi Y, Cecchini G *et al*.: **The bacterial flagellar switch complex is getting more complex.** *The EMBO Journal* 2008, 27: 1134-1144.

188. Awano N, Rajagopal V, Arbing M, Patel S, Hunt J, Inouye M *et al*. (Eds):**Escherichia coli RNase R has dual activities, helicase and RNase.** *The Journal of Bacteriology* 2010, 192: 1344-1352.

189. Stirling C, Colloms S, Collins J, Szatmari G, Sherratt DJ: **xerB, an Escherichia coli gene required for plasmid ColE1 site-specific recombination, is identical to pepA, encoding aminopeptidase A, a protein with substantial similarity to bovine lens leucine aminopeptidase.** *The EMBO Journal* 1989, 8: 1623-1627.

190. Skórko-Glonek J, Zurawa D, Kuczwara E, Wozniak M, Wypych Z, Lipinska B: **The Escherichia coli heat shock protease HtrA participates in defense against oxidative stress.** *Molecular and General Genetics : MGG* 1999, 262: 342-350.

191. Khil P, Camerini-Otero RD (Eds):**Over 1000 genes are involved in the DNA damage response of Escherichia coli.** *Molecular Microbiology* 2002, 44: 89-105.

192. Cusa E, Obradors N, Baldomà L, Badía J, Aguilar J (Eds):**Genetic analysis of a chromosomal region containing genes required for assimilation of allantoin nitrogen and linked glyoxylate metabolism in Escherichia coli.** *The Journal of Bacteriology* 1999, 181: 7479-7484.

193. Henning U, Sonntag I, Hindennach I: **Mutants (ompA) affecting a major outer membrane protein of Escherichia coli K12.** *European Journal of Biochemistry* 1978, 92: 491-498.

194. Kurono N, Matsuda A, Etchuya R, Sobue R, Sasaki Y, Ito M *et al*. (Eds):**Genome-wide screening of Escherichia coli genes involved in execution and promotion of cell-to-cell transfer of non-conjugative plasmids: RodZ (yfgA) is essential for plasmid acceptance in recipient cells.** *Biochemical and biophicial research communication* 2012, 421: 119-123.

195. Han X, Dorsey-Oresto A, Wang JY, Malik M, Drlica K, Zhao X *et al*. (Eds):**Escherichia coli genes that reduce the lethal effects of stress.** *BMC Microbiology* 2010, 10.

196. Arenas F, Díaz W, Díaz W, Pérez-Donoso J, Imlay J, Vásquez C: **The Escherichia coli btuE gene, encodes a glutathione peroxidase that is induced under oxidative stress conditions.** *Biochemical and biophicial research communication* 2010, 398: 690-694.

197. de Veaux L, Clevenson D, Bradbeer C, Kadner R: **Identification of the btuCED polypeptides and evidence for their role in vitamin B12 transport in Escherichia coli.** *The Journal of Bacteriology* 1986, 167: 920-927.

198. Seok Y, Sondej M, Badawi P, Lewis M, Briggs M, Jaffe H *et al*.: **High affinity binding and allosteric regulation of Escherichia coli glycogen phosphorylase by the histidine phosphocarrier protein, HPr.** *The Journal of Biological Chemistry* 1997, 272: 26511-26521.

199. Poole R, Hughes M: **New functions for the ancient globin family: Bacterial responses to nitric oxide and nitrosative stress.** *Molecular Microbiology* 2000, 36: 775-783.

200. Iwamoto A, Osawa A, Kawai M, Honda H, Yoshida S, Furuya N *et al*. (Eds):**Mutations in the essential Escherichia coli gene, yqgF, and their effects on transcription.** In *The journal of Molecular Microbiology and Biotechnology* 2012, 22: 17-23.

201. Aravind L, Makarova K, Koonin E (Eds):**SURVEY AND SUMMARY: Holliday junction resolvases and related nucleases: Identification of new families, phyletic distribution and evolutionary trajectories.** In *Nucleic Acids Research* 2000, 28: 3417-3420.

202. Palchevskiy V, Finkel S (Eds):**Escherichia coli competence gene homologs are essential for competitive fitness and the use of DNA as a nutrient.** *The Journal of Bacteriology* 2006, 188: 3902-3910.

203. Gong S, Ma Z, Foster J: **The Era-like GTPase TrmE conditionally activates gadE and glutamate-dependent acid resistance in Escherichia coli.** *Molecular Microbiology* 2004, 54: 948-961.

204. Inoue T, Shingaki R, Hirose S, Waki K, Fukui K, Mori H (Eds):**Genome-wide screening of genes required for swarming motility in Escherichia coli K-12.** *The Journal of Bacteriology* 2007, 189: 950-957.

205. Zakin M, Duchange N, Ferrara P, Cohen G: **Nucleotide sequence of the metL gene of Escherichia coli. Its product, the bifunctional aspartokinase II-homoserine dehydrogenase II, and the bifunctional product of the thrA gene, aspartokinase I-homoserine dehydrogenase I, derive from a common ancestor.** *The Journal of Biological Chemistry* 1983, 258: 3028-3031.

206. D'Ari L, Rabinowitz J: **Purification, characterization, cloning, and amino acid sequence of the bifunctional enzyme 5,10-methylenetetrahydrofolate dehydrogenase/5,10-methenyltetrahydrofolate cyclohydrolase from Escherichia coli.** *The Journal of Biological Chemistry* 1991, 266: 23953-23958.

207. Clark D, Cronan JJ (Eds):**Acetaldehyde coenzyme A dehydrogenase of Escherichia coli.** *The Journal of Bacteriology* 1980, 144: 179-184.

208. Kessler D, Leibrecht I, Knappe J. **Pyruvate-formate-lyase-deactivase and acetyl-CoA reductase activities of Escherichia coli reside on a polymeric protein particle encoded by adhE.** *FEBS Letters* 281[1-2], 59-63. 1991.

209. E C K, J S P: **Tandem translation starts in the cheA locus of Escherichia coli.** *The Journal of Bacteriology* 2013, 173: 2116-2119.

210. Oosawa K, Hess J, Simon MI: **Mutants defective in bacterial chemotaxis show modified protein phosphorylation.** *Cell* 1998, 53: 89-96.

211. Kneidinger B, Marolda C, Graninger M, Zamyatina A, McArthur F, Kosma P *et al*.: **Biosynthesis pathway of ADP-L-glycero-beta-D-manno-heptose in Escherichia coli.** *Journal of Bacteriology* 2002, 184: 363-9.

212. Spencer J, Stolowich N, Roessner C, Scott A. **The Escherichia coli cysG gene encodes the multifunctional protein, siroheme synthase**. *FEBS Letters* 335[1], 57-60. 1993.

213. Korch S, Henderson T, Hill T (Eds):**Characterization of the hipA7 allele of Escherichia coli and evidence that high persistence is governed by (p)ppGpp synthesis.** *Molecular Microbiology* 2003, 50: 1199-1212.

214. Boehm A, Steiner S, Zaehringer F, Casanova A, Hamburger F, Ritz D *et al*. (Eds):**Second messenger signalling governs Escherichia coli biofilm induction upon ribosomal stress.** *Molecular Microbiology* 2009, 72: 1500-1516.

215. Raffaelli N, Lorenzi T, Mariani P, Amici A, Ruggieri S, Magni G.: **The Escherichia coli NadR regulator is endowed with nicotinamide mononucleotide adenylyltransferase activity.** *The Journal of Bacteriology* 1999, 181: 5509-5511.

216. Ouyang P: **Antibodies differentiate desmosome-form and nucleus-form pinin: Evidence that pinin is a moonlighting protein with dual location at the desmosome and within the nucleus.** *Biochemical and biophicial research communication* 1999, 263: 192-200.

217. Boonacker E, Van Noorden C: **The multifunctional or moonlighting protein CD26/DPPIV.** *European Journal of Cell Biology* 2003, 82: 53-73.

218. Haraguchi C, Mabuchi T, Hirata S, Shoda T, Yamada A, Hoshi K *et al*.: **Spatiotemporal changes of levels of a moonlighting protein, phospholipid hydroperoxide glutathione peroxidase, in subcellular compartments during spermatogenesis in the rat testis.** *Biology of Reproduction* 2003, 69: 885-895.

219. Montfort A, Martin P, Levade T, Benoist H, Ségui B: **FAN (factor associated with neutral sphingomyelinase activation), a moonlighting protein in TNF-R1 signaling.** *Journal of leukocyte biology* 2010, 88: 987-903.

220. Tunio S, Oldfield N, Berry A, Ala'Aldeen D, Wooldridge K, Turner D: **The moonlighting protein fructose-1, 6-bisphosphate aldolase of Neisseria meningitidis: Surface localization and role in host cell adhesion.** *Molecular Microbiology* 2010, 76: 605-615.

221. Vilardo E, Nachbagauer C, Buzet A, Taschner A, Holzmann J, Rossmanith W: **A subcomplex of human mitochondrial RNase P is a bifunctional methyltransferase - Extensive moonlighting in mitochondrial tRNA biogenesis.** *Nucleic Acids Research* 2012, 40: 11583-11593.

222. Urban C, Xiong X, Sohn K, Schröppel K, Brunner H, Rupp S: **The moonlighting protein Tsa1p is implicated in oxidative stress response and in cell wall biogenesis in Candida albicans.** *Molecular Microbiology* 2005, 57: 1318-1341.

223. Moreno J, Patlolla B, Belton K, Jenkins B, Radchenkova P, Piva MA. **Two independent activities define Ccm1p as a moonlighting protein in Saccharomyces cerevisiae.** *Bioscience Reports* 2012,32:549-557.

224. Herbert C, Labouesse M, Dujardin G, Slonimski P: **The NAM2 proteins from S. cerevisiae and S. douglasii are mitochondrial leucyl-tRNA synthetases, and are involved in mRNA splicing.** *The EMBO Journal* 1988, **7:** 473-483.

225. Guo M, Schimmel P: **Essential nontranslational functions of tRNA synthetases.** *Nature Chemical Biology* 2013, 9: 145-153.

226. Herzog W, Müller K, Huisken J, Stainier D: **Genetic evidence for a noncanonical function of seryl-tRNA synthetase in vascular development.** *Circulation Research* 2009, 104: 1260-1266.

227. Xu X, Shi Y, Zhang H, Swindell E, Marshall A, Guo M *et al*.: **Unique domain appended to vertebrate tRNA synthetase is essential for vascular development.** *Nature Commnunications* 2012, 3.

228. Ritterson Lew C, Tolan D: **Aldolase sequesters WASP and affects WASP/Arp2/3-stimulated actin dynamics.** *Jounal of cellular biochemistry* 2013, 114: 1928-1939.

APPENDICES

Appendix A More on Moonlighting Proteins

Table A1 Moonlighting proteins identified in *E. coli*

Protein-Name/UniProt ID/gene ID	First Function	Additional Functions	Class	Ref
b0118/P36683 /AcnB	Aconitate hydratase	Post-transcriptional regulation; mRNA binding	I	[77]
b1019/P31545 /EfeB	Peroxidase on guaiacol	Iron assimilation from heme; response to DNA damage stimulas	I	[183]
b1276/P25516 /AcnA	Aconitate hydratase	Post-transcriptional regulation; mRNA binding	I	[77]
b1967/P31658 /HchA	Molecular chaperone	Glyoxalase activity	I	[184]
b3183/P42641 /ObgE	GTPase	Role in ribosome bio-genesis	I	[185,186]
b4151/P0A8Q 3/FrdD	Membrane bound respiratory protein (anaerobic condition)	Role in bacterial flagellar switch (aerobic conditions)	I	[187]
b4152/P0A8Q 0/FrdC	Membrane bound respiratory protein (anaerobic condition)	Role in bacterial flagellar switch (aerobic conditions)	I	[187]
b4153/P0AC4 7/FrdB	Membrane bound respiratory protein (anaerobic condition)	Role in bacterial flagellar switch (aerobic conditions)	I	[187]

b4154/P00363/ FrdA	Membrane bound respiratory protein (anaerobic condition)	Role in bacterial flagellar switch (aerobic conditions)	I	[187]
b4179/P21499/ Rnr	Helicase	RNase	I	[188]
b4260/P68767/ PepA†b	Plasmid recombination	Peptide catabolic process; DNA binding/transcriptional control	I	[189]
b0161/P0C0V0/ DegP†	Chaperone	Proteolysis	II	[190]
b0509/P77161/ GlxR	Glyoxylate metabolism	Allantoin assimilation; DNA damage response	II	[191, 192]
b0957/P0A910/ OmpA	Transport	1. Viral entry 2.DNA damage response	II	[191, 193]
b1317/P77366/ YcjU	Carbohydrate metabolism	1. Cell-to-cell plasmid transfer 2. Reduce the lethal effects of stress.	II	[194, 195]
b1710/P06610/ BtuE	Glutathione peroxidase	Non-essential role in vitamin-B12 transport	II	[196, 197]
b2415/P0AA04/ PtsH	Phosphocarrier protein essential in sugar transport	Positive regulation of glycogen catabolism	II	[198]
b2552/P24232/ Hmp	(aerobic condition) Nitric oxide dioxygenase (NOD)	(anaerobic condition) Amplifier of superoxide stress, NO and FAD reductase	II	[111, 199]
b2949/P0A8I1/ YqgF	Putative Holliday junction resolvase	Transcription anti-termination	II	[200, 201]
b3414/P63020/ NfuA	Fe-S biogenesis	Necessary for the use of extracellular DNA as the sole source of carbon and energy	II	[202]

b3463/P0A9R7/ FtsE	Cell division	Salt transport by ABC-Transporter	II	[112]
b3706/P25522/ MnmE	tRNA modification	Regulating glutamate-dependent acid resistance	II	[203]
b0135/P31058/ YadC	Cell adhesion	Reduce lethal effects of stress	III	[195]
b0284/P77489/ YagR	Putative xanthine dehydrogenase	DNA damage response	III	[191]
b0543/P23895/ EmrE	Multidrug transporter	DNA damage response	III	[191]
b1018/P0AB24/ EfeO	Involved in Iron uptake	Response to lethal antimicrobial and environmental stress	III	[195]
b2037/P37746/ RfbX	Putative O-antigen transporter	DNA damage response	III	[191]
b2147/P25889/ PreA	Pyrimidine base degradation	Required for swarming Motility	III	[204]
b2290/P0A959/ AlaA	Involved in biosynthesis of alanine	Response to lethal antimicrobial and environmental stress	III	[195]
b3191/P64602/ MlaB	Phospholipid ABC transporter	Response to lethal antimicrobial and environmental stress	III	[195]
b3233/P0A9Q9/ Asd	Aspartate-semialdehyde dehydrogenase	DNA damage response	III	[191]
b4177/P0A7D4/ PurA	Adenylosuccinate synthetase	DNA damage response	III	[191]
b4383/P0A6K6/ DeoB	Phosphopentomutase	DNA damage response	III	[191]

Table A2 Multi-domain proteins with multiple functions in *E.coli*

gene ID /Protein Name/UniProt ID	First Function	Additional Functions	Ref.
b0002/P00561/ ThrA	Aspartokinase	Homoserine dehydrogenase	[205]
b0529/P24186/ FolD	Oxidation of methylene-tetrahydrofolate	Hydrolysis of methenyltetrahydrofolate	[206]
b1241/P0A9Q7 /AdhE	Alcohol dehydrogenase	Acetaldehyde dehydrogen-ase; Pyruvate-formate-lyase deactivase	[207,208]
b1888/P07363/ CheA	Chematoxis sensor kinase	Regulation of protein; dephosphorylation	[77,209, 210]
b2255/P77398/ ArnA	Oxidative decarboxylation of UDP-glucuronic acid	Formyltransferase	[175]
b3052/P76658/ HldE	D-beta-D-heptose 7-phosphate kinase	D-beta-D-heptose 1-phosphate adenosyltransferase	[211]
b3368/P0AEA 8/CysG	SAM-dependent methylation	NAD-dependent ring dehy-drogenation; Ferrorochelation	[212]
b3650/P0AG24 /SpoT	ppGpp synthase	ppGpp hydrolase	[213,214]
b3940/P00562/ MetL	Aspartokinase	Homoserine dehydrogenase	[205]
b4390/P27278/ NadR†	Transcriptional regulator	Nicotinamide mononucleo-tide adenylyltransferase; Ribosylnicotinamide kinase	[215]

Table A3 The MPR3 moonlighting protein dataset.

Uniprot ID /ProteinName	Organism	Primary Function	Secondary Function(s)	Ref
P79149/Pinin	Canis familiaris	Induce junction formation and enhance cell aggregation	Component of the RNP structure	[216]
P27487/DPP**4**	Homo sapiens	Serine protease	1. Cell surface glycoprotein receptor for CAV1 2. Co-stimulatory protein involving in T-cell receptor-mediated T-cell activation and proliferation. 3. Binding collagen and fibronectin 4. Involvement in apoptosis	[217]
Q91XR9/GPx-4	Mus musculus	Antioxidant of mature sperm	Structural protein of the mitochondrial capsule	[218]
O35242/FAN	Mus musculus	Apoptosis	Inflammatory signalling	[219]
E3D2R2/Fructose-1, 6-bisphosphate aldolase	Neisseria meningitidis	Glycolytic enzyme	Host-cell invasion	[220]
Q7L0Y3/ MRP1	Homo sapiens	tRNA methyltransferase	Dehydrogenase	[221]
Q9Y7F0/Peroxiredoxin TSA1	Candida albicans	Antioxidant against sulfur-	Involved in morphology	[222]

		containing radicals		
P48237/CCM1	Saccharomyces cerevisiae	Introns removal in mRNA maturation	Maintains the steady-state levels of the mitoribosome small subunit RNA	[223]
P11325/Nam2p	Saccharomyces cerevisiae	Mitochondrial leucyl-tRNA synthetase	Mitochondrial RNA splicing activity	[224]
Q9P2J5/LeuRS	Homo sapiens	tRNA synthetase	Translocation and activation of mTORC1 to lysosomal membrane	[225]
P47897/GlnRS	Homo sapiens	tRNA synthetase	Suppresses apoptotic acitivities	[225]
Q6DRC0/SerRS	Danio rerio	tRNA synthetase	Regulates development of closed circulatory system	[225-227]
P00883/Fructose-bisphosphate aldolase A	Oryctolagus cuniculus	Glycolytic enzyme	Regulation of cell mobility	[228]
P0A518/Cpn60-1	Mycobacterium tuberculosis	Prototypic molecular chaperone	Osteoclast-inhibitory action	[228]
P0A518/Cpn60-2	Mycobacterium tuberculosis	Prototypic molecular chaperone	Stimulates macrophage pro-inflammatory cytokine synthesis	[228]

Table A4 P-value from Kolmorov-Smirnov test for clustering profiles

For the clustering profiles shown as figures, the Kolmorov-Smirnov test was performed to examine if the number of clusters formed at specified cutoff is significantly different between a moonlighting protein dataset (MPR1, 2, 3, or the E. coli MP set) and E. coli non-moonlighting protein set. Refer to corresponding figure captions and text.

Dataset		MP sets compared with the *E. coli* non-MP set			
Description of data	Score Cutoff	MPR1	MPR2	MPR3	E. coli MP
Number of BP GO term clusters grouped with SS^{rel}	0.1	**< 0.05**	**< 0.05**	**< 0.05**	**< 0.05**
	0.5	**< 0.05**	**< 0.05**	**< 0.05**	**< 0.05**
	1.0	**< 0.05**	**< 0.05**	**< 0.05**	**< 0.05**
Number of MF GO term clusters grouped with SS^{rel}	0.1	**< 0.05**	**< 0.05**	0.37	0.10
	0.5	0.07	0.12	0.10	0.25
	1.0	**< 0.05**	**< 0.05**	0.09	**< 0.05**
Number of clusters of interacting proteins grouped with funsim	0.2	0.61	0.14	0.60	0.16
	0.6	0.96	0.93	**< 0.05**	**< 0.05**
	0.8	**< 0.05**	**< 0.05**	**< 0.05**	**< 0.05**
Number of clusters of interacting proteins grouped with BP-funsim	0.2	0.42	0.33	0.16	**< 0.05**
	0.6	0.89	0.69	**< 0.05**	**< 0.05**
	0.8	0.08	0.19	**< 0.05**	**< 0.05**
Number of clusters of coexpressed proteins grouped with funsim	0.2	-	-	-	0.83
	0.6	-	-	-	0.75
	0.8	-	-	-	0.38
Number of clusters of coexpressed proteins grouped with BP-funsim	0.2	-	-	-	0.82
	0.6	-	-	-	0.35
	0.8	-	-	-	0.17
Number of clusters of phylogenetically related proteins grouped with funsim (Fig. 8B)	0.2	0.07	0.59	0.26	0.27
	0.6	0.16	0.08	0.23	0.30
	0.8	0.15	0.45	**< 0.05**	0.08
Number of clusters of phylogenetically related proteins grouped with BP-funsim	0.2	0.07	0.70	0.47	0.65
	0.6	0.15	0.08	0.17	0.36
	0.8	0.11	**< 0.05**	**< 0.05**	0.29

A 1 Feature selection procedure of MPFit

Detail discussion of feature selection process in the protein-protein interaction (PPI) feature domain is given here. PPI data was extracted from the STRING database [96]. For each protein in the dataset of moonlighting and non-moonlighting proteins (MP and non-MP), we extracted PPI interactions that had sufficient confidence score (> 0.4) in STRING. 124 moonlighting proteins (46.3%) and 61 non-moonlighting proteins (37.7%) in the dataset had such PPI interactions in STRING. Next, we checked the functional divergence of interacting proteins. Interacting proteins for each MP or non-MP were clustered using GO term-based functional similarity. To quantify the functional similarity of two proteins, we used the funsim score [93]. Computation of funsim score is described in Methods section 2.4.1.2, Eqn 2.2-2.8.

Using this framework of GO-based functional similarity (Eqn. 2.5) between two proteins, we clustered the interacting proteins of each of the MPs and non-MPs in the dataset and created a clustering profile (Fig. S1). A clustering profile shows the number of clusters formed by using ten different cutoff values (from 0.1 to 1.0 with an interval of 0.1). For PPI network, we selected three different GO category combinations (Fig. S1). Using these three clustering profiles (Fig. S1A, S1B, S1C), we selected the number of protein clusters (y-axis) at 5 score thresholds each (0.1, 0.3, 0.5, 0.7, and 0.9 at the x-axis). This procedure constructs 15 features in total for each MPs and non-MPs in the PPI feature domain.

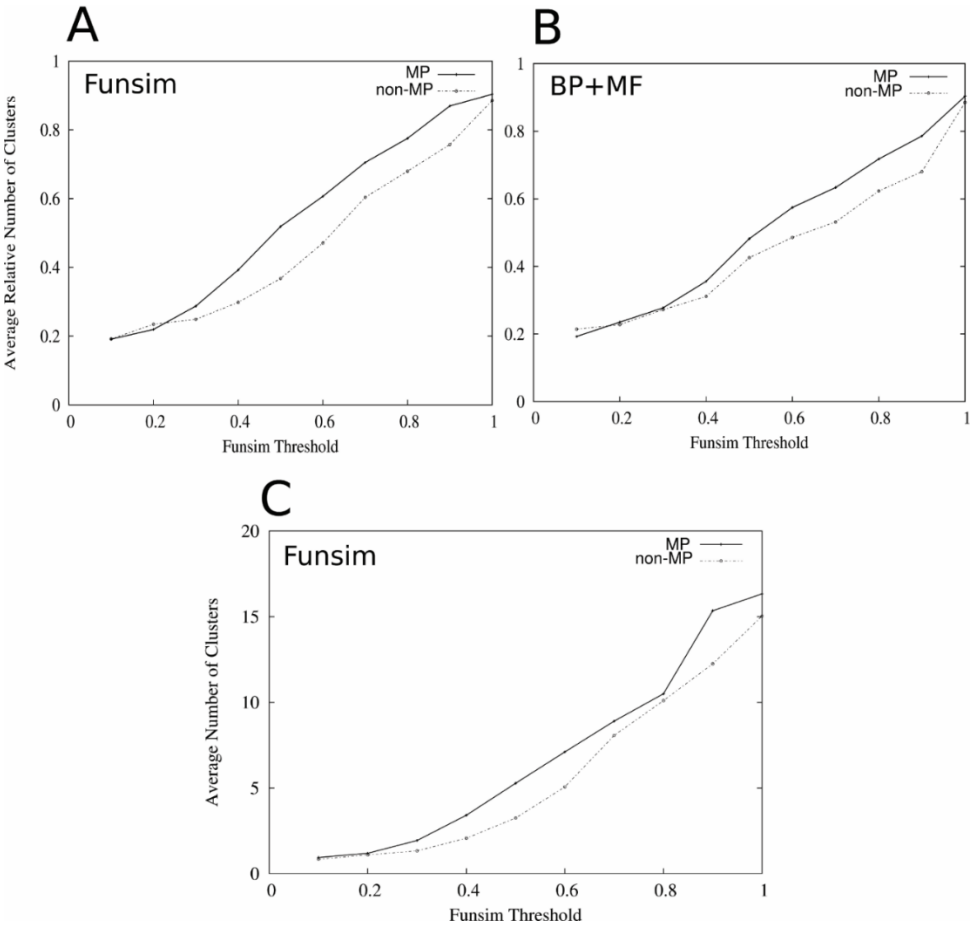

Figure A1 Clustering profiles of interacting proteins of MP and non-MP

Physically interacting proteins for a MP or a non-MP were clustered using 5 cutoff values of a functional similarity score. Single linkage clustering was used. ***(A-B)*** *the average number of clusters of interacting proteins relative to the number of interacting proteins. The funsim score with all three GO categories was used for A, and the funsim score with BP & MF GO term only in Eqn. 2.5 was used for B. C) the funsim score with all three GO categories was used. Note that the y-axis is the average number of clusters per interacting proteins in the PPI network, which is different from the value used in* ***(A).***

Similar feature selection procedure was used for the other four features, i.e., GE using the COEXPRESdb database [115], GI using the BioGRID database [116], Phylo from the STRING database [96], and GO from Uniprot [86] and Gene Ontology [43]. For the NET feature domain, three graph properties of proteins, namely, degree centrality, closeness centrality, and between-ness centrality, based on the PPI network were computed as fea-

tures. For the DOR feature domain using the D2P2 database [108], we computed three properties of protein's intrinsically disordered regions, namely, the number of disordered regions, the length of disordered regions, and the proportion of disordered regions in the sequence.

A 2 Performance of MPFit with random forest for GO and all omics-based feature combinations

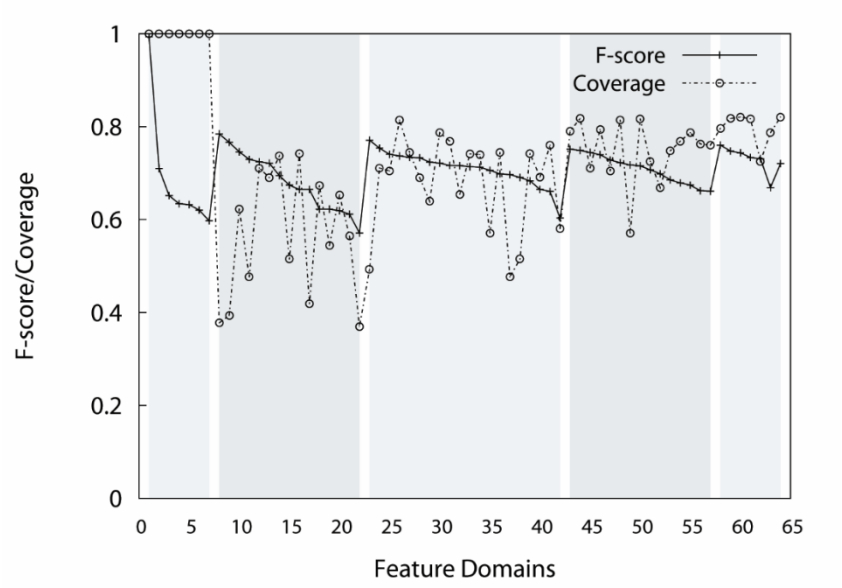

Figure A2 Performance of MPFit with random forest.

Results of 5-fold cross validation of MPFit with random forest classifier for the GO based features, and all possible feature combinations of the six omics-based features. Feature legends – GO: Gene Ontology, PPI: Protein-Protein Interactions, Phylo: Phylogenetic profile, GE: Gene Expression, DOR: DisOrdered Regions, GI: Genetic Interactions, NET: 3 graph properties – betweenness, degree centrality, closeness centrality. F-score computed as 2-class weighted average over MP/non-MP class. Coverage was computed as the mean protein coverage of MP/non-MP classes. For combinations with the same number of features, the results are sorted by their F-scores.

Numbers 1-64 shown on the x-axis represent the following feature combinations:

1:GO , 2:GE , 3:DOR , 4:Phylo , 5:GI , 6:PPI , 7:NET
8:Phylo+GI , 9:Phylo+NET , 10:Phylo+GE , 11:PPI+Phylo , 12:Phylo+DOR , 13:PPI+GE , 14:GE+DOR ,15:PPI+GI , 16:PPI+DOR , 17:PPI+NET , 18:DOR+NET , 19:GE+GI , 20:GI+DOR , 21:GE+NET , 22:GI+NET
23:Phylo+GI+NET , 24:PPI+Phylo+GE , 25:PPI+GE+GI , 26:PPI+GE+DOR , 27:GE+DOR+NET ,28:PPI+GE+NET , 29:Phylo+GE+GI , 30:Phylo+GE+DOR , 31:PPI+Phylo+DOR 32:Phylo+GE+NET , 33:Phylo+GI+DOR , 34:GE+GI+DOR , 35:PPI+Phylo+GI , 36:Phylo+DOR+NET , 37:PPI+Phylo+NET , 38:PPI+GI+NET , 39:PPI+DOR+NET , 40:GI+DOR+NET , 41:PPI+GI+DOR , 42:GE+GI+NET
43:Phylo+GE+GI+DOR , 44:PPI+Phylo+GE+DOR , 45:PPI+Phylo+GE+NET , 46:Phylo+GE+DOR+NET , 47:PPI+GE+GI+NET , 48:PPI+GE+DOR+NET , 49:PPI+Phylo+GI+NET 50:PPI+GE+GI+DOR , 51:PPI+Phylo+GE+GI , 52:Phylo+GE+GI+NET , 53:GE+GI+DOR+NET , 54:PPI+Phylo+DOR+NET , 55:PPI+Phylo+GI+DOR , 56:Phylo+GI+DOR+NET , 57:PPI+GI+DOR+NET
58:Phylo+GE+GI+DOR+NET , 59:PPI+Phylo+GE+DOR+NET , 60:PPI+Phylo+GE+GI+DOR , 61:PPI+GE+GI+DOR+NET , 62:PPI+Phylo+GE+GI+NET , 63:PPI+Phylo+GI+DOR+NET , 64:PPI+Phylo+GE+GI+DOR+NET

Note that the coverage generally increases as the number of used features increases because missing features were imputed for a protein that have at least one feature among a particular combination considered.

A 3 Performance of MPFit with random forest without imputation

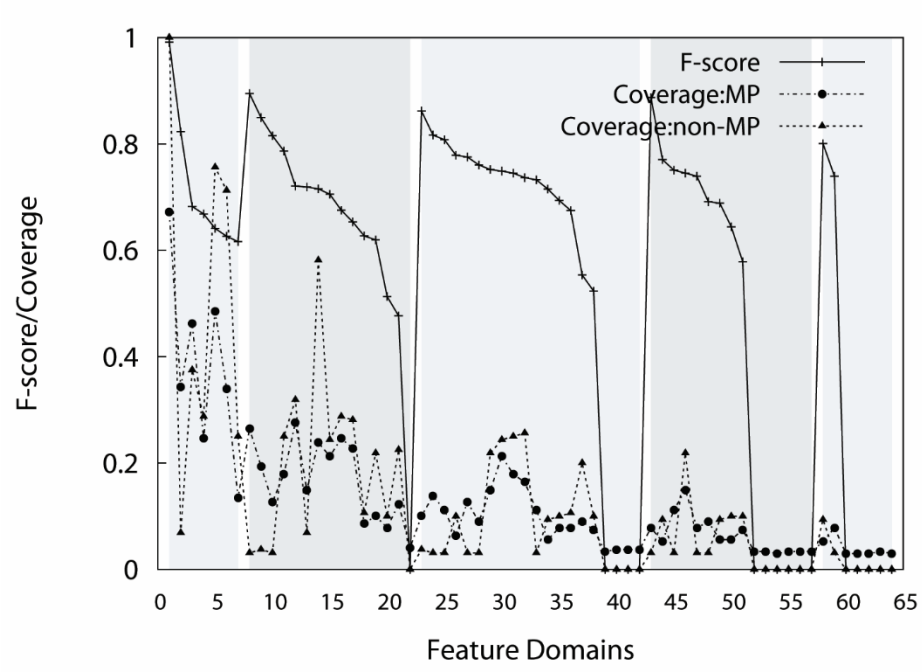

Figure A3 Performance of MPFit with RF without missing feature imputation.

Results of a five-fold cross validation were reported. Coverage is reported separately for the MP class (circles) and non-MP class (triangles). The feature combinations on the x-axis are the same as Fig. A.2:

1:GO , 2:Phylo , 3:PPI , 4:NET , 5:DOR , 6:GE , 7:GI
8:PPI+Phylo , 9:Phylo+DOR , 10:Phylo+NET , 11:DOR+NET , 12:PPI+DOR , 13:Phylo+GE , 14:GE+DOR , 15:GE+NET , 16:PPI+NET , 17:PPI+GE , 18:PPI+GI , 19:GI+DOR , 20:GI+NET , 21:GE+GI , 22:Phylo+GI
23:Phylo+GE+DOR , 24:PPI+Phylo+DOR , 25:Phylo+GE+NET , 26:PPI+GI+DOR , 27:PPI+Phylo+NET , 28:Phylo+DOR+NET , 29:GE+DOR+NET , 30:PPI+GE+NET , 31:PPI+DOR+NET , 32:PPI+GE+DOR , 33:PPI+Phylo+GE , 34:GI+DOR+NET , 35:PPI+GI+NET , 36:PPI+GE+GI , 37:GE+GI+DOR , 38:GE+GI+NET 39:Phylo+GI+DOR , 40:PPI+Phylo+GI , 41:Phylo+GI+NET , 42:Phylo+GE+GI
43:Phylo+GE+DOR+NET , 44:GE+GI+DOR+NET , 45:PPI+Phylo+GE+NET , 46:PPI+GE+DOR+NET , 47:PPI+Phylo+GE+DOR , 48:PPI+Phylo+DOR+NET , 49:PPI+GI+DOR+NET , 50:PPI+GE+GI+DOR , 51:PPI+GE+GI+NET , 52:PPI+Phylo+GI+DOR , 53:Phylo+GI+DOR+NET , 54:Phylo+GE+GI+NET , 55:PPI+Phylo+GI+NET , 56:Phylo+GE+GI+DOR , 57:PPI+Phylo+GE+GI

58:PPI+GE+GI+DOR+NET , 59:PPI+Phylo+GE+DOR+NET ,
60:PPI+Phylo+GE+GI+DOR , 61:Phylo+GE+GI+DOR+NET ,
62:PPI+Phylo+GE+GI+NET , 63:PPI+Phylo+GI+DOR+NET ,
64:PPI+Phylo+GE+GI+DOR+NET

Note that the coverages are low because no imputation was performed.

A 4 Random forest classifier with a probabilistic imputation

We also examined a different way of missing feature imputation. In the alternative approach, unlike filling missing features by voting using temporarily assigned feature values as described in Methods (termed "explicit imputation"), the splitting probabilities in random forest that were learned from the training data were used for imputation. The concrete pipeline of this so-called "probabilistic imputation" is as follows: first, we train the random forest with only those proteins that have non-missing features in a certain feature combination. In each branch of each decision tree in the random forest, a fraction is learned (and stored) from the training data that indicates what portion of the proteins in the training set was split with that branch. Then we run down each protein Pi in the test data through each tree in the trained random forest. Whenever Pi falls into a tree node that splits based on a feature which is missing in Pi, we split Pi using the branch probabilities associated with that node that we learned from the training data. Finally, a majority vote is taken for Pi counting the number of trees that classifies Pi in MP/non-MP class. Two slightly different ways of the probabilistic imputation were implemented. The first method takes a weighted majority vote of the trees that classifies a test protein Pi as MP/non-MP, where a weight for one tree Ti is the fraction that is learned from the train-

ing data for the leaf branch of Ti that leads to a MP/non-MP class for Pi (Random Forest Probabilistic Imputation, Weighted, RF-PI-W). The second method simply takes a non-weighted majority vote for the test data point Pi (RF-PI-NW, Random Forest Probabilistic Imputation, Not Weighted).

Fig. S4 shows that the explicit imputation overall outperforms the two probabilistic imputation methods. Indeed, the explicit imputation showed higher F-score for all the feature combinations except for two cases: The DOR+NET combination had a higher F-score with RF-PI-NW (difference is 0.0156) and DOR had a higher F-score with RF-PI-W than the explicit imputation (difference 0.0139). Comparing the two probabilistic imputation methods, the non-weighted version (RF-PI-NW) showed a higher F-Score than its weighted counterpart (RF-PI-W) in 38 out of 64 (59.38%) feature combinations.

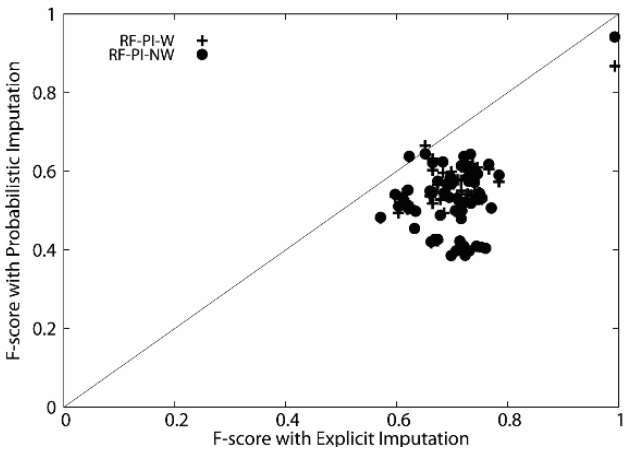

Figure A4 Performance comparison of explicit and probabilistic imputation.

The former is described in Methods. Values shown are the weighted class average F-score over fivefold cross validation. RF-PI-W: Random Forest Probabilistic Imputation, Weighted; RF-PI-NW: Random Forest Probabilistic Imputation, Not Weighted. See text for details.

The reason why the explicit imputation worked better than the probabilistic imputation would be because the latter performs training on only a small the portion of the dataset that have no-missing features for a certain feature combination. For example, for a combination of all six omics features, PPI+Phylo+GE+GI+DOR+NET, there are only eight proteins with no missing features that could be used for training the probabilistic imputation. This lack of sufficient training data resulted in poor F-scores for MPFit with probabilistic imputation (0.409 for both RF-PI-NW and RF-PI-W), which contrasted with the good performance exhibited by MPFit with explicit imputation (F-score: 0.721)

A 5 DextMP additional Data

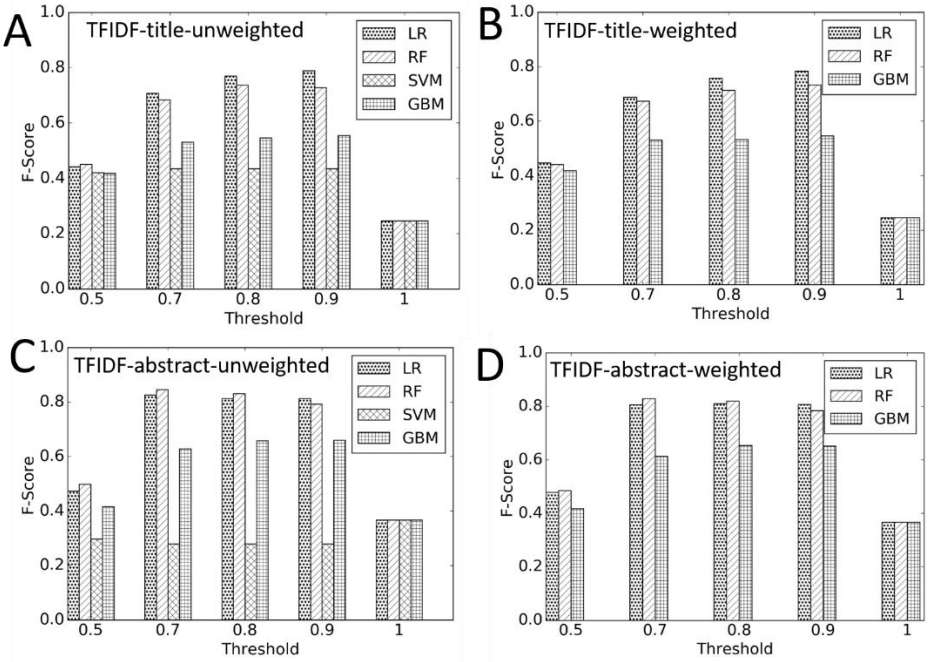

Figure A5 DextMP parameter tuning for TFIDF

5-fold cross validation F-score for protein-level MP prediction for different majority vote cut-offs with TFIDF language model.

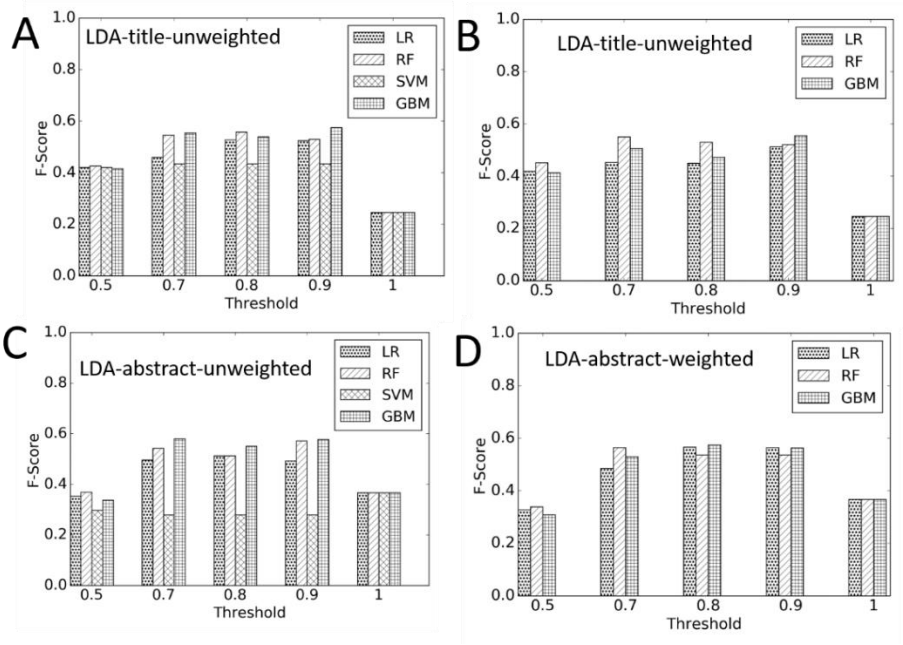

Figure A6 DextMP parameter tuning for LDA

5-fold cross validation F-score for protein-level MP prediction for different majority vote cut-offs with LDA language model.

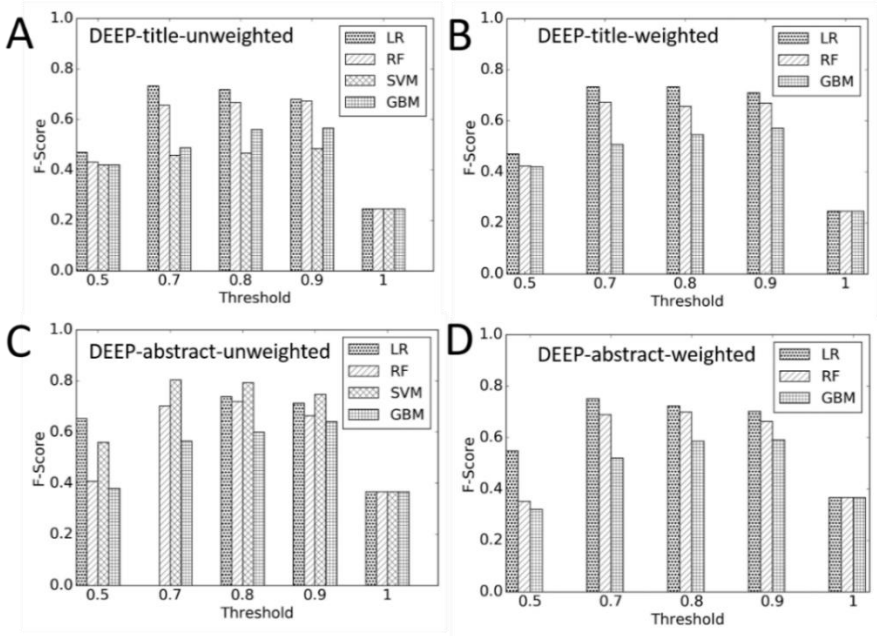

Figure A7 DextMP parameter tuning for DEEP

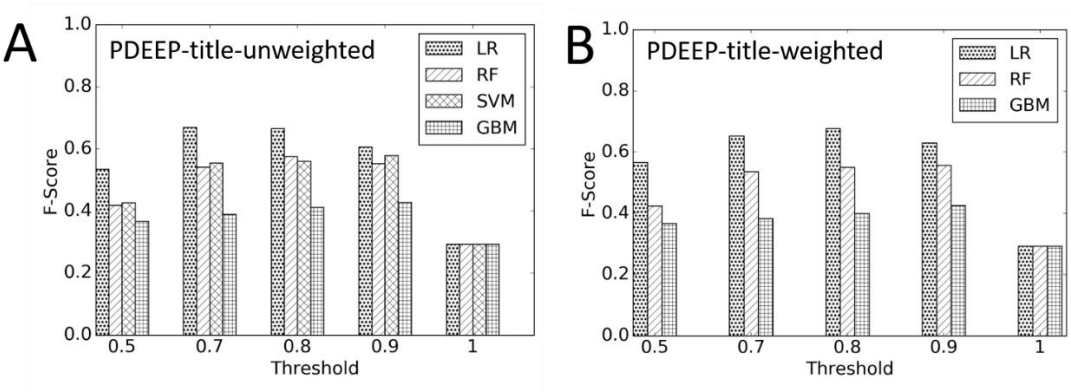

Figure A8 DextMP parameter tuning for PDEEP

5-fold cross validation F-score for protein-level MP prediction for different majority vote cut-offs with PDEEP language model.

Table A5 Selected optimal parameters for DEEP and LDA for different classifiers

	LR	**RF**	**SVM**	**GBM**
LDA-title	# of topics = 50	# of topics = 70	0 # of topics = 60	# of topics = 50
DEEP-title	min_count = 5 window = 3 size = 120	min_count = 5 window = 3 size = 140	min_count = 5 window = 2 size = 80	min_count = 5 window = 2 size = 140
LDA-abstract	# of topics = 70	# of topics = 50	0 # of topics = 20	# of topics = 70
DEEP-abstract	min_count = 3 window = 4 size = 180	min_count = 3 window = 2 size = 20	min_count = 5 window = 7 size = 20	min_count = 2 window = 5 size = 20
LDA-function	# of topics = 70	# of topics = 80	0 # of topics = 10	# of topics = 50
DEEP-function	min_count = 3 window = 8 size = 180	min_count = 4 window = 4 size = 100	min_count = 1 window = 6 size = 20	min_count = 1 window = 2 size = 40

Appendix B More on Group Function Prediction

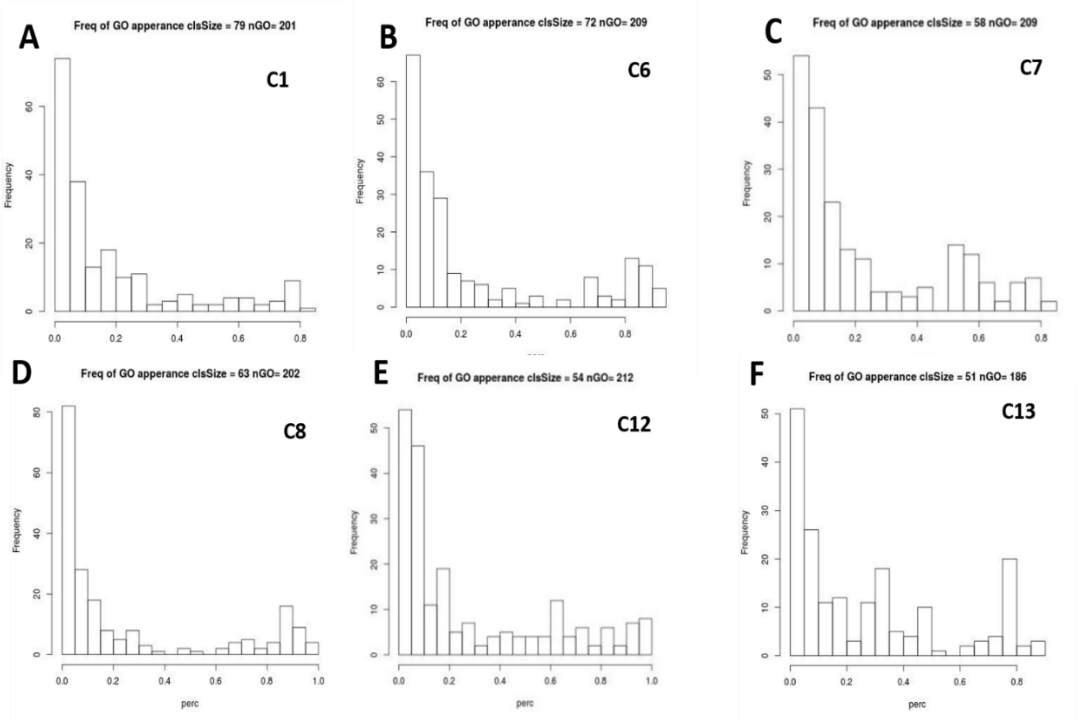

Figure B1 Six human PPI cluster selection for CRF validation

A human protein-protein interaction network of 6124 human proteins that are involved in 112,895 interactions are clustered and out of 16 clusters that had at least 50 member proteins, 6 clusters are selected for Fig. 3-4 that have a non-zero fraction of GO term distributions in the annotations of the proteins in the cluster.

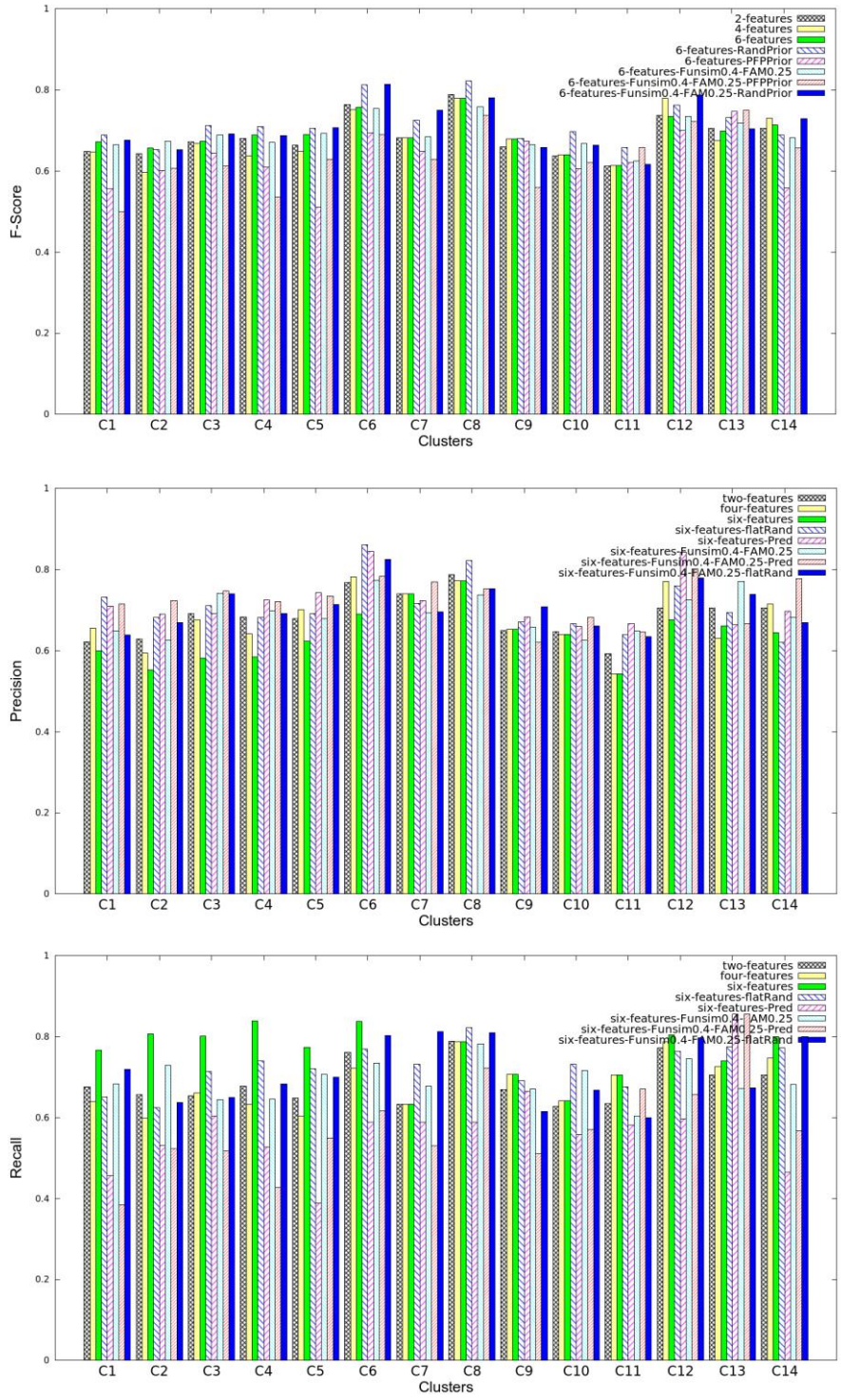

Figure B2 CRF cross validation for 14 Human PPI clusters.

Top: average F-score, Middle: average precision and Bottom: average Recall

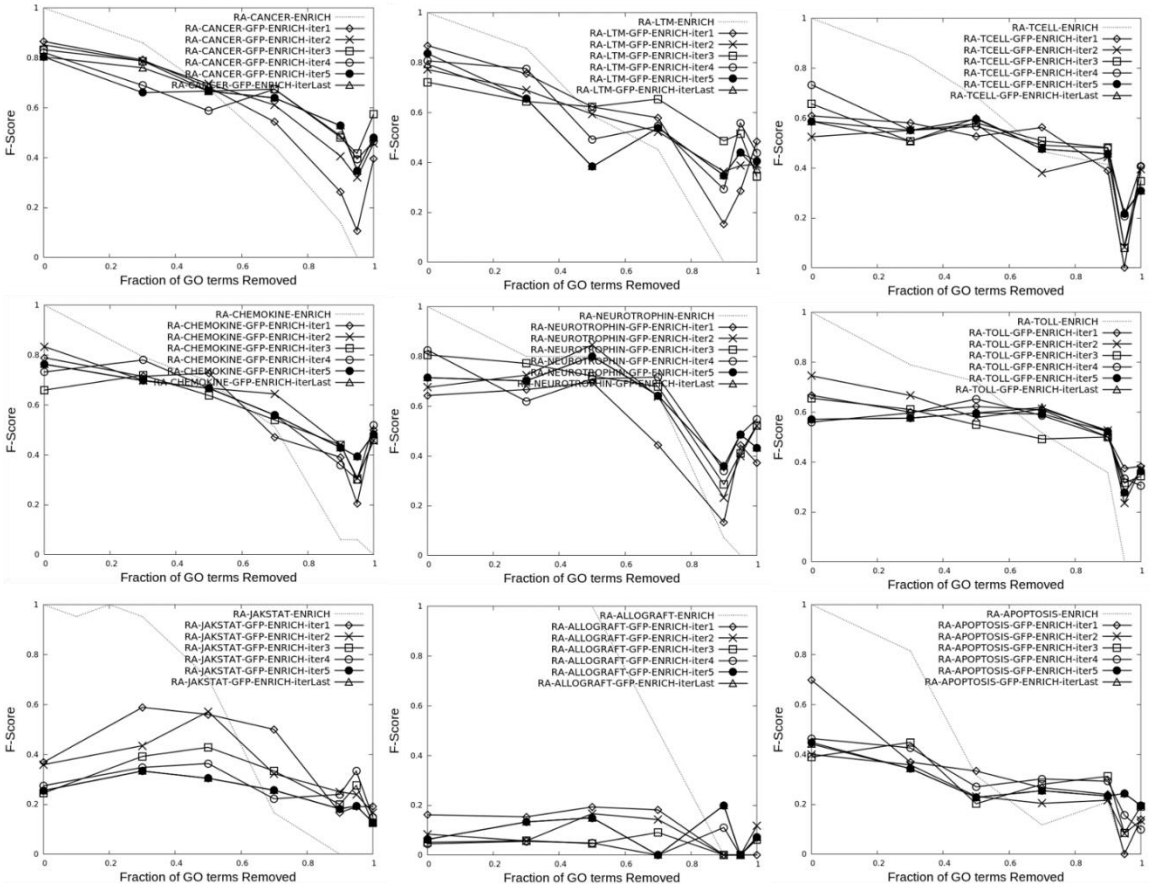

Figure B3 GFP f-score of GO removal simulations

Group function prediction to 9 groups of proteins. F-score of prediction was reported after removing a fraction of GO terms.

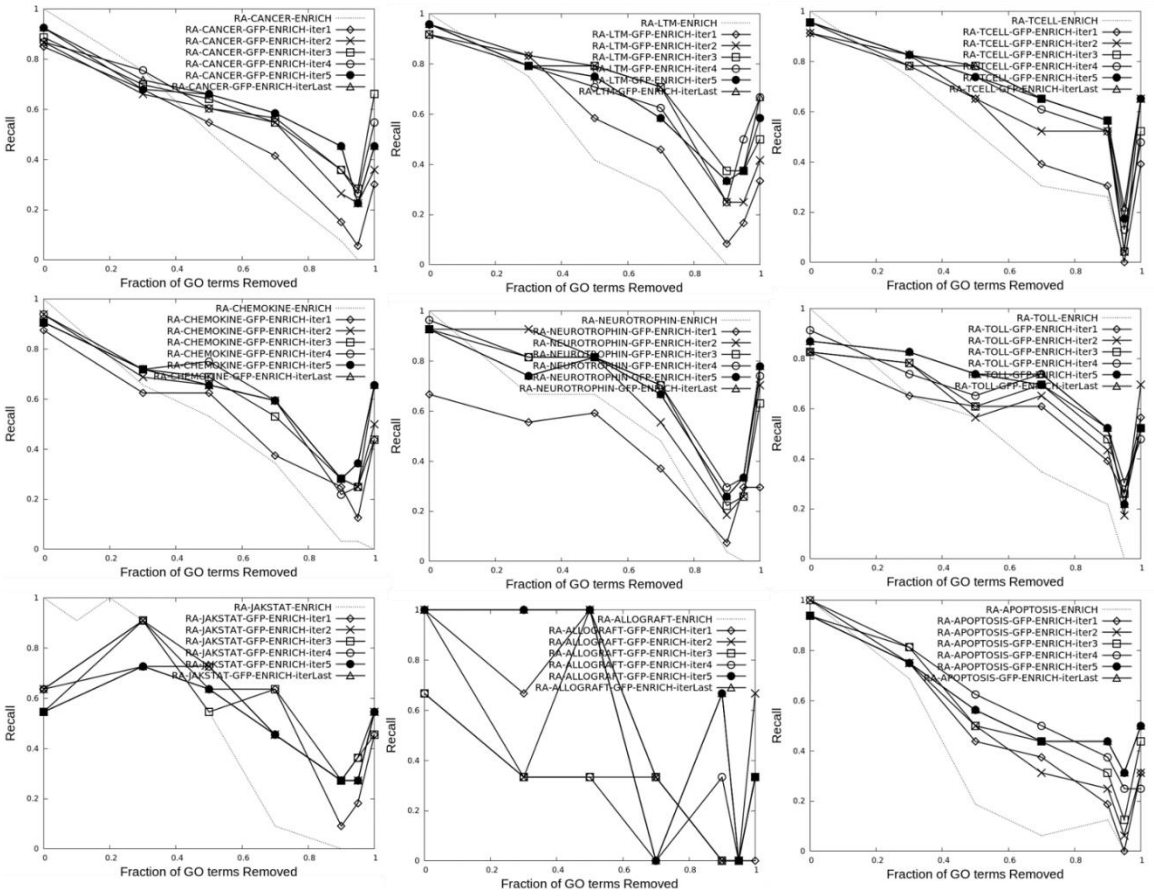

Figure B4 GFP recall of GO removal simulations

Group function prediction to 9 groups of proteins. Recall of prediction was reported after removing a fraction of GO terms.

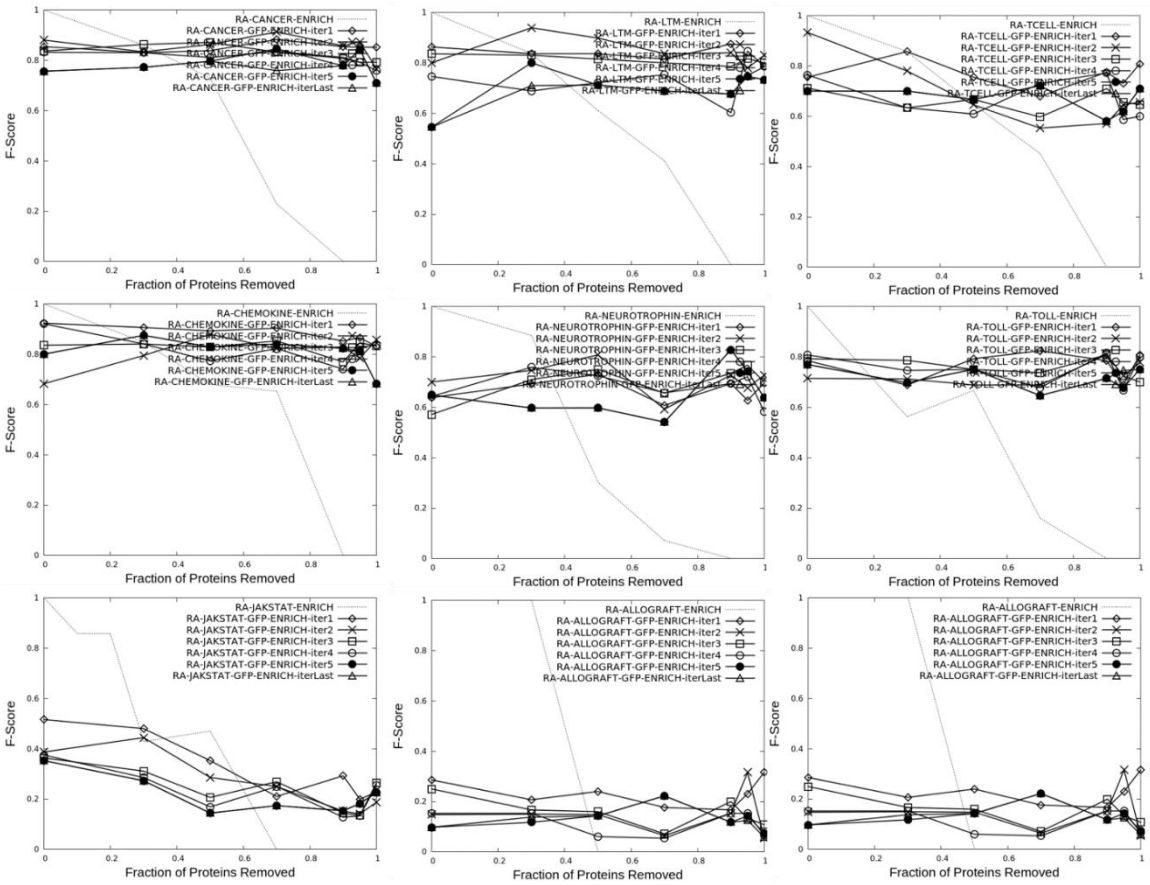

Figure B5 GFP f-score of protein removal simulations

Group function prediction to 9 groups of proteins. F-score of prediction was reported after removing a fraction of proteins.

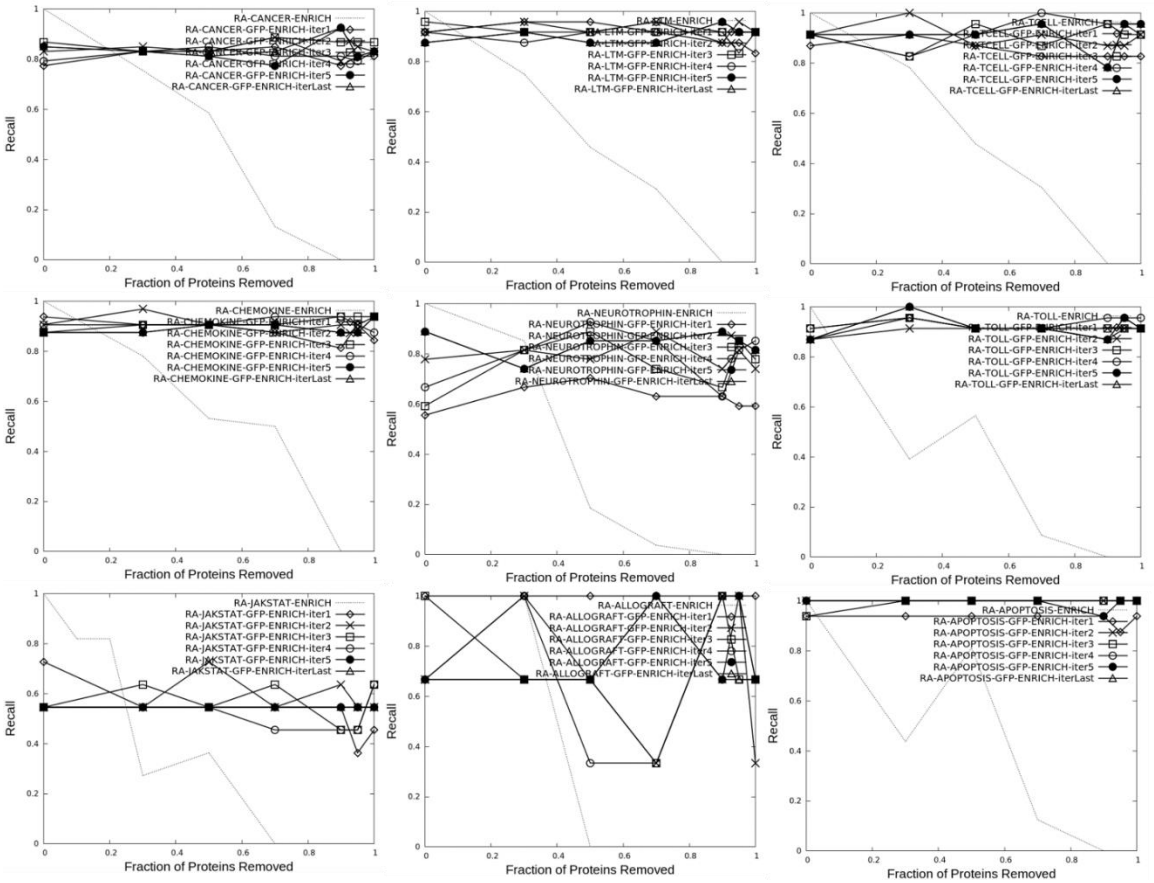

Figure B6 GFP recall of protein removal simulations

Group function prediction to 9 groups of proteins. Recall of prediction was reported after removing a fraction of proteins.

VITA

VITA

Ishita Kamal Khan

Education

B.Sc., Computer Science & Engineering, 2004-2009, Bangladesh University of Engineering & Technology, Dhaka, Bangladesh
Ph.D., Computer Science (Computational Life Sciences specialization), 2010-2016, Purdue University, West Lafayette, Indiana, USA

Research Interests

- Computational biology
- Machine Learning
- Large scale computational data analysis and mining
- Algorithm designing for bioinformatics problems
- Graph and network data analysis
- Biomedical informatics, electronic medical records
- Personalized medicine

PUBLICATIONS

PUBLICATIONS

Published

1. Genome-scale prediction of moonlighting proteins using diverse protein association information, **Khan I**, Kihara D, *Bioinformatics*, [Epub ahead of print] doi: 10.1093/bioinformatics/btw166 (2016)

2. Missing gene identification using functional coherence scores, Chitale M, **Khan I**, Kihara D., I*Scientific Reports*, In Press (2016).

3. An expanded evaluation of protein function prediction methods shows an improvement in accuracy, Jiang X, .., **Khan I**, Kihara D, .., Radivojac P et al. (147 authors) , *Genome Biology*, In Press (2016)

4. PFP and ESG protein function prediction methods in 2014: Effect of database updates and ensemble approaches, **Khan I**[1], Wei Q[1], Chapman S, Dukka B. K. C., Kihara D., *GigaScience*, 4:1-14 (2015)

5. IAS: Interaction specific GO term associations for predicting protein-protein interactions, Yerneni S, **Khan I**, Wei Q, Kihara D, *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, [Epub ahead of print] doi:10.1109/TCBB.2015.2476809 (2015)

6. Genome-scale identification and characterization of moonlighting proteins, **Khan I**, Chen Y, Dong T, Hong X, Takeuchi R, Mori H, Kihara D., *Biology Direct*, 9(1):30 (2014)

7. PFP/ESG: automated protein function prediction servers enhanced with Gene Ontology visualization tool, **Khan I**[1], Wei Q[1], Chitale M, Kihara D., *Bioinformatics*, 31(2):271-2 (2014)

8. Computational characterization of moonlighting proteins, **Khan I**, Kihara D., *Biochemical Society Transactions*, 42(6):1780-5 (2014)

9. In-depth performance evaluation of PFP and ESG sequence-based function prediction methods in CAFA 2011 experiment, Chitale M[1], **Khan I**[1], Kihara D., *BMC Bioinformatics*, 14(S-3): S2 (2013)

10. Evaluation of function predictions by PFP, ESG, and PSI-BLAST for moonlighting proteins, **Khan I**, Chitale M, Rayon C, Kihara D., *BMC Proceedings*, 13:6 Suppl 7:S5 (2012)

In Review

11. NaviGO: Interactive tool for gene ontology visualization and similarity quantification, Wei Q, **Khan I**, Kihara D., In Review *BMC Bioinformatics* (2016)

12. Book chapter on "Exploring Structure-Function Relationship in Moonlighting Proteins" by Das S, **Khan I**, Kihara D, Orengo C., Henderson B ed., Springer (2015)

To be submitted *Manuscripts completed and awaiting revision from PI*

13. Book chapter on "Using PFP and ESG protein function prediction web servers" by Wei Q, McGraw J, **Khan I**, Kihara D, to be submitted in *Methods in Molecular Biology*, Kihara D ed., Springer (2016)

14. Book chapter on "MPFit: Automated prediction of moonlighting proteins using diverse protein association information" by **Khan I**[1], McGraw J[1], Kihara D, to be submitted in *Methods in Molecular Biology*, Kihara D ed., Springer (2016)

15. DextMP: Moonlighting protein prediction by deep dive into text, **Khan I**, Kihara D (2016)

16. Finding functionally relevant genes using logic patterns in genetic phylogeny, Liu L[1], **Khan I**[1], Dong T, Chen L, Luo W, Kihara D (2016)

17. Network-based function prediction of protein groups, **Khan I**, Kihara D (2016)