# Thresholding of Semantic Similarity Networks using a Spectral Graph Based Technique

Pietro Hiram Guzzi, Simone Truglia, Pierangelo Veltri, and Mario Cannataro[1]

Department of Surgical and Medical Sciences, University Magna Graecia of Catanzaro
hguzzi@unicz.it

**Abstract.** Semantic similarity measures (SSMs) refer to a set of algorithms used to quantify the similarity of two or more terms belonging to the same ontology. Ontology terms may be associated to concepts, for instance in computational biology gene and proteins are associated with terms of biological ontologies. Thus, SSMs may be used to quantify the similarity of genes and proteins starting from the comparison of the associated annotations. SSMs have been recently used to compare genes and proteins even on a system level scale. More recently some works have focused on the building and analysis of Semantic Similarity Networks (SSNs) i.e. weighted networks in which nodes represents genes or proteins while weighted edges represent the semantic similarity score among them. SSNs are quasi-complete networks, thus their analysis presents different challenges that should be addressed. For instance, the need for the introduction of reliable thresholds for the elimination of meaningless edges arises. Nevertheless, the use of global thresholding methods may produce the elimination of meaningful nodes, while the use of local thresholds may introduce biases. For these aims, we introduce a novel technique, based on spectral graph considerations and on a mixed global-local focus. The effectiveness of our technique is demonstrated by using markov clustering for the extraction of biological modules. We applied clustering to simplified networks demonstrating a considerable improvements with respect to the original ones.

**Keywords:** Graphs, Semantic Similarity Measures, Thresholding

## 1 Introduction

The accumulation of raw experimental data about genes and proteins has been accompanied by the accumulation of functional information, i.e. knowledge about function. The assembly, organization and analysis of this data has given a considerable impulse to research [1]. Usually biological knowledge is encoded by using annotation terms, i.e. terms describing for instance function or localization of genes and proteins. Such annotations are often organized into ontologies, that offer a formal framework to organize in a formal way biological knowledge [2]. For instance, Gene Ontology (GO) provides a set of annotations (namely GO Terms) of biological aspects, structured into three main taxonomies: Molecular function

(MF), Biological Process (BP), and Cellular Component (CC). Annotations are often stored in publicly available databases, for instance a main resource for GO annotations is the Gene Ontology Annotation (GOA) database [3].

A set of algorithms, referred to as Semantic Similarity measures (SSMs), enabled the comparison of set of terms belonging to the same ontology. SSMs take in input two or more ontology terms and produce as output a value representing their similarity. This enabled the possibility to use such formal instruments for the comparison and analysis of proteins and genes [2].

Consequently, many works have focused on: (i) the definition of ad-hoc semantic measures tailored to the characteristics of Gene Ontology ; (ii) the definition of measures of comparison among genes and proteins; (iii) the introduction of methodologies for the systematic analysis of metabolic networks; (iv) building of *semantic similarity networks*, i.e. edge-weighted graph whose nodes are genes or proteins, and edges represent semantic similarities among them [4].

A semantic similarity network of proteins (SSN) is an edge-weighted graph $G_{ssu}$=($V$,$E$), where $V$ is the set of proteins, and $E$ is the set of edges, each edge has an associated weight that represent the semantic similarity among related pairs of nodes.

These networks are constructed by computing some similarity value between genes or proteins. Nevertheless, such networks are usually quasi complete networks, so the use of them as framework of analysis has many problems.

Thus the definition of a threshold on the edge weight to retain only the meaningful relationships is a crucial step. An high threshold may result on the loss of many significant relationship while a low threshold may introduce a lot of noise-

In other kind of networks many methods have been defined: for instance the use of an arbitrary global threshold [5], or the use only of a fraction of highest relationship [6], or statistical based methods [7]. Nevertheless, internal characteristics of SSMs (as investigated in [8]) do not suggest the use of global thresholds. In fact small regions of relatively low similarities may be due to the characteristics of measures while proteins or genes have high similarity. Thus the use of local threshold may constitute an efficient way, i.e retaining only top k-edges for each node [9]. Although this consideration, this choice may be influenced by the presence of local noise and in general may cause the presence of biases in different regions.

Starting from these considerations, we developed a novel hybrid method that merges together both local and global considerations. This method is based on spectral graph theory and it is based on two main considerations.

We apply a local threshold for each node, i.e we retain only edges whose weight is higher than the average of all its adjacent. The choice of the threshold is made by considering a global consideration: the emergence of nearly-disconnected components by looking at the laplacian of the graph and its eigenvalues [10, 11]. In particular we build a novel graph in which edge weights are 0,5 and 1. The weight 0,5 is associated to edges that are retained considering only one adjacent node, while the weight 1 is associated to edges that are retained.

The choice of this simplification has a biological counterpart on the structure of biological networks. It has been proved in many works that these biological networks tend to have a modular structure in which hubs proteins (i.e. relevant proteins) have many connections [12–14]. Moreover, many works proved the existence of community structures, i.e. small dense regions with few link to other regions [15]. These considerations have usually inspired many algorithms for extracting biological relevant modules by analyzing biological networks [16].

From these consideration arises the main hypothesis of this paper: the simplification of quasi complete SSN by removing non relevant edges to evidence the formation of a structure of networks characterized by relatively-small dense networks loosely coupled with other ones.

After the application of the proposed simplification, we analyze resulting networks by applying a common algorithms used to mine graphs. We show that thresholded networks have in general more performances and that the best ones are reached with nearly-disconnected ones.

## 2 Problem Statement

We here introduce main concepts used for the formulation of the main problem of this article.

### 2.1 Spectral Graph Analysis

Spectral graph theory [17] refers to the study of the properties of a graph by looking at the properties of the eigenvalues and eigenvectors of matrices associated to the graph. In particular we here focus on the Laplacian matrix of a graph that is defined as follows [18, 19].

Given an edge-weighted graph $G$ with $n$ nodes, we may define the weighted adjacency matrix $A$ as the $nxn$ matrix in which the element $a_{i,j}$ is defined as follows.

$$a_{i,j} = \begin{cases} w_{i,j} \text{ if i,j are connected;} \\ 0, \quad \text{if i,j are not connected} \end{cases} \tag{1}$$

For these graphs the notion of degree may be easily extended in this way. For each vertex $v_i$ the degree is defined as the sum of the weights of all the adjacent edges $vol_{v_i} = \Sigma_j w_{i,j}$. Then we may define the Degree Matrix $D$ as follows:

$$d_{i,j} = \begin{cases} vol_{v_i}, \text{ if i=j;} \\ 0, \quad \text{elsewhere} \end{cases} \tag{2}$$

Finally, the Laplacian Matrix $L$ is defined as $L = D - A$. Similarly in literature other slightly definitions of Laplacian (e.g. Signless Laplacian, Normalized Laplacian [20]) have been proposed.

Beside the other properties that are related to the characteristic polynomial of laplacian, we here focus on the smallest nonzero eigenvalue, often referred to as Fiedler vector [21]. It has been shown that the number of connected components

is related to the algebraic multiplicity of the smallest eigenvalues in case of both un-weighted and weighted graphs. Starting from this consideration, Ding et al. [10] observed that also nearly-disconnected components may also identified by analyzing the eigenvector associated to the Fiedler vector.

For this study we analysed the spectrum of the graph obtained after the simplification under the hypothesis that a graph with nearly disconnected component may represent a suitable choice. If the graph is connected we will build a novel graph. If the graph has-nearly disconnected component we end the process and we mine the resulting subgraph for the identification of biological relevant modules.

### 2.2 Semantic Similarity Measures

A semantic similarity measure ($SSMs$) is a formal instrument to quantify the similarity of two or more terms of the same ontology. Measures comparing only two terms are often referred to as pairwise semantic measures, while measures that compare two sets of term yielding a global similarity among sets are referred to as groupwise measures.

Since proteins and genes are associated to a set of terms coming from Gene Ontology, $SSMs$ are often extended to proteins and genes. Similarity of proteins is then translated in the determination of similarity of set of associated terms [22, 23]. Many similarity measures have been proposed (see for instance [2] for a complete review) that may be categorized according to different strategies used for evaluating similarity. We here do not discuss deeply $SSMs$ for lack of space.

## 3 The Proposed Approach

We here introduce a method for threshold selection on weighted graph based on the spectrum of the associated laplacian matrix. The process is straightforward. The pruning algorithm examines each node in the input graph. For each node it stores all the weights of the adjacent edges. Then it determines a local threshold $k = \mu + \alpha * sd$, where $\mu$ is the average of weights, $sd$ is the standard deviation and $\alpha$ is a variable threshold that is fixed globally. In this way we realize an hybrid approach since the threshold $k$ has a global component $\alpha$ and a local one given by the average and standard deviation of the weights of the adjacent.

If the weight of an edge is greater than $k$ considering the adjacent of both its nodes, then it will be inserted into the novel graph with unitary weight. Otherwise, hen if the weight of an edge is greater than $k$ considering only one of its adjacent nodes, then it will be inserted into the novel graph with weight 0,5. At the end of this process, the Laplacian of the spectrum of the graph is analyzed as described in Ding et al [10]. If the graph presents nearly disconnected components, then the process stops, alternatively a novel graph with a more stringent threshold $k$ is generated.

### 3.1 Building Semantic Similarity Networks

Following algorithm explains the building of the semantic similarity network $G_{ssu}$ by iteratively calculating semantic similarity among each pair of proteins. For each step two proteins are chosen and the semantic similarity among them is calculated. Then nodes are added to the graph and an edge is inserted when the semantic similarity is greater than 0.

**Algorithm 1:** Building Semantic Similarity Networks

```
Building Semantic Similarity Networks Data: Protein Dataset P,
        Semantic Similairity Measure SS
Result: Semantic Similarity Network G_ssu=V_ssu, E_ssu
initialization;
forall the p_i in P do
    read p_i;
    add p_i in V_ssu ;
    forall the p_j in P, j ≠ i do
        Let σ=SS(p_i,p_j) ;
        if alpha is greater than 0 then
            add the weighted edge (p_i,p_j,σ) to E_ssu;
        end
    end
end
```

### 3.2 Pruning Semantic Similarity Networks

This section explains the pruning of semantic similarity network through an example. To better clarify the process, we use an auxiliary graph $G_{pr}$ that is the final process of pruning. The graph is built in an incremental fashion by considering all the nodes of $G_{ssu}$. The process is straightforward. The pruning algorithm examines each node $\in G_{ssu}$. For each node it stores all the weights of the adjacent edges. Then it determine a local threshold (for instance the average of the weights or the median value as exposed after). At the end of this step, the node $i$ and all the adjacent ones are inserted in to $G_{pr}$ (only if they are not yet present).

Then each edge adjacent to $i$ with weight greater with the determined local threshold is inserted into $G_{pr}$. If the considered edge is not present in $G_{pr}$, the edge will have weight 0.5, otherwise the weight of the edge is set to 1. We used in this work two simple thresholds, the average and the median of all the weights. Finally all the nodes with 0 degree are deleted from $G_{pr}$.

The rationale of this process is that edges that are *relevant* considering the neighborhood of both nodes will compare in the pruned graph with unitary

weight while edges that are *relevant* considering one node will compare with 0.5 weight. In this way we think that we may reduce the noise.

For instance, let us consider the network depicted in Figure 1 and let us suppose that threshold is represented by the average. Without loss of generality we suppose k=0 in this example. Let $AVG(node_i)$ be the average of the weights of nodes adjacent to $node_i$ that is used as threshold.

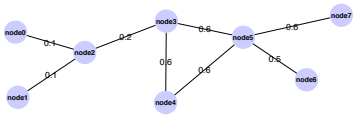


**Fig. 1.** Weighted Semantic Similarity Network.

- The algorithm initially explores $node_0$, since it has degree 1, it is discarded from the analysis.
- Then it explores $node_1$ that is discarded similarly to $node_0$.
- When $node_2$ is considered, the algorithm adds into $G_{pr}$ $node_0, node_1, node_2$, and $node_4$ and the edge $(node_2, node_4)$ with weight 0.5 - (the average of the weights of the neighbours of $node_2$ is equal to 0,13 and other two edges have a lower weight). Figure 2 depicts the produced graph at this step.

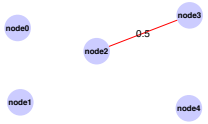


**Fig. 2.** The output of the algorithm at Step 2

- $node_3$ is reached by the visiting. Then $node_4$, and $node_5$ are inserted into $G_{pr}$. The $AVG(node_3)$ is equal to 0,46, so only edges $(node_3, node_5)$ and $(node_3, node_5)$ are inserted into $G_{pr}$ with weight 0.5. Figure 3 depicts $G_{pr}$ after this step.
- $node_4$ is reached. Since all the adjacent nodes have been inserted into $G_{pr}$, no nodes are added into this step. The $AVG(node_4)$ is equal to 0,6, so all the edges must be inserted. In particular edge $(node_4, node_3)$ is yet present, so its weight is updated to 1.0. Diversely, $(node_4, node_5)$ is inserted with weight equal to 0.5. Figure 4 depicts $G_{pr}$ after this step.
- $node_5$ is reached. $node_7$ and $node_8$ are inserted into $G_{pr}$. The $AVG(node_5)$ is 0,575. Consequently the weight of $(node_5, node_3)$ $(node_5, node_3 4$ and in

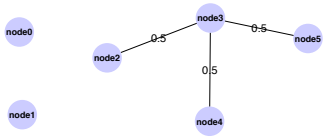


**Fig. 3.** Output after the visit of node3.

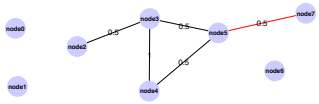


**Fig. 4.** Output after the visit of node4.

$G_{pr}$ is updated to 1, $(node_6, node_7)$ is inserted into $G_{pr}$.Figure 6 depicts $G_{pr}$ after this step.

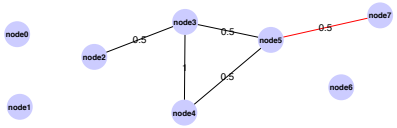


**Fig. 5.** Output after the visit of node5.

- $node_7$ and $node_8$ are visited but discarded since they have degree equal to 1.
- Finally all the nodes with zero degree are eliminated from $G_{pr}$, producing the resulting graph depicted in Figure 6.

The generation of pruned graph is repeated until the graph has nearly disconnected components. This may be evident by analyzing the spectrum of the associated laplacian for value of threshold.

```
Pruning Semantic Similarity Network
 Input SSn Raw Semantic Simlarity Network,
 K  Threshold of Simplification
 Output: SSp Simplified Semantic Similarity Network
  While SSp has not nearly-disconnected component
  $SSp$ = Simplify(SSn,k)
   Increment k
Return: SSp
```

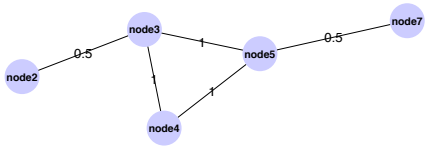


**Fig. 6.** Final pruned graph

### 3.3 Analysis of Semantic Similarity Networks

As introduced, in a Semantic Similarity Networks, nodes represent proteins or genes, and edges represent the value of similarity among them. Starting from a dataset of genes or proteins, a SSN may be built in an iterative way, and once built, algorithms from graph theory may be used to extract topological properties that encode biological knowledge.

As starting point, the global topology of an semantic similarity network, i.e. the study of the clustering coefficient or of the diameter, can reveal main properties of the network and the correspondence with respect to a theoretical model.

In addition to analysis of global properties, the study of recurring local topological features and the extraction of relevant modules, i.e. cliques, has found an increasing interest. For the purposes of this work, we focus on the extraction of dense subgraphs under the hypothesis that they could encode protein complexes.

SS measures are able to quantify the functional similarity of pairs of proteins/genes, comparing the GO terms that annotate them. Thus, there are no constraints on the minimum set size [2].

Since proteins within the same pathway are involved in the same biological process, they are likely to have high semantic similarity. In a similar way, protein belonging to the same complex are likely to have similar biological roles, and therefore they should have high semantic similarity.

The rationale of this study is to demonstrate the ability of semantic similarity networks to represent in a similar way to protein interaction networks. Main difference is represented by the fact that semantic similarity networks may encode more knowledge that is hidden in protein interaction networks.

There exist currently main approaches of analysis of protein interaction networks that span a broad range, from the analysis of a single network by clustering to the comparison of two or more networks trough graph alignment approaches. In this work we consider the use of Markov Clustering Algorithm (MCL) as mining strategy. MCL has been proved to be a good predictor of functional modules when applied to protein interaction networks.

## 4 Case Study

In order to show the effectiveness of this strategy we propose the following assessment:

– we downloaded three dataset of proteins (the CYC2008 dataset [1], the MIPS catalog [24], and the Annotated Yeast High-Throughput Complexes [2] );
– we calculated different semantic similarities among them using FastSemSim tool [3] (we considered 10 semantic similarity measures from those available in FastSemSim ( Czekanowsky-Dice , Dice, G-Sesame, Jaccard, Kin, NTO, SimGic, SimICND, SimIC, SimUI, TO [25] ) and two ontologies Biological Process (BP) and Molecular Function (MF). Consequently we generated 20 SSN for each input dataset.
– we applied the pruning of the semantic network with varying threshold causing the presence of nearly disconnected components and the presence of disconnected components;
– we extracted modules on the raw and simplified networks at various threshold showing the improvements of our strategy showing the improvement in terms of functional enrichment of modules (i.e. the quantification of biological meaning of modules).

As final step we compare our simplification with other global strategies demonstrating the effectiveness of the local simplification.

### 4.1 Results

For each generated network we used the markov clustering algorithm (MCL) to extract modules. The effectiveness of the use of MCL for detecting modules in networks has been demonstrated in many works (see for instance [25]). We here assess how MCL is able to discover *functionally coherent* modules in different semantic similarity network and how this process is positively influenced by the simplification. In particular we show how the process of simplification improves the overall results and how best results are obtained when networks presents nearly disconnected components.

We evaluated the obtained results in terms of *functional coherence* of extracted modules. We define *functional coherence* $FC$ of a module $M$ as the average of semantic similarity values of all the pair of nodes (i,j) composing a module.

$$\sum_{i,j} \frac{SSM(i,j)}{N}$$

, where N is the number of the proteins of the module.

Starting from this definition, we may obtain a single value for all the modules extracted in an execution of MCL by averaging these values. We consider this average value as a representative for the thresholded network. Figures 7, 8, and 9 summarize these results.

[1] wodaklab.org/cyc2008/
[2] wodaklab.org/cyc2008/
[3] fastsemsim.sourceforge.net

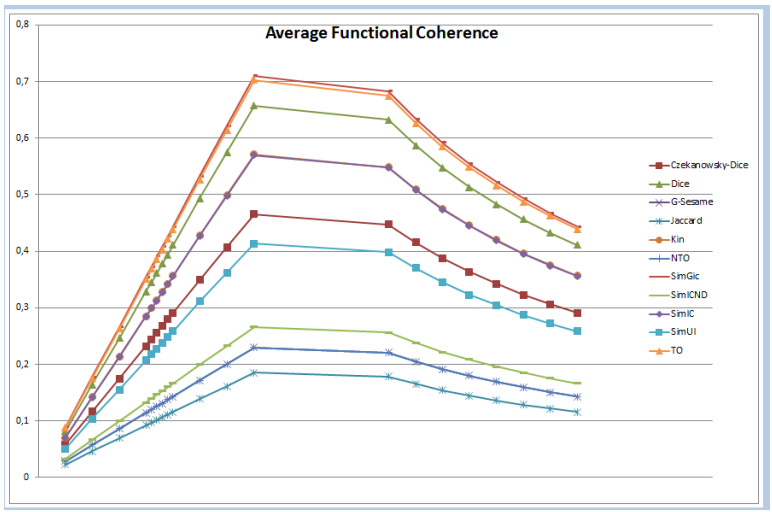


**Fig. 7.** Comparison of Average FC at different Threshold Levels on CYC2008 Dataset

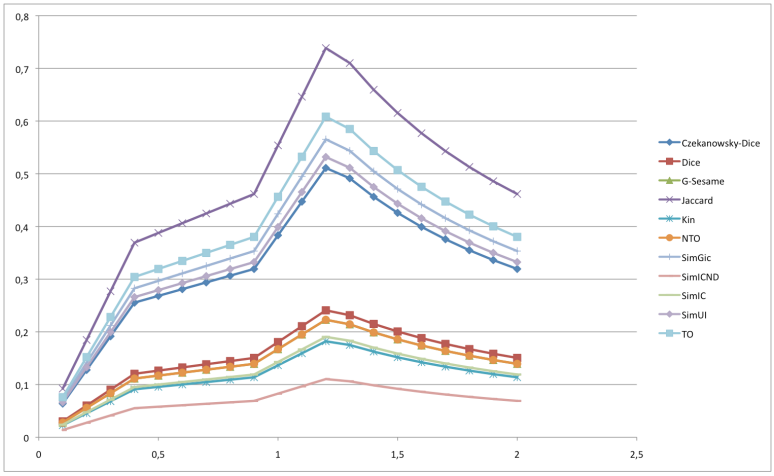


**Fig. 8.** Comparison of Average FC at different Threshold Levels on MIPS

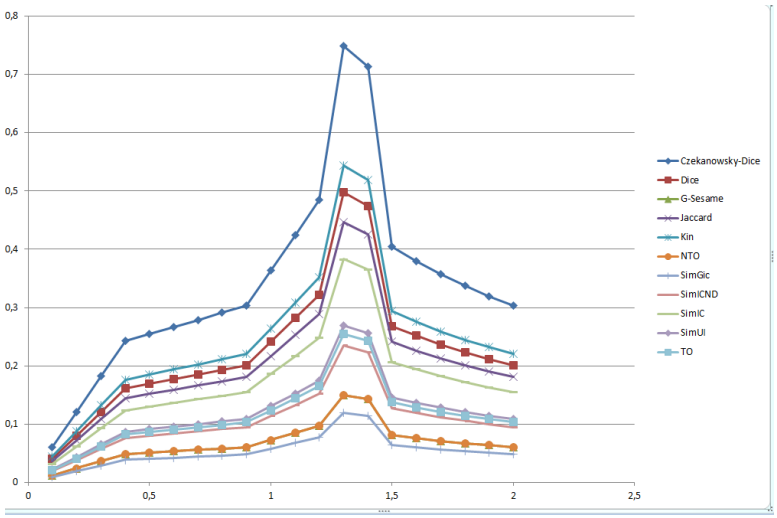


**Fig. 9.** Comparison of Average FC at different Threshold Levels on Annotated High Throughput Complexes Dataset

## 5 Conclusion

Results showed that raw semantic similarity networks contains lot of noise, thus are unsuitable for the analysis. Consequently we proposed a local simplification of networks. Result confirm that mining of simplified networks is a suitable way for extract biologically meaningful knowledge.

## References

1. Cannataro, M., Guzzi, P.H., Sarica, A.: Data mining and life sciences applications on the grid. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery **3**(3) (2013) 216–238
2. Guzzi, P., Mina, M., Guerra, C., Cannataro, M.: Semantic similarity analysis of protein data: assessment with biological features and issues. Briefings in bioinformatics **13**(5) (2012) 569–585
3. Camon, E., Magrane, M., Barrell, D., Lee, V., Dimmer, E., Maslen, J., Binns, D., Harte, N., Lopez, R., Apweiler, R.: The gene ontology annotation (goa) database: sharing knowledge in uniprot with gene ontology. Nucl. Acids Res. **32**(suppl_1) (January 2004) D262–266
4. Pesquita, C., Faria, D., Falcão, A.O., Lord, P., Couto, F.M.: Semantic similarity in biomedical ontologies. PLoS computational biology **5**(7) (July 2009) e1000443
5. Freeman, T., Goldovsky, L., Brosch, M., van Dongen, S., Maziere, P., Grocock, R., Freilich, S., Thornton, J., Enright, A.: Construction, visualization, and clustering of transcription networks from microarray expression data. PLoS Computational Biology **3**(10) (2007) e206

6. Ala, U., Piro, R., Grassi, E., Damasco, C., Silengo, L., Oti, M., Provero, P., Cunto, F.: Prediction of human disease genes by human-mouse conserved coexpression analysis. PLoS Computational Biology **4**(3) (2008) e1000043
7. Rito, T., Wang, Z., Deane, C.M., Reinert, G.: How threshold behaviour affects the use of subgraphs for network comparison. Bioinformatics **26**(18) (2010) i611–i617
8. P., G., M, M.: Investigating bias in semantic similarity measures for analysis of protein interactions. In: Proceedings of 1st International Workshop on Pattern Recognition in Proteomics, Structural Biology and Bioinformatics (PR PS BB 2011). (13th September 2011 2012) 71–80
9. Lee, H., Hsu, A., Sajdak, J., Qin, J., Pavlidis, P.: Coexpression analysis of human genes across many microarray data sets. Genome Res **14** (2004) 1085–1094
10. Ding, C., He, X., Zha, H.: A spectral method to separate disconnected and nearly-disconnected web graph components. Proceedings of the Seventh ACM International Conference on Knowledge Discovery and Data Mining: 26-29 August 2001; San Francisco (2001)
11. Ng, A., Jordan, M., Weiss, Y.: On spectral clustering: analysis and an algorithm. Advances in Neural and Information Processing Systems: 3-8 December 2001; Vancouver (2001)
12. Ma, X., Gao, L.: Biological network analysis:insights into structure and functions. Briefings in Functional Genomics **11**(6) (2012) 434–442
13. : On the functional and structural characterization of hubs in protein-protein interaction networks. Biotechnology Advances **31**(2) (2013) 274 – 286
14. Zhu, X., Gerstein, M., Snyder, M.: Getting connected: analysis and principles of biological networks. Genes & Development **21**(9) (2007) 1010–1024
15. Su, G., Kuchinsky, A., Morris, J.H., States, D.J., Meng, F.: Glay: community structure analysis of biological networks. Bioinformatics **26**(24) (2010) 3135–3137
16. Ji, J., Zhang, A., Liu, C., Quan, X., Liu, Z.: Survey: Functional module detection from protein-protein interaction networks. IEEE Transactions on Knowledge and Data Engineering **99**(PrePrints) (2013) 1
17. Chung, F.: Spectral graph theory. Regional Conference Series in Mathematics, Providence: American Mathematical Society **92** (1994)
18. Cvetković, D., Simić, S.K.: Towards a spectral theory of graphs based on the signless laplacian, ii. Linear Algebra and its Applications **432**(9) (2010) 2257–2272
19. : Spectra and optimal partitions of weighted graphs. Discrete Mathematics **128**(13) (1994) 1 – 20
20. Merris, R.: Laplacian matrices of graphs: a survey. Linear algebra and its applications **197** (1994) 143–176
21. Mohar, B.: The laplacian spectrum of graphs. In: Graph Theory, Combinatorics, and Applications. Volume 2. (1991) 871–898
22. Pesquita, C., Faria, D., Falcao, A., Lord, P., Couto, F.M.: Semantic similarity in biomedical ontologies. PLoS Comput Biol **5**(7) (07 2009) e1000443
23. Wang, H., Zheng, H., Azuaje, F.: Ontology- and graph-based similarity assessment in biological networks. Bioinformatics **26**(20) (October 2010) 2643–2644
24. Guldener, U., Munsterkotter, M., Oesterheld, M., Pagel, P., Ruepp, A., Mewes, H., Stumpflen, V.: Mpact: the mips protein interaction resource on yeast. Nucleic Acids Res **34** (2006) D436–441
25. Cannataro, M., Guzzi, P.H., Veltri, P.: Protein-to-protein interactions: Technologies, databases, and algorithms. ACM Comput. Surv. **43** (December 2010) 1:1–1:36