LINK CLUSTERING ALGORITHM FOR ACCURATE PROTEIN ANNOTATION

Marija Stankova
Faculty of Electrical Engineering and Information Technologies
Skopje, R. Macedonia

Aleksandra Bogojeska
Faculty of Computer Science and Engineering
Skopje, R. Macedonia

ABSTRACT

In order to understand the complex biological processes that take part in the living cells it is necessary to interpret the functional activities and capabilities of one of the main cell structures, the proteins. The functional annotation of the proteins helps determining their role in these processes. In this paper, we present the analysis of complex protein interaction networks using a novel algorithm for network clustering. This method uses link clustering for detection of overlapping functional modules. The newly discovered functional modules then can be extended and used for accurate and highly reliable functional annotation of proteins.

I. INTRODUCTION

The technological advances in the fields of molecular biology and computer science resulted with gathering of large amounts of proteomic and genomic data. The true challenge lies in the step of the data analysis and results interpretation. One of these fields embodies the protein interaction data. The protein interaction networks (PIN) or protein-protein interaction networks (PPI) give information for the common biological processes that proteins perform together. Each protein can have many functions and each functional domain can be seen at different proteins. Yet, even for the most well-studied organisms such as baker yeast, about one-fourth of the proteins remain uncharacterized [1].

Proteins have been observed to act rarely as single objects while performing their function. It is very common, proteins involved in the same biological processes to interact with each other. Therefore, the functions of uncharacterized proteins can be predicted through comparison with the interactions of similar known proteins. This indicates that a detailed examination of the PPI network can reveal significant information about protein function. Clustering is the process of grouping data objects into sets (clusters) which demonstrate greater similarity among objects in the same cluster than in different clusters. In the PPI network context, clustering groups together proteins which share a larger number of interactions. As a result of this process, the modular structure of the PPI network can be uncovered and possible functions for members of the cluster which were previously uncharacterized can be predicted [2].

We are aware of many different clustering techniques for network analysis. Over the last few years, scientists have developed fast, accurate algorithms for network clustering that are able to process the large amounts of data present in social networks. Some of the novel algorithms also take into account the overlapping structure of these networks. One person can belong to different clusters, e.g. his/her personal network and his/her professional network. The advances of these algorithms can be used in the analysis of the PPI networks. It is easy to see the need for exploring overlapping communities in PPI networks since one protein can have multiple functions and therefore will be part of different clusters.

Having this information for the protein and its belongings to different clusters we can then assign each uncharacterized protein with the most frequent functions from the cluster. In our research we used a novel link clustering method which generates clusters in the network, based on the connections between the nodes, rather then the nodes themselves. At the end each link between two nodes belongs to cluster, representing the type of the connection between these nodes and allowing one node to have several types of connections.

The modular structure of the proteins has been explored previously using different methods for network clustering. In [3] Chen and Yuan use modified edge-betweennes clustering method to find functional modules in the network using weights on the nodes generated from microarray expression profiles. Spirin and Mirny [4] in their work use detection of highly connected subgraphs (cliques) combined with Monte Carlo optimization. They also distinguish two types of clusters: protein complexes and dynamic functional modules. Sen *et al.* [5] use spectral clustering for modules generation and later possible functional relationships among the members of the cluster are investigated by predicting new protein-protein connections.

In our work we use the link clustering technique for functional modules detection that will help performing accurate protein annotation assignment.

This paper is structured as follows. In Section II we present the methods for link clustering and protein functional annotation used in our work. In Section III the results of our research are presented and discussed in Section IV.

II. METHOD

Network clustering is a method whose goal is to find organizational structure in a graph by grouping similar

vertices in modules or clusters. It is than considered that vertices that belong to same cluster share common properties.

With the availability of huge amounts data of many different complex networks, diversity of fast clustering methods were developed. Here we can list the Girvan – Newman quality function as one of the most widely used methods [6]. Infomap [7] is currently the most accurate clustering algorithm [8] that uses data flow dynamics using random walks on graphs combined with effective coding maps for cluster detection.

However, the disadvantage of these methods is their inability to detect nodes that can belong to multiple communities. Recent work in the field of community detection recommends avoiding the classical node clustering methods and using novel method that clusters the links instead. By intuition each person can have different types of connection with other people, for example family, co – workers, friends. Here by using link clustering we can easily allow one node to belong to multiple clusters. This link clustering can be used for PPI networks where as mentioned before one protein can be involved in more than one biological processes with other proteins and thus can be part of overlapping communities.

A. *Graph Representation, Line Graphs*

In PPI networks the nodes (vertices) of the graph will represent proteins and the edges their interactions. We represent graph G (N, E) with N set of nodes and E (E ⊆ N x N) set of edges. The inclusive neighborhood of a node *i* from the graph G is denoted as:

$$n_+(i) = \{x \mid d(i,x) \le 1\} \quad (1)$$

where *d(i,x)* is the length of the shortest path between nodes *i* and *x*. This simply includes the node itself and all its neighbors. We can transform the graph G into line graph L(G) by creating a node in L(G) for each edge in G. Two nodes in L(G) will be adjacent if the corresponding edges in G have a node in common [9]. The line graph of a graph G with *n* nodes, *e* edges and d_i vertex degrees will have *n'=e* nodes and the number of edges will be equal to [10]:

$$e' = \frac{1}{2}\sum_{i=1}^{n} d_i^2 - e \quad (2)$$

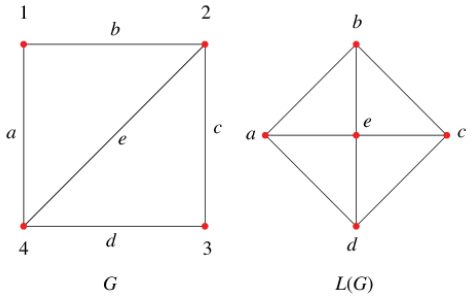

Figure 1. Graph G and its corresponding line graph L(G)

B. *Protein Protein Interaction Data*

Different biochemical methods for extracting protein interaction data are available. However, most of them provide many false positive interactions, which result in non-existing interactions in the databases. Due to this, their results have to be verified and confirmed by at least two of these methods. Moreover, the data sets in each database may include results from different methods. As a consequence, none of the existing databases is reliable and the PPI dataset should be carefully chosen.

The PPIN data we are using is compiled, pre-processed and purified from a number of datasets, like: DIP [11], MIPS [12], MINT [13], BIND [14] and BioGRID [15]. The functional annotations of the proteins were taken from the SGD databases [16], which are unified with Gene Ontology (GO) terminology [17]. The GO consists of three structured ontologies: cellular component, biological process and molecular function.

Using the information and scripts proposed in [18] the dataset is preprocessed by removing the trivial functional annotations, additional annotations are calculated for each protein by the policy of transitive closure and extremely frequent functional labels (appearing in more than 300 proteins) are also excluded because of their generality.

The resulting highly reliable dataset consist of 2502 proteins from the interaction of the baker's yeast (Saccaromyces cerevisiae) with 12708 interactions, and 888 functional annotations.

C. *Link Clustering*

Unlike traditional clustering methods which assumes that a cluster is a set of nodes with many links between them, the approach used by the link clustering method redefines clusters as sets of closely interrelated links. There are two approaches for performing link clustering on a graph. The first one transforms the original graph into a corresponding line graph and then performs any standard node clustering technique. [19]

Our work is based on agglomerative hierarchical clustering for classifying links into topologically related groups presentet in publication [20]. With this clustering technique,

the elements are initially assigned to separate clusters, with most similar clusters being iteratively merged until all elements belong to a single cluster. In the alternative link clustering approach used here, the links are considered as elements, opposite to nodes in standard clustering algorithms. The similarity metrics is also extended for links, which means we calculate the similarity between pairs of links, rather than pairs of nodes with similarity S given by the Jaccard index:

$$S(e_{ij}, e_{jk}) = \frac{|n_+(i) \cap n_+(j)|}{|n_+(i) \cup n_+(j)|} \qquad (3)$$

The clustering process is started by assigning each element to its own cluster. For simplicity, the similarity between clusters is considered equal to the highest similarity between pairs of links, with each link belonging to one of the clusters. This is called single linkage hierachial clustering. We construct a dendrogram by choosing the pair of links with highest similarity value and merging their clusters until all links are unified in one cluster. Ties are being processed simultaneously.

As a result of this process a link dendrogram is created, where each leaf is a link from the graph and the branches represent clusters. The similarity metrics used for merging the clusters is saved as height of the relevant branch. In this dendrogram, the links occupy unique position, while the nodes owing to their links are associated to multiple positions. Every node inherits all the memberships of its links, and therefore can belong to multiple, overlapping clusters.

In order to reveal the most meaningful clusters, the link dendrogram needs to be cut at a certain height. For this reason, a function which measures the quality of a link partition, called partition density, is used. By calculating the partition density D at each step, we are able to determine the best level for cutting the link dendrogram. For a network divided in C subsets $\{P_1, P_2, \ldots Pc\}$, with each subset having $m_c = |Pc|$ links and $n_c = |U_{eij \in Pc}\{i,j\}|$ nodes, we can calculate the partition density D_c of C as:

$$D_c = \frac{m_c - (n_c - 1)}{\frac{n_c(n_c-1)}{2} - (n_c - 1)} \qquad (4)$$

If M represents the total number of links in the network, the partition density D, which is the average of D_c, can be calculated as:

$$D_c = \frac{2}{M} \sum m_c \frac{m_c - (n_c - 1)}{(n_c - 2) - (n_c - 1)} \qquad (5)$$

D reaches its maximum, 1, when each cluster is a fully connected clique, and it is equal to 0 when every cluster is a tree. If a cluster has disconnected components, this has a negative effect on D. Basically D expresses how "clique-ish" compared to how "tree-ish" the partition is.

D. Proteins Functional Annotation

After the clustering step next annotation and characterization of query proteins is performed. For this purpose a simple method is used, defined as follows: for each protein find the clusters where it belongs, calculate the frequencies of each annotation that is present in the clusters and annotate the query protein with the most frequent annotations. The rank of each annotation is calculated with the equation:

$$f(j)_{j \in F} = \sum_{i \in K} z_{ij} \ , \qquad (6)$$

where,

$$Z = \begin{cases} 1, \text{if i - th protein in K is annotated with j - th function from F} \\ 0, \text{otherwise} \end{cases} \qquad (7)$$

and F is the set of functions present in cluster K.

III. Results

The link clustering algorithm was implemented in Matlab as described above. The implementation was first tested on Zachary Karate Network for its accuracy and gave the same network clustering as in the reference publication [20]. On the PPI network dataset the clustering resulted in 2753 clusters, 573 nontrivial (with more than 2 members in the clusters). The greatest cluster has 35 proteins in the component. In the protein annotation step each protein from the network is considered as query protein using the leave-one out method. Then for each term present in the clusters we calculate rank which is than scaled from 0 to 1. The query protein is annotated with all functions that have rank above a previously determined threshold ω. For example, for ω = 0, the query protein is assigned with all the functions present in its cluster. We change the threshold with step 0.1 and compute the number of true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN).

For comparison of different clustering methods we use standard statistical metrics as sensitivity, specificity, precision and recall defined as follows:

$$sensitivity / recall = \frac{TP}{TP + FN} \ , \qquad (8)$$

$$specificity = \frac{TN}{TN + FP} \ , \qquad (9)$$

$$precision = \frac{TP}{TP + FP} \ , \qquad (10)$$

Our method was evaluated against four most used clustering algorithms: Infomap [7], Girvan-Newman Edge Betweenness [6], Girvan-Newman Modularity Function [6]

and another hierarchical clustering method, Common Neighbors Clustering [21].

The defined statistical metrics are used for generation of Fig. 3 where the common statistical plots are presented (Precision-recall, sensitivity-specificity and ROC curve) Table 1 shows the Area Under Curve (AUC) value for each of the tested algorithms.

Table 1: AUC value comparison for different clustering algorithms

Algorithm	Value
Edge Betweenness	0.8430
Infomap	0.8241
BGLL	0.7955
Link Clustering Matlab	**0,7674**
CNH	0,6538

As can be seen from the results our algorihm doesn't produce the highest AUC value. However, more importantly, our algotithm produces highly reliable annotations as can be seen from the precision-recall curve where its values rise above those of all other algorithms.

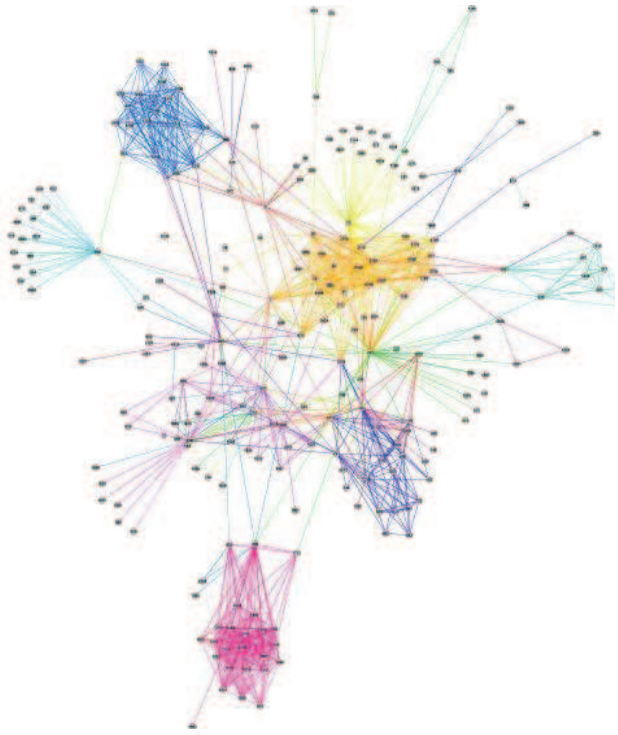

Figure 2. Overlapping community structure in the PPI network of Saccaromyces cerevisiae

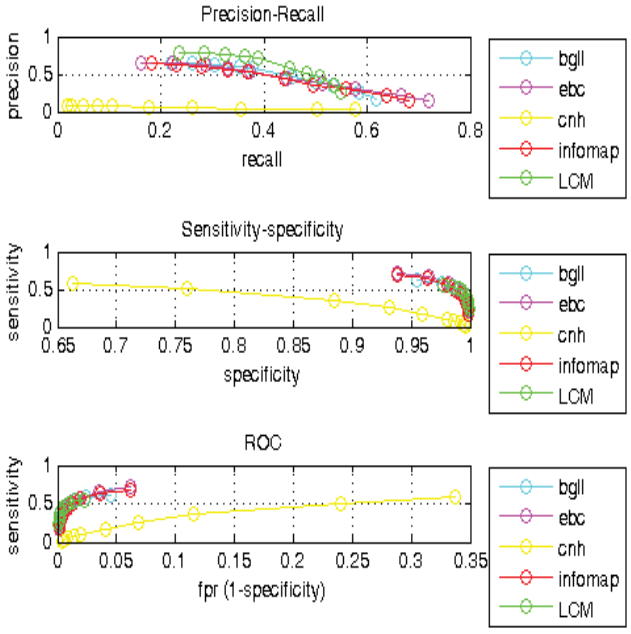

Figure 3. Statistical plots for functional prediction comparison of different clustering methods

Table 2: Function prediction evaluation using different clustering methods

Algorithm	**ω=**	**0.0**	**0.1**	**0.3**	**0.5**	**0.7**	**0.9**
BGLL	sens	0.616	0.584	0.499	0.377	0.304	0.219
	fpr	0.044	0.023	0.010	0.004	0.002	0.001
EBC	sens	0.718	0.665	0.513	0.374	0.287	0.162
	fpr	0.061	0.035	0.011	0.004	0.002	0.001
CNH	sens	0.578	0.503	0.260	0.105	0.049	0.018
	fpr	0.336	0.240	0.068	0.019	0.008	0.003
Infomap	sens	0.682	0.637	0.495	0.369	0.279	0.181
	fpr	0.061	0.035	0.013	0.004	0.002	0.001
LCM	sens	0.548	0.535	0.483	0.388	0.324	0.236
	fpr	0.022	0.014	0.006	0.002	0.001	0.001

IV. Discussion

The usage of overlapping clustering in PPI networks is crucial in order to fully cover and include the proteins characteristics in the process of modules detection. The link clustering algorithm allows fast and accurate overlapping cluster discovery. Using this algorithm we were able not only to generate modular structure in the baker's yeast PPI network but also use this information for further protein annotation.

We can see from the results that the annotation generated with usage of these clusters gives more accurate and reliable results compared to the other methods. Having a false positive rate that is only half of the ones of common clustering methods, we can say that the proteins are annotated with a high rate of true positives functions. In other words, we can conclude that link clustering is more efficient in detecting relevant annotations, compared to the other clustering methods.

REFERENCES

[1] R.Sharan, I.Ultsky, R.Shamir, "Network based prediction of protein function", *Moleculat Systems Biology*, 3:88, 2007

[2] C.Lin, Y.Cho, W.C.Hwang, P.Pei, A.Zhang, "Knowledge Discovery in Bioinformatics: Techniques, Methods and Application", *John Wiley&Sons, Inc.,*2006

[3] J.Chen and B. Yuan, "Detecting functional modules in the yeast protein-protein interaction network," *Bioinformatics* (Oxford, England), vol. 22, no. 18, pp. 2283–2290, Sep. 2006.

[4] V.Spirin and L.A.Mirny, "Protein complexes and functional modules in molecular networks," *Proceedings of the National Academy of Sciences*, vol. 100, no. 21, pp. 12 123–12 128, Oct. 2003.

[5] T.Sen, A.Kloczkowski, and R. Jernigan, "Functional clustering of yeast proteins from the protein-protein interaction network," *BMC Bioinformatics*, vol. 7, no. 1, pp. 355+, Jul. 2006.

[6] S.Fortunato, "Community detection in graphs", *Physics Report,* 2010, vol. 486

[7] M.Rosvall, C.T.Bergstrom "Maps of random walks on complex networks reveal community structure", *PNAS*, January 2008

[8] A. Lancichinetti and S. Fortunato, "Community detection algorithms: A comparative analysis," *Physical Review E*, vol. 80, no. 5, pp. 056 117+, Nov. 2009

[9] J.L.Gross, J.Yellen, "Graph Theory and its application", *Chapman & Hall*, 2006

[10] S.Pemmaraju, S.Skiena, "Computational Discrete Mathematics – Combinatorics and Graph Theory with Mathematica", *Cambridge University Press*, 2003, pp.241

[11] L.Salwinski, C.S.Miller, A.J.Smith, F.K.Pettit, J.U.Bowie, D.Eisenberg, "The Database of Interacting Protein", *Nucleic Acids Research*, 2004

[12] U.Guldener, M.Munsterkotter, M.Oesterheld, P.Ragel, A.Ruepp, and H.W.Mewes " MPact: the MIPS protein interaction resource on yeast" *Nucleic Acids Research Vol*. 34, 2006

[13] A.Chatr-aryamontri, A.Ceol, L.Montecchi Palazzi, G.Nardelli, M.V.Schneider, L.Castagnoli, G.Cesareni "MINT: the Molecular INTeraction database" *Nucleic Acids Research* 35, D572--D574, 2007

[14] G.D. Bader, C. W. V. Hogue "BIND–a data spec. for storing and describing biomolecular interactions, molecular complexes and pathways" *Bioinformatics*, Vol.16(5), 465-477, 2000

[15] B.J.Breitkreutz, C.Stark, M.Tyers " The GRID: The General Repository for Interaction Datasets", *Genome Biology*, 2003

[16] S.Dwight, M.Harris, K.Dolinski, CBall, G. BUnkley. K.Christie, D.Fisk, L.Issel-Tarver, M.Schroeder, G.Sherlock, A.Sethuraman, SWeng, D.Botstein, J.M.Cherry "Saccharomyces Genome Database (SGD) provides secondary gene annotation using Gene Ontology (GO)", *Nucleic Acids Research* 30(1), 2002

[17] The gene ontology consortium "Gene ontology: Tool for the unification of biology", *Nature Genetics* 25(1), 25-29, 2000

[18] S. Letovsky and S. Kasif, "Predicting protein function from protein/protein interaction data: a probabilistic approach," *Bioinformatics*, vol. 19, no. suppl 1, pp. i197–i204, Jul. 2003.

[19] T.S.Evans, R.Lambiotte, "Line Graphs, Link Partitions and Overlapping Communities", *Phys. Rev. E*, 2009, vol. 80

[20] Y.Y.Ahn, J.P.Bargow, S.Lehmann, "Link communities reveal multiscale complexity in networks", *Nature,* 2010, vol. 466, pp. 761–764

[21] M.P.Samanta, S.Liang "Predicting protein functions from redundancies in large scale protein interaction networks", *PNAS*, October 2003