DUJASE Vol. 3(2) 127-133, 2016 (January)

Protein Function Prediction Based On Protein-Protein Interaction Together with Sequence and Structural Similarity Information

Shimul Chandra Mondal[1] and Saifuddin Md. Tareeq[1]

[1] *Department of Computer Science and Engineering, University of Dhaka, Dhaka 1000, Bangladesh*

E-mail: shimulcsedu@gmail.com, smtareeq@cse.du.ac.bd

Received on 21.06.2015. Accepted for publication on 02.11.2015.

ABSTRACT

Recent advances in experimental biology makes large amounts of protein-protein interaction (PPI) data available. Thus, using PPI data to functionally annotate proteins has been extensively studied. But if there is not enough information about annotation available in the network, most existing network-based approaches do not work well. In a recent interaction network based research work proposal has been made to combine PPI data and sequence similarity information to boost up the prediction performance. But we know that structural similarity is much more affective for predicting protein functions, because protein structure is far more conserved than sequence. Here we have proposed to use structural similarity information together with PPI data and sequence similarity information for predicting protein function. Our method divides function prediction into two phases: first, the original PPI network is enriched by adding a number of implicit edges that are inferred from protein sequence and structural similarity information. Second, a collective classification algorithm is employed on the new network to predict protein function. The experimental results support our assumption and provide better function prediction results than method with PPI and sequence similarity information only.

Keywords: Protein-Protein Interaction, Sequence and Structural Similarity, Classification.

1. Introduction

Proteins function can be determined accurately through experimental approaches. But experimental approaches are not only costly but also time consuming. Computational approaches can be applied to predict functions of protein because these processes are cheap and faster than experimental approaches. Though computational approaches can't give absolute accurate result, but a partially accurate result can be used to narrow down the experimental domain.

Protein function prediction using computational approaches are mostly driven by data-intensive procedures [1]. Recent development in experimental biology makes large amounts of PPI data available. These data are commonly represented as networks [2] and in such network a node corresponds to a protein and an edge corresponds to an interaction between a pair of proteins. Edge weights are determined based on the type of interaction and this weight is proportional to interaction strength among proteins [3].

If the PPI network is sparse meaning that it doesn't consist of enough interaction data, then the prediction algorithm gives poor performance [4]. A way to enrich the PPI network is to adding some extra edges based on some biological insight. If two proteins have sequence similarity over a percentage, then an edge can be added between them, where the weight of this edge will be decided based on their similarity. Using sequence similarity information to enrich a poor PPI network can increase the prediction performance [4].

In this paper we propose to use structural similarity information together with sequence similarity information to increase prediction performance because functionality of any protein is closely related and depends on the 3D structure [5]. If only structural similarity is used to enrich a sparse PPI network then there is a possibility that the network can be remained sparse. In this research work we provide a way to get structural similarity score for determining extra edges through 3D templates as in [6]. Experimental results suggest that the protein function prediction accuracy is increased than adding extra edge with only sequence information.

2. Literature Review

Classical computational approaches try to characterize each protein by collecting a set of features. Then apply suitable machine-learning algorithms to develop annotation rules based on those features [7]. But recently available vast networks of protein interactions within the cell have made it possible to go beyond those one-dimensional approaches and now it is possible to study protein function in the context of a network [2]. A node in the network corresponds to a protein and an edge corresponds to an interaction between a pair of proteins.

Existing computational approaches based on PPI data for protein function prediction might be distinguished in two types of approaches [3]: direct annotation methods and module-assisted methods. Direct annotation methods determine functions of a protein from the known functions of its neighbor. Direct annotation methods determine functions of a protein from the known functions of its neighbor. Neighbor counting approaches [8, 9, 10, 4] uses biological hypothesis that interacting proteins might have similar functions and rank each function based on occurrence in the neighbor. The problem with one or more of these methods are that association is not given any significance value and the full topology of the network is taken into account and the protein at different distances are

treated in the same way. In graph theoretic methods [11, 12, 13] basic idea is to simulate the spread of each protein over time through the network and assign each un-annotated protein a score based on the function flow it receive during simulation. Some of the graph theoretic approaches consider full topology of the network but do not take local proximity into account. A number of probabilistic approaches [14, 15] have been suggested based on the fact that the function of a protein is independent of all other proteins given the functions of its immediate neighbors. These methods first estimate prior and conditional probabilities of annotations and then optimize the joint likelihood of all target annotations.

Module-assisted approaches first identify coherent groups of genes and then assign functions to all the genes in each group. The module-assisted methods differ from one another by their module detection technique [16] namely hierarchical clustering [17, 18] and graph clustering [19, 7, 16]. A key problem of this kind of approaches is how to define the similarity between two proteins.

There are some integrative methods [20, 21] which integrate diverse information for protein function prediction. In those methods if the query protein has PPI information prediction is carried out using network based approach otherwise hybrid property based method is applied. Indeed most network based approaches do not work well if there is not enough PPI information [4]. Therefore we have proposed a method which combines PPI information, sequence similarity information and structural similarity information to improve prediction performance.

3. Proposed Algorithm

Protein function prediction is a multi-label classification problem. Different proteins have different functions and one protein can have multiple functions, this can be represented as a set F=(F_1, ,F_m) and also proteins can be represented as a set =(P_1, , P_n). Of this set some proteins are labeled, suppose first x proteins are labeled as y_1,…, y_x. Each protein label y_i is represented as a vector. If the protein P_i is associated with a function F_j, then y_{ij}=1 otherwise y_{ij} = 0, that means they don't have any association between them. Table 1 shows the tabular representation of protein function relation.

Table 1: Tabular representation of Protein-Function relation

	P_1	P_2	…	P_{n-1}	P_n
F_1	0	1	…	0	0
F_2	1	0	…	0	0
F_3	1	1	…	0	0
F_4	1	1	…	0	0
…	…	…	…	…	…
F_m	0	0	…	0	0

The problem is, we have to predict the labels y_{1+1}, ..., y_x for the remaining protein which are not associated with any function. It is possible to think the PPI network as a finite undirected graph, G=(P,E,W), with a vertex set $P = A \bigcup U$ where A corresponds to the set of annotated proteins and U corresponds to the set of un-annotated proteins. Interaction between proteins are represented as edge and each edge $e_{i,j}$ ε E denotes an observed interaction between protein P_i and P_j. The weight between edges $w_{i,j}$ ε W indicates the interaction confidence between P_i and P_j [4]. Table 2 shows a sample tabular representation for a PPI network.

But if this network is sparse which means there is not enough interaction data among proteins available then collective classification will not be able to produce expected result. Therefore in such situation the network must be enriched with biological insightful information.

Table 2: Tabular representation of Protein-Protein interaction

	P_1	P_2	P_3	…	P_{n-2}	P_{n-1}	P_n
P_1	0	87	34	…	91	76	43
P_2	-	0	91	…	0	0	47
P_3	-	-	0	…	45	67	72
P_4	-	-	-	0	94	0	0
…	-	-	-	-	0	…	…
P_{n-1}	-	-	-	-	-	0	53
P_n	-	-	-	-	-	-	0

3.1 Enriching Protein Network

At first the similarity score between each pair of proteins are calculated using the Basic Local Alignment Search Tool (BLAST) [26]. For the protein P_x, we define its sequence similarity scores with other proteins like this:

$$SEQ(P_x)=[s_{x,1},s_{x,2},..,s_{x,v}] \quad (1)$$

For *v* numbers of proteins this can be represented as a matrix or a table. Table 3 shows the tabular representation of protein similarity scores. Where the similarity score between protein P_x and protein P_i is $s_{x,i}$. Here $s_{x,i}$ = 0 if x = i, that means, self-similarity are not considered.

Table 3: Tabular representation of Protein-Protein sequence similarity scores

	P_1	P_2	P_3	…	P_{n-2}	P_{n-1}	P_n
P_1	0	0	0	…	31	56	23
P_2	-	0	0	…	0	0	34
P_3	-	-	0	…	48	0	0
P_4	-	-	-	0	76	0	0
…	-	-	-	-	0	…	…
P_{n-1}	-	-	-	-	-	0	31
P_n	-	-	-	-	-	-	0

Then we use 3D-Coffee [6] tools for finding out the structural similarity score between each pair of proteins. For the protein P_x, we define its structural similarity scores with other proteins. For *v* numbers of proteins this can be also represented as a matrix like we did for sequence.

$$STRUC(P_x)=[t_{x,1},t_{x,2},..,t_{x,v}] \quad (2)$$

In the next step of the method we have used a collective classification method similar to [22] given in Algorithm 1 in order to predict protein function based on this new network.

A pair of proteins may have three types of edges between them and they are: physical interaction edge (explicit edge), sequence similarity based edge (implicit edge), structure similarity based edge (implicit edge). A pair of proteins may have all of these edges or two of these edges or one of these edges or none of these edges. If there is no edge then our algorithm has nothing to do with it. If they have both explicit and any of implicit edge then we need to control the trade-off between them. For this reason parameter $\lambda \in (0, 1)$ is used. Another parameter $\beta \in (0, 1)$ is used to control the source of implicit neighbors using $\beta = 1$ for sequence similarity, $\beta = 0$ for structural similarity. If the value of β is in between 0 or 1 we have got a network which is enriched by mixed similarity score.

Formally, for a query protein P_x that has k_x explicit neighbors and k_1 implicit neighbors for sequence similarity and k_2 implicit neighbors for structural similarity, we define the corresponding edge weights as:

$$\begin{aligned} e_x &= [w_{x,1}, w_{x,2} w_{x,k_x}], e_1 \\ &= [s_{x,1}, s_{x,2}, s_{x,k_1}], e_2 = [t_{x,1}, t_{x,2}, t_{x,k_2}] \end{aligned} \quad (3)$$

where e_x are the vector of explicit edges and e_1 and e_2 are the vectors of implicit edges for sequence similarity and structural similarity respectively. The probability of protein P_x having the j-th function F_j is computed as:

$$P_x^j = \lambda \frac{1}{\eta_x} \sum_{i=1}^{k_x} f_{ij} w_{x,i} + (1-\lambda) \left(\beta \frac{1}{\eta_1} \sum_{i=1}^{k_1} f_{ij} s_{x,i} + (1-\beta) \frac{1}{\eta_2} \sum_{i=1}^{k_2} f_{ij} t_{x,i} \right) \quad (4)$$

where η_x, η_1 and η_2 are the normalizers and calculated as:

$$\eta_x = \sum_{j=1}^{k_x} \sum_{i=1}^{m} f_{i,j} w_{x,i}, \quad \eta_1 = \sum_{j=1}^{k_1} \sum_{i=1}^{m} f_{i,j} s_{x,i}, \quad \eta_2 = \sum_{j=1}^{k_2} \sum_{i=1}^{m} f_{i,j} t_{x,i} \quad (5)$$

For any query protein P_x, the initial functional probability distribution can be denoted as an m-dimensional vector which is defined as:

$$\bar{a}_x = [P_x^1, P_x^2, \ldots P_x^m] \quad (6)$$

Most of the proteins may have more than one functions, therefore protein function prediction is a multi-label classification problem. For the query protein P_x, its most related function can be computed using the Equation (7).

$$b_x^1 = \arg\max_{j \in [1,m]} P_x^j \quad (7)$$

Where b_x^1 is the argument value of j, that maximizes the value of P_x^j and it is regarded as the 1st-rank result. Accordingly, the second most associate function b_x^2 is the 2nd-rank result and the third most likely function b_x^3 is the 3rd-rank result. In some conditions, when more than one element P_x^j has the same score, their ranks will be assigned based on their order of appearance [4]. An m-dimensional vector $\bar{b}_{x,i}$ can be used for the query protein P_x to record its ranking result in the i-th iteration using Equation 8.

$$\bar{b}_{x,i} = [b_{x,i}^1, b_{x,i}^2, .., b_{x,i}^m] \quad (8)$$

Once the threshold number of iterations is reached, there is a matrix M_x with S rows and m columns for the query protein P_x like Equation 9.

$$M_x = [\bar{b}_{x,1}, \bar{b}_{x,2}, .., \bar{b}_{x,s}]^T \quad (9)$$

The most frequently sampled function is denoted by c_x^1 which appears in the first column of the matrix M_x giving the first rank predicted function. Therefore the final result is an m-dimensional vector $\bar{c}_x$ for the query protein P_x and can be given as Equation 10.

$$\bar{c}_x = [c_x^1, c_x^2, .., c_x^m] \quad (10)$$

Algorithm 1 collective classification using Gibbs sampling

1: for each query protein P_x do
2: compute the initial $\bar{a}_x$ using $A \cap e_x$, $A \cap e_1$, *and* $A \cap e_2$
3: end for
4: for i=1 to m do
5: for each query protein P_x do
6: Update $\bar{a}_x$ using $A \cap e_x$, $A \cap e_1$, *and* $A \cap e_2$
7: end for
8: end for
9: for i=1 to n do
10: for each query protein P_x do
11: Update $\bar{a}_x$ using $A \cap e_x$, $A \cap e_1$, *and* $A \cap e_2$
12: Create $\bar{b}_{x,i}$ to record the m-rank result
13: end for
14: end for
15: for each query protein P_x do
16: calculate the final result $\bar{c}_x$ based on matrix M_x
17: end for

4. Experimental Set up

For the experiment all protein identifiers are collected from Saccharomyces Genome Database (SGD) [23]. There are total 5911 proteins listed here. We need four types of information to be given as input and these are protein annotation information, protein-protein interaction information, sequence similarity information and structural similarity information. Protein annotation information is collected from gene ontology database [24]. Gene ontology annotations are arranged in a hierarchical order, and consist of three basic gene ontology namespaces: molecular function, biological process (if known) and cellular component (if known). Experimental data contains protein name and function separated by a '>' character. This information is prepared as a simple text file. Figure 1 shows the input file structure for protein annotation inputs. Protein-protein interaction data for this research work is collected from string database [25]. STRING is a database of known and predicted protein interactions including both direct (physical) and indirect (functional) associations and were mainly derived from four data sources: genomic context, high-throughput experiments, conserved co-expression and previous knowledge.

```
AAD14>cellular aldehyde metabolic process
AAD14>aryl alcohol dehydrogenase (NAD+) activity
AAP1>glycogen metabolic process
AAP1>proteolysis
AAP1>aminopeptidase activity
AAR2>spliceosomal tri snRNP complex assembly
AAR2>molecular function
AAT1>asparagine biosynthetic process from oxaloacetate
AAT1>aspartate biosynthetic process
AAT1>chronological cell aging
AAT1>replicative cell aging
AAT1>L-aspartate:2-oxoglutarate aminotransferase activity
SRV2>actin filament organization
SRV2>actin filament severing
SRV2>Ras protein signal transduction
SRV2>actin binding
SRV2>adenylate cyclase binding
ORC6>chromatin silencing at silent mating type cassette
ORC6>DNA replication initiation
ORC6>pre-replicative complex assembly involved in nuclear cell cycle DNA replication
ORC6>DNA replication origin binding
BRR2>spliceosome conformational change to release U4 (or U4atac) and U1 (or U11)
BRR2>ATP dependent RNA helicase activity
SNU114>generation of catalytic spliceosome for first transesterification step
SNU114>mRNA splicing, via spliceosome
```

Fig. 1: A sample input file for protein annotations

This PPI information is prepared as a simple text file. Figure 2 shows the input file structure for protein annotation inputs. Each line of these file contains protein name and function. Protein name and function is separated by a ' ' (space). For a protein there can be more than one function.

```
ABF2 ACT1 98
ACT1 SRV2 99
ACT1 ABP1 99
ABP1 ARK1 99
ABP1 PRK1 99
PRK1 ARK1 99
ORC6 AAD1 99
BRR2 AAR2 99
BRR2 SNU114 99
AAR2 SNU114 99
```

Fig. 2: A sample input file for PPI information

Sequence similarity between each pair of proteins is calculated using BLAST [26]. We have generated this information as simple file using PHP and mysl. The format for sequence similarity information input file is given on Figure 3. Each line of this file contains two protein names and the sequence similarity score between them. Each of protein names and score are separated by ' '(space).

```
AAC1 ACT1 75.00
AAC1 AAC3 74.00
AAC1 PRK1 62.00
AAC1 AAT2 57.00
AAC1 ABP1 50.00
AAC3 AAR2 80.00
AAC3 SRV2 67.00
AAC3 ARK1 63.00
AAC3 BRR2 53.00
AAC3 ADP1 50.00
ORC6 ABF2 83.00
ORC6 AAT2 83.00
ORC6 SRV2 57.00
ORC6 AAD3 50.00
ORC6 AAD14 40.00
AAD14 AAD4 93.00
AAD14 AAD6 88.00
AAD14 AAD3 80.00
AAD14 ABF2 42.00
AAD14 SRV2 39.00
PRK1 ARK1 63.00
PRK1 SNU114 47.00
PRK1 ACT1 45.00
PRK1 AAD6 42.00
PRK1 AAT1 41.00
```

Fig. 3: A sample input file for sequence similarity information

```
AAC1 BRR2 66.00
AAC3 ACT1 76.00
ORC6 AAT2 61.00
AAD14 AAD6 92.00
PRK1 ARK1 67.00
ACT1 BRR2 76.00
AAD3 AAD4 92.00
AAD4 AAD6 71.00
AAD6 ABF2 58.00
AAH1 BRR2 100.00
AAP1 AAR2 64.00
AAR2 BRR2 62.00
AAT1 BRR2 78.00
AAT2 BRR2 76.00
SRV2 ABF2 51.00
ARK1 ABF2 59.00
ABF2 BRR2 46.00
BRR2 ABP1 49.00
```

Fig. 4: A sample input file for structural similarity information

We have generated structural similarity scores between each pair of proteins using 3D-coffee [6]. We also generated this information as simple file using PHP and mysql. The input format for protein structure similarity information input is given on Figure 4. Each line of this file contains two protein names and structural similarity score between them. Each of protein names and score are separated by ‘’(space).

Required program code for the research work is implemented using java. The input files are specified using command-line arguments. Files are then read using java file reader line by line. Then based on file type input lines are spitted using ‘ ’(space) or ‘>’. We stored this information using array map.

Array maps are of dynamic length and can be used like associative array. That means one can access the values using index and these index can be a string or a integer or any user defined index. A reverse array is also maintained for this information, so that it is possible to get the index using values. This removes the necessity for searching through a map for a desired value and also decreased the access time.

Because it is not possible to simply determine whether a prediction is correct or wrong[27] we use Positive Predictive Value(PPV), a well known and widely-used performance measure [28]. We have also used recall and f-measure for better clarity of the result. We have run eight different experiments with the combination of different enriching method and different edge selection method. In line with previous research [4] we have set the value of λ to be 0.3 and K = 5 for our experiments.

Value of similarity score (H) is determined experimentally. Figure 5 shows the relation among PPV, value of H and amount of extra edges. Figure 6 shows the relation among Negative Predictive Value (NPV), the value of H and amount of extra edges. Therefore based on the result for both PPV and NPV we define H as 65. At this point if the value of β is not 0 or 1, then we have got a network which is enriched by mix similarity score and to emphasis both sequence and structural similarity equally we have assigned β as 0.5.

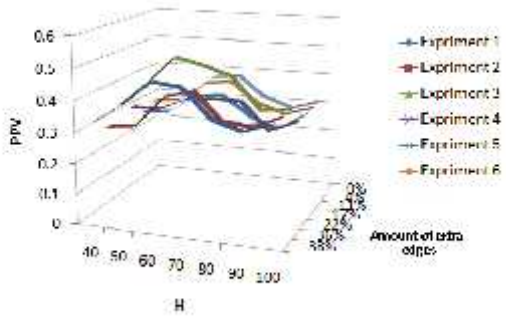

Fig. 5: Results for different values of PPV and H for different network setup

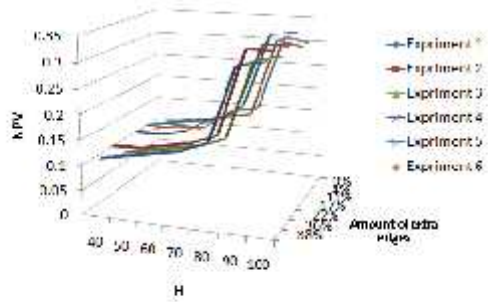

Fig. 6: Results for different values of NPV and H for different network setup

5. Results and Discussion

We have run these experiments and Table 4 shows the results. From these result we observe that mix similarity information based enriching method gives more accurate result than only sequence similarity information based enriching method. From our experiments we get some clear indication. First of all we can say that adding implicit edges definitely helps to increase the prediction performance. Though protein functions are more closely related with protein 3D structures, it has been observed that mixing sequence and structure similarity information together increases the prediction performance than with either sequence or structural similarity information. This is because of the fact that functionally unrelated may conform to similar structure.

Table 4: Experiment results

Exp. No	Network type	Enriching Method	Source of Edge	PPV	Recall	f-measure
1.	Standard network	Sequence similarity	Top K	0.56	0.73	0.63
2.	Standard network	Sequence similarity	Edges over H	0.60	0.78	0.67
3.	Standard network	Structure similarity	Top K	0.58	0.80	0.67
4.	Standard network	Structure similarity	Edges over H	0.62	0.83	0.70
5.	Standard network	Mix similarity	Top K	0.64	0.84	0.72
6.	Standard network	Mix similarity	Edges over H	0.68	0.89	0.77
7.	Sparse network	Sequence similarity	Top K	0.47	0.70	0.56
8.	Sparse network	Sequence similarity	Edges over H	0.49	0.73	0.58
9.	Sparse network	Structure similarity	Top K	0.50	0.71	0.58
10.	Sparse network	Structure similarity	Edges over H	0.51	0.73	0.60
11.	Sparse network	Mix similarity	Top K	0.61	0.76	0.60
12.	Sparse network	Mix similarity	Edges over H	0.62	0.78	0.69

6. Conclusion

In this research work we have developed a more effective method for enriching a poor PPI network to predict functionality of proteins. Here we have done our experiment with one set of data that we have got from Saccharomyces Genome Database (SGD). And the experimental result suggests that our method predict protein function better than the method with PPI data and sequence information only. In this research work we have chosen the number of implicit edges experimentally but in future we would like to devise a automatic method so that the performance our algorithm can be optimized.

References:

1. Tornow S, Mewes HW, Functional modules by relating protein interaction networks and gene expression Nucleic Acids Res 31(21): 6283-6289 (2003).
2. Pavlidis P, Weston J, Cai J, Grundy WN, Gene functional classification from heterogeneous data Proceedings of the Fifth Annual International Conference on Computational Biology. Montreal, Quebec, Canada: ACM Press (2001).
3. Sharan R, Ulitsky I, Shamir R, Network-based prediction of protein function, Mol Syst Biol (2007).
4. Xiong W, Liu H, Guan J and Zhou S, Protein function prediction by collective classification with explicit and implicit edges in protein-protein interaction networks BMC Bioinformatics, 14(Suppl 12):S4 (2013).
5. Lord PW, Stevens RD, Brass A, Goble CA, Investigating semantic similarity measures across the Gene Ontology: the relationship between sequence and annotation, Bioinformatics 19: 1275-1283 (2003).
6. Armougom F, Moretti S, Poirot O, Audic S, Dumas P, Schaeli B, Keduas V, Notredame C, Expresso: automatic incorporation of structural information in multiple sequence alignments using 3D-Coffee, Nucleic Acids Res. (2006).
7. Dunn R, Dudbridge F, Sanderson C, The use of edge-betweenness clustering to investigate biological function in protein interaction networks. BMC Bioinformatics, 6:39 (2005).
8. Schwikowski B, Uetz P, Fields S, A network of protein-protein interactions in yeast Nat Biotechnol 18: 1257-1261. (2000).
9. Chua HN, Wong L, Exploiting indirect neighbors and topological weight to predict protein function from protein-protein interactions, Bioinformatics, 22:1623-1630 (2006).
10. Ng KL, Ciou JS, Huang CH, Prediction of protein functions based on function-function correlation relations, Computers in Biology and Medicine, 40(3):300-305 (2010).
11. Vazquez A, FLammini A, Maritan A, et al, Global protein function prediction from protein-protein interaction networks, Nature biotechnology, 21(6):697-700, (2003).
12. Karaoz U, Murali TM, Letovsky S, et al, Whole-genome annotation by using evidence integration in functional-linkage networks Proceedings of the National Academy of Sciences of the United States of America,101(9):2888-2893 (2004).
13. Nabieva E, Jim K, Agarwal A, Chazelle B, Singh M, Whole-proteome prediction of protein function via graph-theoretic analysis of interaction maps, Bioinformatics, 21(Suppl 1):302-310 (2005).
14. Deng M, Tu Z, Sun F, Chen T, Mapping Gene Ontology to proteins based on proteinprotein interaction data, Bioinformatics 20: 895-902 (2004).
15. Letovsky S, Kasif S, Predicting protein function from protein/protein interaction data: a probabilistic approach, Bioinformatics, 19(suppl 1):197-204 (2003).
16. Becker E, Robisson B, Chapple CE, Multifunctional proteins revealed by overlapping clustering in protein interaction network, Bioinformatics, 28(1):84-90 (2012).
17. Brun C, Chevenet F, Martin D, Functional classification of proteins for the prediction of cellular function from a protein-protein interaction network, Genome Biol, 5(1):R6 (2003).
18. Arnau V, Mars S, Marin I, Iterative cluster analysis of protein interaction data, Bioinformatics, 21:364-378, (2005).

19. Bu D, Zhao Y, Cai L, et al, Topological structure analysis of the protein-protein interaction network in budding yeast, Nucleic Acids Research, 31(9):2443-2450 (2003).
20. Chua H, Sung W, Wong L, An efficient strategy for extensive integration of diverse biological data for protein function prediction, Bioinformatics, 23(24):3364-3373 (2007).
21. Hu L, Huang T, Shi X, Lu W, Cai Y, et al, Predicting functions of proteins in mouse based on weighted protein-protein interaction network and protein hybrid properties. PLoS ONE, 6(1):e14556, (2011).
22. Sen P, Namata G, Bilgic M, Getoor L, Gallagher B, Eliassi-Rad T, Collective classification in network data, AI Magazine, 29:93-106 (2008).
23. Michael C, J, Caroline Adler, Catherine Ball, SGD: Saccharomyces Genome Database, Nucleic Acids Research 26(1):73-79 (1998).
24. Ashburner M, Catherine AB, Judith AB, Gene Ontology: tool for the unification of biology, Nature Genetics, 25:25-29 (2000).
25. Damian S, Andrea F, Michael K, Milan S, The string database in 2011: functional interaction networks of proteins, globally integrated and scored, Nucleic Acids Research 2011, 39:D561-D568 (2011).
26. Altschul S. F, Gish W, Miller W, Myers EW, Lipman DJ, Basic local alignment search tool, Journal of Molecular Biology, 215(3):403-410 (1990).
27. Fan RE, Lin CJ, A study on threshold selection for multi-label classification Tech. rep., National Taiwan University; (2007).
28. Bogdanov P, Singh AK, Molecular Function Prediction Using Neighborhood Features IEEE/ACM Transactions on Computational Biology and Bioinformatics, 7:208-217 (2010).